

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

The Development of a Visual Language for Image Processing Applications

A thesis presented in partial fulfilment of the requirements for
the degree of Doctor of Philosophy in Computer Science at
Massey University, Palmerston North, New Zealand.

Phillip Michael Ngan

1992

006.42

Nga

DC20

Abstract

The research described in this thesis is based on the hypothesis that computer support for the heuristic development of image processing algorithms can be improved by the provision of a human-computer interface that is suited to the task. Current interfaces are largely text based and are not specifically designed to provide this support. It is suggested that an interface incorporating aspects of menu-based, direct manipulation, and visual languages, can provide the necessary support.

The research of this thesis begins with an analysis of the task of image processing algorithm development. It is found that in development, algorithms are more appropriately viewed as data-oriented networks of imaging operations than as process-oriented lists. The representation of algorithms in most current interfaces, particularly in text based systems, do not clearly convey the multi-threaded data paths in an algorithm. However, a data-oriented representation expresses such parallel paths clearly and naturally.

The second finding of the analysis is that human designers employ a set of problem solving strategies or heuristics in the interactive development of algorithms. These strategies include the top-down decomposition of the imaging task, the identification and focus of critical sub-goals, the progressive refinement of an algorithm, and the modification of existing algorithms. These heuristics are used implicitly in the development of algorithms, but the ease with which they are used in text based interfaces is restricted by the lack of appropriate interactive facilities.

An evaluation of interface techniques suggests that an interface that combines aspects of menu-based, direct manipulation, and visual languages, can support the required interaction for the heuristic development of algorithms. The required data-flow view can be provided by an iconic data flow language. Such a representation is highly visible, and can be interpreted by a user at a glance. Quick and convenient specification and editing of a data flow network can be performed via direct manipulation interaction facilities. The search for suitable operations can be facilitated by menu systems.

On the basis of the arguments for the adoption of a data-flow representation of algorithms, a problem solving approach to algorithm development, and highly interactive facilities, a software package, called OpShop, has been implemented. Examples which compare OpShop to text based systems show that four major tasks involved in algorithm development are better supported with the new interface. These tasks are the visualisation of multi-threaded data paths, the interactive experimentation with algorithm parameters, the modification of algorithm topology, and the comparison of alternative algorithms. In these examples, the OpShop software represents the tangible outcome of the design for an interface that specifically supports the heuristic development of image processing algorithms.

Table of Contents

	Abstract.....	ii
	Preface.....	vi
	Acknowledgments.....	ix
1	Introduction.....	1
1.1	The context.....	1
1.2	The approach.....	3
1.3	Thesis overview.....	4
2	Image Processing Algorithm Development.....	6
2.1	Introduction.....	6
2.2	General structure of an algorithm.....	9
2.2.1	Outside world.....	10
2.2.2	General image.....	11
2.2.3	Segmented image.....	12
2.2.4	Compact structures.....	13
2.2.5	Shape measurement.....	14
2.2.6	Pattern classification.....	15
2.2.7	An example algorithm.....	15
2.3	Data-oriented view of algorithms.....	17
2.4	Algorithm development is problem solving.....	18
2.5	The solution graph.....	19
2.6	Solution development.....	20
2.7	The heuristic approach.....	22
2.7.1	The broad decomposition of the task.....	22
2.7.2	Identification of a critical subgoal.....	22
2.7.3	Jumping to an arbitrary location.....	23
2.7.4	Application of well known techniques.....	23
2.7.5	Exemplar based development.....	24
2.7.6	Progressive refinement.....	24
2.8	Summary and Conclusions.....	25
3	Human Computer Interface Techniques.....	26
3.1	Introduction.....	26
3.2	Command line interfaces.....	27
3.2.1	Advantages.....	28
3.2.2	Disadvantages.....	29

3.3	Menu based interface.....	30
3.3.1	Advantages.....	33
3.3.2	Disadvantages.....	33
3.4	Direct manipulation interface.....	34
3.4.1	Key characteristics.....	37
3.4.2	Disadvantages.....	38
3.5	Visual language interface.....	39
3.5.1	Visual programming vs program visualisation.....	39
3.5.2	Visually transformed vs naturally visual.....	41
3.5.3	Program responsiveness.....	42
3.5.4	Advantages.....	43
3.5.5	Disadvantages.....	43
3.6	Summary and conclusions.....	44
4	An Interface Design.....	46
4.1	Design philosophy.....	46
4.1.1	Stepwise refinement.....	48
4.1.2	Dynamic exploration.....	48
4.2	Generation.....	49
4.2.1	An iconic data flow language for image processing.....	49
4.3	Execution.....	53
4.3.1	Specification of input and output data.....	53
4.3.2	Specification of parameter values.....	54
4.3.3	Operation invocation.....	56
4.4	Evaluation of the results.....	57
4.5	Summary and Conclusions.....	58
5	OpShop: An Implementation.....	60
5.1	An overview of OpShop.....	60
5.2	Elements of the visual language environment.....	63
5.2.1	Whiteboard.....	63
5.2.2	Operations.....	64
5.2.3	Algorithms.....	68
5.2.4	Subflows.....	71
5.3	User Interaction.....	74
5.3.1	Execution.....	74
5.3.2	Parameter exploration.....	74
5.3.3	Topology exploration.....	75
5.4	The OpShop software design.....	77
5.4.1	Why THINK C?.....	77
5.4.2	Data structures.....	79
5.4.3	A data-driven execution scheme.....	83

5.5	Feasibility of continuous interaction	88
5.6	Summary and conclusions	90
6	OpShop Examples	92
6.1	Introduction	92
6.2	Data flow view of an algorithm	93
6.2.1	Colour classification	93
6.2.2	A command line implementation.....	95
6.2.3	OpShop implementation.....	96
6.3	Parameter exploration	97
6.3.1	The Abingdon Cross benchmark.....	97
6.3.2	A command line implementation.....	98
6.3.3	An OpShop implementation.....	100
6.4	Topological exploration	102
6.4.1	Segmentation of a non-uniformly illuminated scene	102
6.4.2	A command line implementation.....	105
6.4.3	An OpShop implementation.....	106
6.5	Choosing between algorithm alternatives.....	107
6.5.1	Generation of alternative solutions.....	107
6.5.2	Command line implementation.....	107
6.5.3	An OpShop implementation.....	108
6.6	Summary	109
7	Summary and Conclusions.....	110
7.1	Suggestions for future work	113
	References.....	117
	Summary of the OpShop Software.....	127
A1.1	Introduction	127
A1.2	System requirements.....	128
A1.3	Description of Operations.....	128
A1.3.1	General Menu	128
A1.3.2	Preprocessing Menu.....	135
A1.3.3	Segmentation Menu	140
A1.3.4	Measurement Menu.....	145
A1.3.5	Miscellaneous interface elements.....	145
A1.4	Operation Icons.....	146
A1.4.1	General Menu	146
A1.4.2	Preprocessing Menu.....	146
A1.4.3	Segmentation Menu	146
A1.4.4	Measurement Menu.....	146

Preface

Background

This project grew out of a perceived need for a highly interactive computing environment to support the heuristic approach to imaging algorithm development. I was introduced to the field of image processing while I was an undergraduate Electrical Engineering student at the University of Canterbury. During the course of studying my B.E. and M.E. I had used, or at least been exposed to, four interactive image processing systems. These systems, written by post-graduate students of Richard Bates and Bob Hodgson, served their intended purposes well; that of providing computing environments to support research into image processing. However, it never occurred to me then, that good engineers can (sometimes) create bad interfaces¹. These were fine systems for performing post graduate research, but they were not necessarily the easiest to use.

Richard Bates, on finding out that I was considering continuing my studies overseas, strongly recommended Mark Apperley to me as a supervisor for Ph.D. studies. As it turned out, Mark Apperley had a happy blend of an electrical engineering background and research interests in both image processing and human-computer interaction. After enrolling in a Ph.D degree course, Bob Hodgson and Don Bailey, who had been key figures in the image processing work at Canterbury, joined the staff at Massey University. They accepted invitations to co-supervise my Ph.D. project. So it was with a definite research objective and a proficient team of supervisors that I started on the work reported in this thesis.

Original contributions

This thesis describes the development of an interactive user interface for image processing applications. The primary goal of the interface is to facilitate the development of imaging algorithms by enhancing a user's ability to directly interact with the imaging task. In the course of pursuing this goal a number of original contributions were made:

- *A new viewpoint concerning the task of image processing algorithm development.* It is recognised that, for development, an algorithm is more appropriately viewed as a data-oriented network of operations than as a process-oriented list of imaging operations. This latter view is prevalent in most current systems for imaging algorithm development.
- *A data-oriented view, called the solution graph, was formulated as a graphical model to represent image processing algorithms.*

¹ Genter, D.R. & Grudin, J. (1990): Why good engineers (sometimes) create bad interfaces, *CHI'90 Conference Proceedings*, Seattle, Washington, 1 - 5 April, 1990.

- *Another new viewpoint concerning the task of image processing algorithm development.* The pragmatic approach to the development of an algorithm can be regarded as an example of a problem solving task. It is demonstrated that the pragmatic approach involves the application of a set of heuristics, to increase the likelihood of finding a satisfactory solution. The developed interface incorporates a graphical language that enables the explicit representation of these heuristics.
- *Demonstration of a user-oriented approach to software design.* The user-oriented approach to software design was shown to be feasible provided that a designer is fully aware of the user's needs and interests regarding the task, and that the designer carefully applies interaction techniques to support users' concerns. In this study, the end-product is an interactive environment for the development of imaging algorithms.
- *Demonstration of ways to simplify algorithm development.* Four demonstrations were given to show how the implemented design simplified the development of imaging algorithms.

As with most software projects, the process of development tends to be more evolutionary than sequential. In practice, a development of a product rarely proceeds in the distinct stages of analysis, design, implementation, and testing, as indicated by the structure of this thesis. Rather, a typical development process involves iteration of these stages. The research for this thesis was conducted in such an evolutionary manner. Many of the insights and perceptions presented were a product of hindsight and reflection of previous iterations. Aspects of the design were formalised before, during, and after, the writing of software. Although the form of a thesis constrains one to present material in a linear sequence of logical steps, the reader should be aware that the ideas were not necessarily developed in that order.

Publications

The following publications and presentations were prepared during the research for this thesis:

- Ngan, P.M., Apperley, M.D. & Hodgson, R.M. (1989): Towards a user model for image processing and analysis, *Proceedings of the 4th New Zealand Image Processing Workshop*, Auckland Industrial Development Division, Department of Scientific and Industrial Research, Auckland, New Zealand, 14-15 August, 1989, 94-100.
- White A.G. & Ngan, P.M. (1989): The measurement of red colour of apple fruit using digital imaging, *Proceedings of the 4th New Zealand Image Processing Workshop*, Auckland Industrial Development Division, Department of Scientific and Industrial Research, 14-15 August, 1989, 13-19.
- Ngan, P.M., Apperley, M.D. & Hodgson, R.M. (1990): The user-oriented development of an interface for image processing, *Proceedings of the 5th New Zealand Image Processing Workshop*, Massey University, Palmerston North, New Zealand, 9-10 August, 1990, 63-68.
- Cochrane, T., Matthew, C., Apperley, M.D. & Ngan, P.M. (1990): Plant root length and diameter determination using an image analysis thinning algorithm, *Proceedings Agronomy Society of New Zealand*, 20, 1990, 77-82.
- Ngan, P.M. (1991): OpShop: an iconic programming system for image processing, *Proceedings of the Australasian Apple University Consortium Conference*, Australian National University, Canberra, Australia, 1-4 July 1991.
- Ngan, P.M. & Apperley, M.D. (1992): Opportunistic design in image processing algorithm development, submitted to *Journal of Visual Languages and Computing*.

Acknowledgments

During this Ph.D., I was fortunate to have had three exemplary researchers as my supervisors: Mark Apperley, Bob Hodgson, and Don Bailey. Each tended to help me in different aspects of my study, yet these differences were complementary. This complementarity somewhat parallels the concept of generation-execution-evaluation discussed in the thesis. Mark, my first supervisor, helped me generate the ideas of this work through his acute insight of important and promising avenues of research. Don helped in the carrying out of the work by providing timely and key advice during both the programming and writing phases. Bob often served as the beta-tester for the research and provided greatly appreciated constructive criticism to ensure the work was carried out to a full professional standard. I greatly appreciate the support, encouragement, and openness, that my supervisors have extended to me during this apprenticeship in the craft of research.

Kirsten, my fiancé and constant source of love and support during this work, exercised her meticulous editing skills on many of the chapter drafts.

I appreciate the contribution of Paul Mudgeway relating to the graphical design of the icon symbols for OpShop. His work demonstrates the value of involving graphics designers in graphics-oriented HCI projects.

Throughout the Ph.D., I received financial support in the form of the VC's Ph.D. Study Award. I am indebted (and thankfully not financially!) to Massey University for this assistance.

A man can do nothing better than to eat and drink and find satisfaction in his work. This too, I see, is from the hand of God ... (Eccles. 2:24). This work has been carried out to the glory of God.