# **Evolutionary Synthesis of Pattern Recognition Systems**

## Monographs in Computer Science

#### Abadi and Cardelli, A Theory of Objects

Benosman and Kang [editors], Panoramic Vision: Sensors, Theory and Applications

Broy and Stølen, Specification and Development of Interactive Systems: FOCUS on Streams, Interfaces, and Refinement

Brzozowski and Seger, Asynchronous Circuits

Cantone, Omodeo, and Policriti, Set Theory for Computing: From Decision Procedures to Declarative Programming with Sets

Castillo, Gutiérrez, and Hadi, Expert Systems and Probabilistic Network Models

Downey and Fellows, Parameterized Complexity

Feijen and van Gasteren, On a Method of Multiprogramming

Herbert and Spärck Jones [editors], **Computer Systems: Theory, Technology, and Applications** 

Leiss, Language Equations

McIver and Morgan [editors], Programming Methodology

McIver and Morgan, Abstraction, Refinement and Proof for Probabilistic Systems

Misra, A Discipline of Multiprogramming: Program Theory for Distributed Applications

Nielson [editor], ML with Concurrency

Paton [editor], Active Rules in Database Systems

Selig, Geometric Fundamentals of Robotics, Second Edition

Tonella and Potrich, Reverse Engineering of Object Oriented Code

Bir Bhanu Yingqiang Lin Krzysztof Krawiec

## **Evolutionary Synthesis of Pattern Recognition Systems**



Bir Bhanu Center for Research in Intelligent Systems University of California at Riverside Bourns Hall RM B232 Riverside, CA 92521 Yingqiang Lin Center for Research in Intelligent Systems University of California at Riverside Bourns Hall RM B232 Riverside CA 92521 Krzysztof Krawiec Center for Research in Intelligent Systems University of California at Riverside Bourns Hall RM B232 Riverside CA 92521

Series Editors David Gries Dept. of Computer Science Cornell University Upson Hall Ithaca NY 14853-7501

Fred B. Schneider Dept. Computer Science Cornell University Upson Hall Ithaca NY 14853-7501

Library of Congress Cataloging-in-Publication Data Bhanu, Bir.

Evolutionary Synthesis of Pattern Recognition Systems /Bir Bhanu, Yingqiang Lin, and Krzysztof Krawiec.

p. cm. –(Monographs in Computer Science) Includes bibliographic references and index.

ISBN 0-387-21295-7 e-ISBN 0-387-24452-2 Printed on acid-free paper.

© 2005 Springer Science+Business Media, Inc.

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, Inc., 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now know or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks and similar terms, even if the are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed in the United States of America. (BS/DH)

9 8 7 6 5 4 3 2 1 SPIN (HC) 10984741 / SPIN (eBK) 11381136

springeronline.com

## Contents

LIST OF FIG	URES	5	xi
LIST OF TAE	BLES		xvii
PREFACE			xxi
CHAPTER 1	INTI	RODUCTION	1
	1.1	<b>Object Detection and Recognition Problem</b>	1
	1.2	Motivations for Evolutionary Computation	3
	1.3	Evolutionary Approaches for Synthesis and Analysis	5
	1.4	Outline of the Book	7
CHAPTER 2	FEA DET	TURE SYNTHESIS FOR OBJECT ECTION	11
	2.1	Introduction	11
	2.2	Motivation and Related Research 2.2.1 Motivation 2.2.2 Related research	<b>12</b> 12 13
	2.3	<b>Genetic Programming for Feature Synthesis</b> 2.3.1 Design considerations	<b>15</b> 16

		2.3.2 Selection, crossover and mutation	20
		programming	23
	2.4	Experiments	27
		2.4.1 SAR Images	28
		2.4.2 Infrared and color images	45
		2.4.3 Comparison with GP with hard limit on	
		composite operator size	53
		2.4.4 Comparison with image-based GP	62
		2.4.5 Comparison with a traditional ROI	
		extraction algorithm	68
		2.4.6 A multi-class example	73
	2.5	Conclusions	78
CHAPTER 3	MDI PRC	L-BASED EFFICIENT GENETIC OGRAMMING FOR OBJECT DETECTION	79
	3.1	Introduction	79
	3.2	Motivation and Related Research	80
	3.3	Improving the Efficiency of GP	84
		3.3.1 MDL principle-based fitness function	84
		3.3.2 Genetic programming with smart crossover	
		and smart mutation	86
		3.3.3 Steady-state and generational genetic	
		programming	90
	3.4	Experiments	93
		3.4.1 Road extraction	95
		3.4.2 Lake extraction	103
		3.4.3 River extraction	105
			109
		3.4.4 Field extraction	100
		3.4.4 Field extraction 3.4.5 Tank extraction	110

	3.5	Conclusions	119
CHAPTER 4	FEA DET	TURE SELECTION FOR OBJECT ECTION	121
	4.1	Introduction	121
	4.2	Motivation and Related Research	123
	4.3	Feature Evaluations and Selection	125
		4.3.1 Feature selection	126
		4.3.2 Various criteria for fitness function	127
	4.4	System Description	131
		4.4.1 CFAR detector	131
		4.4.2 Feature extractor	134
		4.4.3 GA for feature selection	142
	4.5	Experiments	143
		4.5.1 MDL principle-based fitness function	144
		4.5.2 Other fitness functions	153
		4.5.3 Comparison and analysis	154
	4.6	Conclusions	164
CHAPTER 5	EVO OBJ	LUTIONARY FEATURE SYNTHESIS FOR ECT RECOGNITION	165
	5.1	Introduction	165
	5.2	Motivation and Related Research	167
		5.2.1 Motivation	167
		5.2.2 Related research	168
	5.3	Coevolutionary GP for Feature Synthesis	170
		5.3.1 Design considerations	170
		5.3.2 Selection, crossover and mutation	174

		5.3.3 Generational coevolutionary genetic	
		programming	175
		5.3.4 Bayesian classifier	177
	5.4	Experiments	177
		5.4.1 Distinguish objects from clutter	178
		5.4.2 Recognize objects	182
		5.4.3 Comparison with other classification	
		algorithms	193
		5.4.4 Discussion	197
	5.5	Conclusions	199
CHAPTER 6	LINI	AR GENETIC PROGRAMMING FOR	
	OBJ	ECT RECOGNITION	201
	6.1	Introduction	201
	6.2	Explicit Feature Construction	202
	6.3	Linear Genetic Programming	205
	6.4	Evolutionary Feature Programming	206
		6.4.1 Representation and its properties	208
		6.4.2 Execution of feature extraction procedure	216
		6.4.3 Locality of representation	218
		6.4.4 Evaluation of solutions	221
	6.5	<b>Coevolutionary Feature Programming</b>	223
	6.6	Decomposition of Explicit Feature	
		Construction	226
	6.7	Conclusions	232

CHAPTER 7	APP PRC	LICATIONS OF LINEAR GENETIC OGRAMMING FOR OBJECT RECOGNITION	233
	7.1	Introduction	233
	7.2	Technical Implementation	234
	7.3	<b>Common Experimental Framework</b> 7.3.1 Background knowledge 7.3.2 Parameter settings and performance	<b>235</b> 235
		measures	237
	7.4	<b>Recognition of Common Household Objects</b> 7.4.1 Problem and data 7.4.2 Parameter settings 7.4.3 Results	<b>238</b> 238 240 241
	7.5	<ul> <li>Object Recognition in Radar Modality</li> <li>7.5.1 Problem decomposition at instruction level</li> <li>7.5.2 Binary classification tasks</li> <li>7.5.3 On-line adaptation of population number</li> <li>7.5.4 Scalability</li> <li>7.5.5 Recognizing object variants</li> <li>7.5.6 Problem decomposition at decision level</li> </ul>	245 247 252 256 259 260 264
	7.6	Analysis of Evolved Solutions	268
	7.7	Conclusions	275
CHAPTER 8	SUM	IMARY AND FUTURE WORK	277
	8.1	Summary	277
	8.2	Future Work	280
REFERENCE	S		282
INDEX			291

## List of Figures

Figure 2.1. Steady-state genetic programming algorithm.	25
Figure 2.2. Generational genetic programming algorithm.	26
Figure 2.3. Training SAR image containing road.	30
Figure 2.4. Sixteen primitive feature images of training SAR image	
containing road.	31
Figure 2.5. Learned composite operator tree	32
Figure 2.6. Fitness versus generation (road vs. field).	32
Figure 2.7. Utility of primitive operators and primitive feature images	34
Figure 2.8. Feature images output by the nodes of the best composite	
operator. The ouput of the root node is shown Figure 2.3(c)	35
Figure 2.9. ROIs extracted from the output images of the nodes of the	
best composite operator. The fitness value is shown for the	
entire image. The ouput of the root node is shown Figure	
2.3(d)	36
Figure 2.10. Testing SAR images containing road.	37
Figure 2.11. Training SAR image containing lake	38
Figure 2.12. Testing SAR image containing lake.	38
Figure 2.13. Training SAR image containing river	39
Figure 2.14. Learned composite operator tree	40
Figure 2.15. Fitness versus generation (river vs. field).	40
Figure 2.16. Testing SAR image containing river.	40
Figure 2.17. Training SAR image containing field.	41
Figure 2.18. Testing SAR image containing field	42
Figure 2.19. Training SAR image containing tank.	42
Figure 2.20. Learned composite operator tree in LISP notation	43
Figure 2.21. Fitness versus generation (T72 tank).	43
Figure 2.22. Testing SAR image containing tank.	44
Figure 2.23. Training IR image containing a person	46

Figure 2.24. Learned composite operator tree in LISP notation	47
Figure 2.25. Fitness versus generation (person).	47
Figure 2.26. Testing IR images containing a person.	49
Figure 2.27. Training RGB color image containing car.	50
Figure 2.28. Learned composite operator tree in LISP notation	50
Figure 2.29. Fitness versus generation (car).	51
Figure 2.30. Testing RGB color image containing car.	51
Figure 2.31. Training and testing RGB color image containing SUV	52
Figure 2.32. Results on SAR images containing road.	55
Figure 2.33. Learned composite operator tree in LISP notation	56
Figure 2.34. Fitness versus generation (road vs. field)	56
Figure 2.35. Results on SAR images containing lake	57
Figure 2.36. Results on SAR images containing river	58
Figure 2.37. Learned composite operator tree in LISP notation	59
Figure 2.38. Fitness versus generation (river vs. field)	59
Figure 2.39. Results on SAR images containing field.	60
Figure 2.40. Results on SAR images containing tank.	61
Figure 2.41. Learned composite operator tree in LISP notation	61
Figure 2.42. Fitness versus generation (T72 tank)	61
Figure 2.43. Results on SAR images containing road.	64
Figure 2.44. Results on SAR images containing lake	64
Figure 2.45. Results on SAR images containing river	66
Figure 2.46. Results on SAR images containing field.	66
Figure 2.47. ROIs extracted by the traditional ROI extraction algorithm	71
Figure 2.48. ROIs extracted by the GP-evolved composite operators	72
Figure 2.49. SAR image containing lake, road, field, tree and shadow	74
Figure 2.50. Lake, road and field ROIs extracted by the composite	
operators learned in Examples 1, 2 and 4	74
Figure 2.51. Histogram of pixel values (range 0 to 200) within lake and	
road regions	75
Figure 2.52. SAR image containing lake and road	75
Figure 2.53. lake and road ROIs extracted from training images	76
Figure 2.54. Lake, road and field ROIs extracted from the testing image	77
Figure 2.55. Lake, road and field ROIs extracted by the traditional	
algorithm	77

Figure 3.1. Modified Steady-state genetic programming9	91
Figure 3.2. Modified Generational genetic programming	92
Figure 3.3. Training SAR image containing road.	95
Figure 3.4. Learned composite operator tree in LISP notation	96
Figure 3.5. Fitness versus generation (road vs. field).	97
Figure 3.6. Frequency of primitive operators and primitive feature	
images	98
Figure 3.7. Feature images output at the nodes of the best composite	
operator learned by smart GP10	)0
Figure 3.8. ROIs extracted from the output images at the nodes of the	
best composite operator from smart GP. The goodness value	
is shown for the entire image10	)1
Figure 3.9. Testing SAR images containing road10	)2
Figure 3.10. Training SAR image containing lake10	)3
Figure 3.11. Testing SAR image containing lake 10	)4
Figure 3.12. Learned composite operator tree in LISP notation 10	)5
Figure 3.13. Training SAR image containing river10	)5
Figure 3.14. Learned composite operator tree in LISP notation 10	)6
Figure 3.15. Fitness versus generation (river vs. field)10	)7
Figure 3.16. Testing SAR image containing river 10	)7
Figure 3.17. Training SAR image containing field10	98
Figure 3.18. Testing SAR image containing field10	)9
Figure 3.19. Learned composite operator tree in LISP notation	10
Figure 3.20. Training SAR image containing a tank11	11
Figure 3.21. Learned composite operator tree in LISP notation	12
Figure 3.22. Fitness versus generation (T72 tank) 11	12
Figure 3.23. Testing SAR image containing tank	13
Figure 3.24. The average goodness of the best composite operators	
versus generation11	15

Figure 4.1. System diagram for feature selection	. 125
Figure 4.2. SAR image and CFAR detection result.	. 133
Figure 4.3. Example of the standard deviation feature	. 135

Figure 4.4. Example of the fractal dimension feature	136
Figure 4.5. Examples of images used to compute size features (4-6) for	
(a) object and (b) clutter	138
Figure 4.6. Fitness values vs. generation number	150
Figure 4.7. Training error rates vs. generation number.	151
Figure 4.8. The number of features selected vs. generation number	152
Figure 4.9. Average performance of various fitness functions	162

Figure 5.1. System diagram for object recognition using coevolutionary	
genetic programming17	71
Figure 5.2. Computation of fitness of jth composite operator of ith sub-	
population17	13
Figure 5.3. Generational coevolutionary genetic programming17	16
Figure 5.4. Example object and clutter SAR images17	<i>1</i> 9
Figure 5.5. Composite operator vector learned by CGP18	32
Figure 5.6. Five objects used in recognition	35
Figure 5.7. Composite operator vector learned by CGP with 5 sub-	
populations	39
Figure 5.8. Composite operator vector learned by CGP19	)2

Figure 6.1. The outline of evolutionary feature programming (EFP)	.207
Figure 6.2. Graph representation of an exemplary feature extraction	
procedure	.211
Figure 6.3. Details on genotype-phenotype mapping.	.212
Figure 6.4. Execution of feature extraction procedures for a single	
training example (image) x	.216
Figure 6.5. Comparison of particular decomposition levels for	
evolutionary feature programming	.231

Figure 7.1. Software implementation of CVGP. Dashed-line components	
implement background knowledge2	35
Figure 7.2. Exemplary images from COIL20 database (one representative	
per class)	38
Figure 7.3. Apparent size changes resulting from MBR cropping for	
different aspects of two selected objects from the COIL20	
database	39
Figure 7.4. Fitness of the best individual, test set recognition ratio, and	
test set TP ratio for binary COIL20 experiments (means over	
10 runs and 0.95 confidence intervals)	42
Figure 7.5. Test set FP ratio and tree size for binary COIL20 experiments	
(means over 10 runs and 0.95 confidence intervals)	43
Figure 7.6. Decision tree h used by the final recognition system evolved	
in one of the COIL20 binary experiments	45
Figure 7.7. Selected vehicles represented in MSTAR database24	49
Figure 7.8. Exemplary images from the MSTAR database	49
Figure 7.9. Three vehicles and their correspondings SAR images	50
Figure 7.10. Fitness graph for binary experiment (fitness of the best	
individual for each generation)	54
Figure 7.11. True positive (TP) and false positive (FP) ratios for binary	
recognition tasks (testing set, single recognition systems).	
Chart presents averages over 10 independent synthesis	
processes and their .95 confidence intervals	56
Figure 7.12. True positive (TP) and false positive (FP) ratios for binary	
recognition tasks (testing set, single recognition systems,	
adaptive CC). Chart presents averages over 10 independent	
synthesis processes and their 0.95 confidence intervals	59
Figure 7.13 Test set recognition ratios of compound recognition systems	
for different number of decision classes	61
Figure 7.14 Curves for different number of decision classes (base	
classifier: SVM)	62
Figure 7.15. True positive and false positive ratios for hinary recognition	02
tacks (testing set compound recognition systems)	67
Figure 7.16 Representative images of objects used in experiments	
concerning object variants (all nictures taken at 101°	
concerning object variants (an pictures taken at 191	

aspect/azimuth, cropped to central 64×64 pixels, and magnified to show details)	.267
Figure 7.17. Image of the ZSU class taken at 6° azimuth angle (cropped	
to input size, i.e. 48×48 pixels)	.269
Figure 7.18. Processing carried out by one of the evolved solutions (individual 1 of 4; see text for details)	.271
Figure 7.19. Processing carried out by one of the evolved solutions (individual 2 of 4; see text for details)	.272
Figure 7.20. Processing carried out by one of the evolved solutions (individual 3 of 4; see text for details)	.273
Figure 7.21. Processing carried out by one of the evolved solutions (individual 4 of 4; see text for details)	.274

## **List of Tables**

## Chapter 2

Table 2.1. Sixteen primitive feature images used as the s	et of terminals17
Table 2.2. Seventeen primitive operators	19
Table 2.3. The performance on various examples of SAF	c images29
Table 2.4. The performance results on IR and RGB color	: images45
Table 2.5. The performance results on various examples	of SAR images.
The hard limit on composite operator size is t	ısed54
Table 2.6. The performance results of image-based GP	on various SAR
images.	65
Table 2.7. Average training time of region GP and	l image GP (in
seconds)	67
Table 2.8. Comparison of the performance of traditional	I ROI extraction
algorithm and composite operators generated	by GP70
Table 2.9. Average running time (in seconds) of the cor	nposite operators
and the traditional ROI extraction algorithm.	
algorithm and composite operators generated Table 2.9. Average running time (in seconds) of the cor	by GP70 nposite operators

Table 3.1. The performance of the best composite operators from normal	
and smart GPs	94
Table 3.2. The average goodness of the best composite operators from	
normal and smart GPs.	116
Table 3.3. The average size and performance of the best composite	
operators from normal and smart GPs	117
Table 3.4. Average training time of Normal GP and Smart GP	117
Table 3.5. The average performance of the best composite operators from	
smart GPs with and without the public library	118
Table 3.6. Average running time (in seconds) of the composite operators	
from normal and smart GPs	118

Table 4.1. Experimental results with 300 training target and clutter chips	
(MDL, equation (4.2); $\varepsilon = 0.002$ )	146
Table 4.2. Experimental results with 500 training target and clutter chips	
(MDL, equation (4.2); $\varepsilon = 0.0015$ )	147
Table 4.3. Experimental results with 700 training target and clutter chips	
(MDL, equation (4.2); $\varepsilon = 0.0015$ )	148
Table 4.4. Experimental results with 700 training target and clutter chips	
(MDL, equation (4.2); $\varepsilon = 0.0011$ )	149
Table 4.5. Experimental results with 500 training target and clutter chips	
(penalty function, equation (4.4); $\varepsilon = 0.0015$ )	.155
Table 4.6. Experimental results with 500 training target and clutter chips	
(penalty and # of features, equation (4.5); $\gamma = 0.1$ ; $\varepsilon =$	
0.0015)	.156
Table 4.7. Experimental results with 500 training target and clutter chips	
(penalty and # of features, equation (4.5); $\gamma = 0.3$ ; $\varepsilon =$	
0.0015)	.157
Table 4.8. Experimental results with 500 training target and clutter chips	
(penalty and # of features, equation (4.5); $\gamma = 0.5$ ; $\varepsilon =$	
0.0015).	158
Table 4.9. Experimental results with 500 training target and clutter chips	
(error rate and # of features, equation (4.6); $\gamma = 0.1$ ; $\varepsilon =$	
0.0015).	159
Table 4.10. Experimental results with 500 training target and clutter	
chips (penalty and # of features, equation (4.6); $\gamma = 0.3$ ; $\varepsilon = 0.0000$	1.00
0.0015)	160
Table 4.11. Experimental results with 500 training target and clutter	
chips (penalty and # of features, equation (4.6); $\gamma = 0.5$ ; $\varepsilon = 0.015$	1.61
0.0015)	161
Table 4.12. Experimental results using only one feature for	160
ascrimination (target cnips = $500$ , cutter cnips = $500$ )	102
Table 4.15. The number of times each feature is selected in MDL	162
Experiments 1, 2 and 4.	103

Table 5.1. Twelve primitive operators.	172
Table 5.2. Parameters of CGP used throughout the experiments	178
Table 5.3. Recognition rates of 20 primitive features.	180
Table 5.4. Performance of composite and primitive features on	
object/clutter discrimination	181
Table 5.5. Recognition rates of 20 primitive features (3 objects).	187
Table 5.6. Performance of composite and primitive features on 3-object	
discrimination	188
Table 5.7. Recognition rates of 20 primitive features (5 objects).	190
Table 5.8. Performance of composite and primitive features on 5-object	
discrimination	191
Table 5.9. Average recognition performance of multi-layer neural	
networks trained by backpropagation algorithms (3 objects)	195
Table 5.10. Average recognition performance of multi-layer neural	
networks trained by backpropagation algorithms (5 objects)	196
Table 5.11. Recognition performance of C4.5 classification algorithm	197

\_\_\_\_\_

Table 7.1. Elementary operations used in the visual learning experiments	
(k and l denote the number of the input and output arguments,	
respectively)	6
Table 7.2. Parameter settings for COIL20 experiments24	1
Table 7.3. Description of data for the experiment concerning cooperation	
on genome level	0
Table 7.4. Performance of recognition systems evolved by means of	
cooperation at genome level25	1
Table 7.5. Test set confusion matrix for selected EFP recognition system25	1
Table 7.6. Test set confusion matrix for selected CFP recognition system25	1
Table 7.7. True positive (TP) and false positive (FP) ratios for SAR	
binary recognition tasks (testing set). Table presents averages	
over 10 independent synthesis processes and their 0.95	
confidence intervals25	5
Table 7.8. True positive (TP) and false positive (FP) ratios for SAR	
binary recognition tasks (testing set, CFP-A; means over 10	

independent synthesis processes and 0.95 confidence intervals)	257
Table 7.9. Mean and maximum number of populations for SAR binary	
recognition tasks (CFP-A)	258
Table 7.10. Confusion matrices for recognition of object variants for 2-	
class recognition system.	262
Table 7.11. Confusion matrices for recognition of object variants for 4-	
class recognition system.	263
Table 7.12. True positive and false positive ratios for binary recognition	
tasks (testing set, off-line decision level decomposition)	266

## Preface

Designing object detection and recognition systems that work in the real world is a challenging task due to various factors including the high complexity of the systems, the dynamically changing environment of the real world and factors such as occlusion, clutter, articulation, and various noise contributions that make the extraction of reliable features quite difficult. Furthermore, features useful to the detection and recognition of one kind of object or in the processing of one kind of imagery may not be effective in the detection and recognition of another kind of object or in the processing of another kind of imagery. Thus, the detection and recognition system often needs thorough overhaul when applied to other types of images different from the one for which the system was designed. This is very uneconomical and requires highly trained experts. The purpose of incorporating learning into the system design is to avoid the time consuming process of feature generation and selection and to lower the cost of building object detection and recognition systems.

Evolutionary computation is becoming increasingly important for computer vision and pattern recognition fields. It provides a systematic way of synthesis and analysis of object detection and recognition systems. With learning incorporated, the resulting recognition systems will be able to automatically generate new features on the fly and cleverly select a good subset of features according to the type of objects and images to which they are applied. The system will be flexible and can be applied to a variety of objects and images.

This book investigates evolutionary computational techniques such as genetic programming (GP), linear genetic programming (LGP), coevolutionary genetic programming (CGP) and genetic algorithms (GA) to automate the synthesis and analysis of object detection and recognition systems. The ultimate goal of the learning approaches presented in this book is to lower the cost of designing object detection and recognition systems and build more robust and flexible systems with human-competitive performance.

The book presents four important ideas.

*First*, this book shows the efficacy of GP and CGP in synthesizing effective composite operators and composite features from domain-independent primitive image processing operations and primitive features (both elementary and complex) for object detection and recognition. It explores the role of domain knowledge in evolutionary computational techniques for object recognition. Based on GP and CGP's ability to synthesize effective features from simple features not specifically designed for a particular kind of imagery, the cost of building object detection and recognition systems is lowered and the flexibility of the systems is increased. More importantly, a large amount of unconventional features are explored by GP and CGP and these unconventional features yield exceptionally good detection and recognition performance in some cases, overcoming the human experts' limitation of considering only a small number of conventional features.

Second, smart crossover, smart mutation and a new fitness function based on the minimum description length (MDL) principle are designed to improve the efficiency of genetic programming. Smart crossover and smart mutation are designed to identify and keep the effective components of composite operators from being disrupted and a MDL-based fitness function is proposed to address the well-known code bloat problem of GP without imposing severe restriction on the GP search. Compared to normal GP, smart GP algorithm with smart crossover, smart mutation and a MDL-based fitness function finds effective composite operators more quickly and the composite operators learned by smart GP algorithm have smaller size, greatly reducing both the computational expense during testing and the possibility of overfitting during training.

*Third*, a new MDL-based fitness function is proposed to improve the genetic algorithm's performance on feature selection for object detection and recognition. The MDL-based fitness function incorporates the number of features selected into the fitness evaluation process and prevents GA from selecting a large number of features to overfit the training data. The goal is to select a small set of features with good discrimination performance on both training and unseen testing data to reduce the possibility of overfitting the training data during training and the computational burden during testing.

*Fourth*, adaptive coevolutionary linear genetic programming (LGP) in conjunction with general image processing, computer vision and pattern recognition operators is proposed to synthesize recognition systems. The basic two-class approach is extended for scalability to multiple classes and various architectures and strategies are considered.

The book consists of eight chapters dealing with various evolutionary approaches for automatic synthesis and analysis of object detection and recognition systems. Many real world imagery examples are given in all the chapters and a comparison of the results with standard techniques is provided.

The book will be of interest to scientists, engineers and students working in computer vision, pattern recognition, object recognition, machine learning, evolutionary learning, image processing, knowledge discovery, data mining, cybernetics, robotics, automation and psychology.

Authors would like to thank Ken Grier, Dale Nelson, Lou Tamburino, and Bob Herklotz for their guidance and support. Many discussions held with Ed Zelnio, Tim Ross, Vince Velten, Gregory Power, Devert Wicker, Grinnell Jones, and Sohail Nadimi were very helpful.

The work covered in this book was performed at the University of California at Riverside. It was partly supported by funding from Air Force Research Laboratory during the last four years. Krzysztof Krawiec was at the University of California at Riverside on a temporary leave from Poznan University of Technology, Poznan, Poland. He would like to acknowledge the support from the Scientific Research Committee, Poland (KBN). Authors would like to thank Julie Vu and Lynne Cochran for their secretarial support.

Riverside, California November 2004 Bir Bhanu Yingqiang Lin Krzysztof Krawiec