Chapter 5

EVOLUTIONARY FEATURE SYNTHESIS FOR OBJECT RECOGNITION

5.1 Introduction

In this chapter, we investigate the effectiveness of domain knowledge in improving the efficiency of the evolutionary search and the efficacy of genetic programming in synthesizing composite features for object recognition. The basic task of object recognition is to identify the kinds of objects in an image, and sometimes the task may include estimating the pose of the recognized objects. One of the key approaches to object recognition is based on features extracted from images. These features capture the characteristics of the object and are fed into a classifier to perform recognition. The quality of object recognition is heavily dependent on the effectiveness of the features. However, it is difficult to extract good features from real images due to various factors, including noise. More importantly, there are many features that can be extracted. What are the appropriate features or how to select an appropriate set of features from the available features? If it is very difficult or even impossible to extract effective features from images, then how to synthesize useful features based on the available ones? To make use of knowledge about a specific domain and improve the quality of synthesized features, the question is how to incorporate domain knowledge in the feature synthesis? The answers

to these questions are largely dependent on the instinct, knowledge, experience, and the bias of human experts.

In this chapter, the effectiveness of coevolutionary genetic programming (CGP) [57], [69], [94] in generating composite operator vectors for object recognition is investigated. As presented in Chapter 2, genetic programming (GP) is an evolutionary computational paradigm that is an extension of genetic algorithm and works with a population of individuals. An individual in a population can be any complicated data structure such as linked lists, trees and graphs, etc. CGP is an extension of GP in which several populations are maintained and employed to evolve solutions cooperatively. A population maintained by CGP is called a sub-population and it is responsible for evolving a part of a solution. A complete solution is obtained by combining the partial solutions from all the sub-populations. In this chapter, individuals in sub-populations are composite operators, which are the elements of a composite operator vector. A composite operator is represented by a binary tree whose internal nodes are the pre-specified domain-independent primitive operators and leaf nodes are primitive features. It is a way of combining primitive features. The advantage of using a tree structure is that it is powerful enough in expressing the ways of combining primitive features and unlike a graph, it has no loops and this guarantees that the execution of individuals represented by trees terminate and not be trapped in an infinite loop. The primitive features can be directly extracted simple features or complicated features designed by human experts based on the characteristics of objects to be recognized in a particular kind of imagery (e.g., SAR images). The primitive features are real value attributes in this chapter. With each element evolved by a sub-population of CGP, a composite operator vector is cooperatively evolved by all the sub-populations. By applying composite operators, corresponding to each sub-population, to the primitive features extracted from images, composite feature vectors are obtained. These composite feature vectors are fed into a classifier for recognition. It is worth noting that the primitive operators and primitive features are decoupled from the CGP mechanism that generates composite features, so they can be tailored to particular recognition tasks without affecting the other parts of the system. Thus, the method and the recognition system are flexible and can be applied to a wide variety of images.