Chapter 6

LINEAR GENETIC PROGRAMMING FOR OBJECT RECOGNITION

6.1 Introduction

In this chapter, we describe a feature construction method which uses a special linear variety of genetic programming for feature construction. We provide rationale for the design of the method and present its two varieties: using evolutionary computation for evolutionary feature programming (EFP) and cooperative coevolution for coevolutionary feature programming (CFP). We discuss different decomposition strategies for breaking up the feature construction process. The practical utility of EFP and CFP is verified in real-case studies presented in chapter 7.

Evolutionary computation (EC) has several virtues which make it appealing from a computer vision and pattern recognition perspective. As a general template of universal search procedure, it needs relatively little task-specific tailoring to make it work within a specific application. The evolutionary search is usually characterized by low risk of being trapped in local minima, has sound rationale in both computational biology and theory (schemata theorem) [36], [42], and has proven effective in a wide spectrum of benchmarks and real-world applications. In particular, it has found a significant number of applications in image processing and analysis as discussed in the previous

chapters. In this chapter we discuss the synthesis of entire feature extraction procedures using linear genetic programming [59], [60], [61].

To make EC work as a search engine for feature construction, two important questions have to be answered: how to represent feature mappings G as solutions $\mathbf{s} \in S$, and how to evaluate individuals. This chapter gives answers to these questions and provides rationale for the proposed EFP method. However, we abstract here from any application-specific knowledge (e.g., knowledge related to computer vision). The particular examples of applying the proposed approach to specific applications will be provided in chapter 7.

6.2 Explicit Feature Construction

In most machine learning and visual learning approaches, EC operates in the space of hypotheses. An outstanding manifestation of this convention are the famous 'Michigan' [43] and 'Pittsburgh' [112] approaches for GA-based rule induction [76], [78]. In EFP, on the contrary, EC is employed to perform a search in the space of feature definitions. The evolutionary computation has been also applied to search such spaces, serving the purpose of transformation of training data. Most of the work done, however, concerned feature selection. There are several publications on applying evolutionary computation to feature selection [98], [122], [129]. A new approach was presented in Chapter 4. The superiority of global feature selection methods, EC in particular, over local search methods, was shown experimentally in early 90's [45], [46].

In the framework of learning from examples, a complete description of the learning problem is represented by the (often infinite) universe of examples (instances, objects) $\mathbf{x}$. The learning task posed to the learner (learning algorithm, inducer) consists in finding a hypothesis h (classifier) that optimizes some performance measure f defined with respect to the training data T, which is in fact a sample from the universe. In the following we assume that f is scalar and it is to be maximized, and that the learning is supervised, i.e., a discrete decision class label $d(\mathbf{x})$ is given for each training example $\mathbf{x} \in T$.