MATRIX FORMULATION AND PARALLEL ALGORITHMS FOR RELAXATION OBJECT LABELING

Eva Leung and Xiaobo Li

Department of Computing Science
University of Alberta
Edmonton, Alberta, T6G 2H1

ABSTRACT

Object labeling by relaxation is a very powerful tool in iteratively reducing local ambiguities by employing contextual information. Applications of relaxation labeling cover a diversity of different disciplines ranging from scene analysis to graph isomorphism. This paper presents a generalized matrix formulation of relaxation object labeling including the discrete, the fuzzy and the linear stochastic model, upon which two parallel algorithms on SIMD machines with different interconnection networks and different number of PEs (processing elements) are presented. The generalized model transforms the labeling problem into a matrix/vector "multiplication" problem. The "multiplication" is defined depending on specific model used. The labeling problem involves the matching of M labels to N objects. Time complexity of the algorithm is $O(N^2M^2)$ per iteration on a sequential machine. The first parallel algorithm is written on a mesh connected machine with boundary end-around using NM PEs. Time complexity of the algorithm is $O(NM)$. The second algorithm is developed on hypercube connected machine with N^2M^2 PEs. The complexity of the second algorithm is $O(\log M + \log N)$.

KEYWORDS : Vision architecture, Relaxation object labeling, SIMD computers, Parallel algorithm, Interconnection network.

I. INTRODUCTION

Relaxation labeling processes are a class of iterative algorithms which reduce local ambiguities by using contextual information. Relaxation techniques have been found to be very useful and attractive for reducing labeling errors in various types of image data and their performance is remarkably well in many domains. For instance, the relaxation labeling of simple tasks, such as labeling the sides of a triangle [7, 13], has demonstrated that perfect labeling is possible. Not only has the labeling problem been shown to play an important role in both scene analysis and computer vision [3, 7, 11], it is also a generalization of several specific problems belonging to different areas: the subgraph isomorphism problem, the graph homomorphism problem, the automata homomorphism problem, the packing problem, the shape matching problem [3].

The authors are supported in part by the Canadian National Science and Engineering Research Council under Grant A9198 and the Central Research Fund of University of Alberta.

Several algorithms [7, 8, 12] have been proposed for modifying an initial estimate of the labeling of a scene element by reference to spatial context. Many of them are actually variations of the relaxation techniques devised by Rosenfeld *et al* [8, 12]. A total of four different models are discussed in [7], namely, the discrete, the fuzzy, the linear probabilistic and the nonlinear probabilistic models. In this paper, a generalized matrix representation for various models is introduced which can readily be implemented on parallel processing machines. With sequential machine, the time complexity per iteration is $O(N^2M^2)$ which can be reduced to $O(NM)$ if parallelism is exploited.

In the next section, we use a generalized matrix form for the first three relaxation models defined in [7]. A SIMD machine model is described in section III. Section IV gives an algorithm for mesh network with boundary end-around connection, i.e., the torus network, using NM PEs. An alternative to mesh is the hypercube network which is described together with an algorithm working on N^2M^2 PEs in section V. Lastly, a conclusion is given in section VI.

II. MATRIX FORMULATION

In this section, we use a generalized matrix form for several different relaxation labeling models, mainly Rosenfeld's discrete model, fuzzy model and linear stochastic model [7]. Let $A=\{a_1, \ldots, a_N\}$ be a set of N objects and $\Lambda=\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$ be a set of M labels where M and N are integers. The objective of labeling is to assign a unit label, say λ_α, α in $\{1, 2, \ldots, M\}$, to each object a_i, i in $\{1, 2, \ldots, N\}$, in an input image. We use $a_i=\lambda_\alpha$ to represent the assignment of label λ_α to object a_i. A mapping, $l_i:\Lambda\rightarrow Range$, defines a labeling on object a_i where the range of l_i, *Range*, depends on the particular model employed. In the discrete model, *Range* is the set $\{0,1\}$, and $l_i(\lambda_\alpha)=1$ indicates that λ_α is a legal (allowed) label for object a_i, while $l_i(\lambda_\alpha)=0$ means