National Conference on
Recent Advances in Power, Signal Processing and Control
16 - 17, November, 2004
Department of Electrical Engineering, NIT, Rourkela

Parallel Genetic Algorithm based Textured Image Segmentation using Markov Random Field Model

P. K. Nanda, Sucheta Panda and P. Kanungo
Department of Electrical Engineering
National Institute of Technology
Rourkela, Orissa
India-769008

Abstract

In this paper, we address the problem of texture in image segmentation in an unsupervised frame work. Markov Random Field model is employed to model the textured images. The problem is formulated as a pixel labeling problem. The labels as well as the MRF model parameters are assumed to be unknown. A coarse grained notion based Parallel Genetic Algorithm (PGA) is proposed to estimate the pixel label together with the model parameters. With the evolution of the algorithm, the model parameters, starting from an arbitrry value, evolve to converge to the optimal estimates. The algorithm starts with arbitrary pixel labels and evolve to converge eventually to stable labels. In the proposed PGA algorithm the crossover and mutation probabilities are adaptive with the progress in generation. The algorithm is validated for synthetic as well as real images.

1 Introduction

Different texture image segmentation strategies have evolved during the last two decades[1, 2]. In the model based approach, stochastic models, in particular Markov Random Field models have been widely used to model textures[1, 2, 3, 4]. Since MRF model takes care of local characteristics and provides a link between local and global distributions, it has yielded satisfactory results in many problems. Often, the model based segmentation problem is cast as pixel labeling problem. The problem can be viewed as either supervised or unsupervised. In the supervised mode the model parameters are assumed to be known. The unsupervised frameworks assumes to have no a priori knowledge of the model parameters, the number of classes and image labels. Although various methodologies have been evolved, recently, evolutionary computation based unsupervised image segmentation schemes have been proposed[5, 6]. In [5], a relaxation algorithm has been employed to simultaneously estimate the model parameters and the image labels. The proposed scheme employs the evolutionary strategy to estimate labels and the model parameters. In this paper, we propose a Parallel Genetic Algorithm (PGA) based unsupervised scheme for image segmentation. The problem is formulated as a pixel labeling problem. The texture is modeled as a MRF model and specifically we employ Ising model as the texture model. We assume the parameters, associated with the clique potential function, number of classes and the pixel labels to be unknown.

In this paper, PGA based scheme is used to estimate the labels and the model parameters simultaneously. The proposed PGA based scheme is based on the course grain strategy [7, 8]. In this notion, the total population is divided into a number of sub-population called deme, that evolve based on the underlying notion of crowding. The coarse grained approach uses the Island model, where the migration takes place among all the demes. In our scheme, a unit is associated with pixel and each unit is characterized by a label and model parameter vector. The initial population of individuals are generated by mapping the whole image into population of elements. The population of elements is divided into four demes. In a given deme, a window is created around the pixel selected for evolvement. The parameters associated with the pixel are estimated using a fitness function which is the likelihood match of the model with the data of the window. This is based on the notion of crowding. The labels and the parameters of the unit is evolved. Thus, the algorithm is applied to each deme and after a preselected number of generations, migration takes place among the demes. Eventually, the model parameter converges and the algorithm converges to a stale labellization and hence segmentation. The algorithm started with arbitrary number of classes or labels and eventually converged to the required labels. The proposed algorithm has been validated for two class synthetic as well real textures.

2 Image Segmentation

The images assumed to be defined on a discrete rectangular lattice $S = (N \times N)$. Let X denote the random process associated with the given textured image and x be a realization of X. We model the label process Z of the image as a MRF. The noise free textured image X is assumed to be a Markov random field with respect to a neighbourhood system η, and is described by in terms of its local characteristics

$$\begin{aligned} &P(Z_{i,j} = z_{i,j} \mid Z_{k,l} = z_{k,l}, \\ &k,l \in N \times N, (k,l) \neq (i,j)) \\ &= P(Z_{i,j} = z_{i,j} \mid Z_{k,l} = z_{k,l}, (k,l) \in \eta) \end{aligned} \tag{1}$$

Here X is a MRF or equivalently Gibb's distributed (GD) which is considered as a priori distribution. This is expressed as

$$P(X = x \mid \phi) = \frac{1}{X'} e^{-U(x,\phi)} \tag{2}$$

where $X' = \sum_x e^{-U(x,\phi)}$ is the partition function, ϕ represents the clique parameter vector, the exponent term $U(x,\phi)$ is called the energy function and is of the form $U(x,\phi) = \sum_{C:(i,j)\in c} V_c(z,\phi)$, with $V_c(x,\phi)$ being referred to as the potential.

The textured image is modeled as a generalized Ising model where only the cliques that contain no more than two sites have non zero potential. Thus for example in a second order model, the number of clique types $p = 4$ and each clique i is associated with a parameter p_i and the potential in a pair clique $c = s,r \in C_i$ is $V_c(x) = \triangle_c(x)\beta_i$ where $\triangle_c(x) = -1$ if $x_s = x_r$ and $\triangle_c(x) = 1$ otherwise. Denoting $B = (\beta_1, \cdots, \beta_p)$ the vector of model parameter the energy function corresponding to a configuration x can be written as

$$U(x;B) = B.K'(x) \tag{3}$$

Where $K(x) = (k_1(x), \cdots, k_p(x))$ is defined as $K_i(x) = \sum_{c\in C_i} \triangle_c(x)$. Each site (i,j) of the input image has an associated unit $U_(i,j) = (B_(i,j), L_(i,j)$, where $B_(i,j) = \beta_{(i,j),1} \cdots, \beta_{(i,j),p}$ is a candidate vector of the texture model parameter and $L_{(i,j)}$ is a label assigned to site (i,j). A collection of $U_{(i,j)}, (i,j) \in M$ is called population. Each unit $U_(i,j)$ is assigned a fitness value $f(U_(i,j))$, which is a measure of the matching of a given unit with that of the data of window $w \times w$, $W_(i,j)$ centered around the site (i,j). The likelihood $P(X_{W_{(i,j)}} = x_{W_{(i,j)}}; B_{(i,j)})$ is a measure of this match. Thus the fitness function is defined as

$$f(U_{(i,j)}) = \frac{exp(-U(x_{W_{i,j}}; B_{(i,j)}))}{\overline{Z}_{W_{(i,j)}}(B_{(i,j)})} \tag{4}$$

where the $\overline{Z}_{W_{(i,j)}} = \sum_{y\in\Omega_W} exp(-U(x_{W_{i,j}}))$ is the approximated partition function over the window W. The approximated partition function is derived in [5] and the expression is given as follows.

$$\begin{aligned} \tilde{Z} &= \sum_{x\in\Omega} 1 - \sum_{x\in\Omega} U(x;B) + \frac{1}{2}\sum_{x\in\Omega} U(x;B)^2 \\ &= Z_0(B) - Z_1(B) + \frac{1}{2} Z_2(B) \end{aligned} \tag{5}$$

Where $Z_0 = g^n$, n is the number of sites in the window W, g is the maximum gray value from the Gray level set $G = \{0, \ldots, g-1\}$, Z_1 and Z_2 are given as

$$Z_1(B) = n(g-2)g^{n-1} \sum_{i=1}^{p} \beta_i$$

$$Z_2(B) = 4n(g-1)g^{n-2} \sum_{i=1}^{p} \beta_i^2 + n^2(g-2)^2 \{\sum_{i=1}^{p} \beta_i\}^2$$

Thus PGA is employed to determine the labels and parameters which maximizes the above likelihood function for all the units of the given textured image.

3 Parallel Genetic Algorithm

In GAs the population size is one of the parameters governing the quality of solution. As population size increases, GA has a better chance of finding the global solution. The increase in population size results in high computational burden. Hence, with serial GA one has to choose between getting a good result with a high confidence and pay a high computational cost or loosen the confidence requirement and get possibly poor result fast. In contrast, parallel GAs can keep the quality of the results high and find them fast because, using parallel machines, larger populations can be processed in less time. The Parallel Genetic Algorithms (PGAs) have been used to find solutions to many complex problems[7, 8]. The motivation behind the use of PGAs is two fold: (i)to reduce the processing time to reach the acceptable solution, (ii) to obtain better solutions in some cases in comparably sized serial GAs. GAs can be parallelized using either coarse grained approach or fine grained approach. In fine grained parallel GAs, the evaluation of individuals and the application of genetic operators are explicitly parallelized where every individual has a chance to mate with all the rest. The speed up gained is proportional to the number of processor. In case of coarse grained approach, the population is divided into a few sub-populations keeping them relatively isolated from each other. This method of parallelization introduces a migration operator and migration policy which help to send some individuals from one sub-population to another.

Two population genetic models of population structures are also in implementation of coarse grained GAs: (i) the Island Model, (ii) The stepping stone model. The population in Island model is partitioned into small sub-populations by geographic isolations and individuals can migrate to any other sub-populations as shown in Figure 1. The migration rate can be decided basing upon the problem considered. As seen from figure 1, self loops have been introduced in each deme. These loops take care of the intra deme migration that has been introduced to accelerate the rate of convergence. The rate of self migration can be lower as compared to the inter deme migration. In stepping stone model, the population is partitioned into small sub-populations but migration is restricted to neighbouring sub-populations. Our algorithm is implemented based on coarse grained approach with Island model. There are different migration policies and the solutions greatly depends upon the use of the migration policy. We have used the migration policy where good migrants replace bad individuals of a deme.

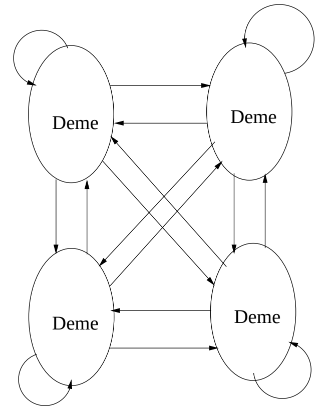

Fig.1. Interconnection model with having demes with the Intra and Inter Migration

4 Unsupervised Algorithm

The initial population of elements $U(0)$ is generated by assigning each unit $U_{(i,j)}$ a parameter vector $B_{(i,j)}$ and a label $L_{(i,j)}$ as a raster scan index of the site. Total population is divided into a few sub-populations or demes and in each deme GA based on the notion of crowding is applied. In each deme a window of size$(w \times w)$ is considered around the pixel (i, j) and the operators like crossover, mutation and selection is applied for label estimation. The operations on the edge pixels are different from that of the non-edge pixels. The tournament selection mechanism for the edge pixel is that the edge unit is compared with the immediate neighbours and an additional distant unit picked up at random from a given deme. Thus the labels and the MRF model parameters are estimated simultaneously. After a number of generations when the average fitness is above a preselected threshold a selected number of elements are migrated to the neighbouring demes. With a suitable migration policy the elements are migrated from one deme to the other. In each deme, the label and the MRF model parameters are estimated by maximizing the likelihood function defined in Section 2. The units thus evolved and converged to stable labels. These stable labels produce the necessary segmentation results.

4.1 Algorithm

1. The initial population of elements $U(0)$ is generated by assigning each unit $U_{(i,j)}$, a parameter vector $B_{(i,j)}$ sampled from an uniform

distribution over a small interval $[-\delta, \delta]$ and a label $L_{(i,j)}$ as the raster scan index of site (i, j).

2. The whole population is partitioned into a number of sub-populations called deme and to each deme the GA based crowding scheme is applied.

   (a) Crossover operation: A neighbouring unit $U_{(k,l)}$ is randomly picked in the neighbourhood $\eta_{(i,j)}$. Then one component of the parameter vector$B_{(i,j)}$ is chosen at random and is assigned the corresponding value of $B_{(k,l)}$.

   (b) Mutation operation: A random position l is chosen along the parameter vector $B_{(i,j)}$ and the corresponding parameter is added to a value m sampled from an uniform distribution.

   (c) Selection: The selection scheme is based on the notion of the tournament selection used in the crowding algorithm. The unit whose fitness is highest in the neighbourhood $\eta_{(i,j)}$ is selected to replace the unit $U_{(i,j)}$.

   (d) Steps (a), (b) and (c) are repeated for all the non-edge pixels and for the edge pixels the following selection mechanism is applied. The edge unit is compared with the immediate neighbours and an additional distant unit picked up at random for a given deme.

3. Select few individuals with high fitness to be migrated to the neighbouring demes with a selected migration policy and migration probability p_{mig}.

4. Steps 2 and 3 are repeated till the convergence is achieved that is stable labellization is achieved. The algorithm terminates when the stopping criterion is met.

   When percentage change in the labels of the image is within a threshold then the algorithm stops for stable labellization.

5 Simulation

In our simulation, we have considered a synthetic as well as real image of size (64x64) as shown in Fig 2(a) and Fig 3(a) respectively. The images are divided into 4 demes of size (16x16). The unsupervised algorithm of section 4.1 is applied to each deme. In each deme, a window size of (5x5) is considered around a pixel. The parameters used in each deme are of clique=4, maximum gray level=8, probability of migration=0.9, rate of migration=20percent, probability of crossover=0.88 and probability of mutation=0.0008. We introduce a notion of fitness threshold which is adaptive with generation. The crossover is considered if the average fitness of a deme is above the fitness threshold. This helps the algorithm to converge once the solution is localized. The mutation probability is decayed in accordance with an exponential function. After every 10 generation migration takes place among the demes. Fig 2(a) shows the original synthetic image consisting of two textures and after 10 generation the evolved labels are shown in Fig 2(b). This indicates that labels have been formed. The number of initial labels are as many as pixels in the image. There are many labels in Fig 2(b) and with progress in generation, the number of labels decreases to 4 after 20 generations as shown in Fig 2(c). The algorithm converged at 31 generations as shown in Fig 2(d). It is clear from Fig 2(d) that broadly there are two classes except two misclassified classes. The boundary between the two prominent classes are distorted. The stability is 0.94. This is the best one that we obtained by tuning the parameters, however, the results may still be improved by tuning the parameters. We have considered a real image as shown in Fig 3(a) where there are again two distinct textures. The image is divided into four demes. The parameters used were same as that of the synthetic image case except g=2 and mutation probability is P_m=0.0005. The initial labels are as many as pixels in the image and the parameter vectors were selected low value. δ is selected to be $\frac{1}{w^2}$ in both the cases, where w is the size of the window. With progress in generation labels were formed as shown in Fig 3(b). The labels and parameters evolve and after 18 generations there are four distinct regions as shown in figure 3(c). The number of labels further reduced to two broad classes after convergence as shown in Fig 3(d). Thus, the algorithm could successfully segment the two textures except the white square, which is the misclassification. The algorithm could segment broadly into two classes.

6 Conclusion

We proposed an PGA based unsupervised algorithm for textured image segmentation. The proposed scheme does not require *apriori* knowledge of the either number of textures or the associated MRF model parameters. The crossover and mutation probabilities have been adapted with generation in order to hasten the search process in the search space. The fitness function provides the likelihood match which in turn estimates the model parameters. We have employed the coarse grain approach and satisfactory results are obtained. However, attempts are made to implement the stepping stone model based PGA scheme. Since the scheme employs the Parallel Genetic Algorithm, the algorithm is more suitable from a practical standpoint. The results presented are the serial implementation of the proposed algorithm. The algorithm could properly segment the textured images with unknown number of textures. Currently, attempts are made to obtain results based on the parallel implementation of the proposed scheme.

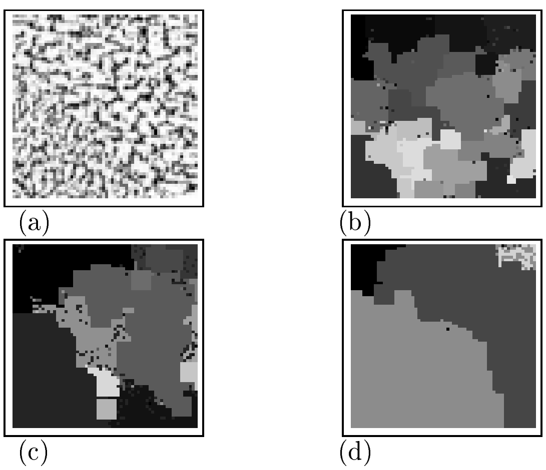

Fig.2 Unsupervised segmentation of synthetic images of size (64×64) with two textures (a) Original Image of size $(64x64)$, (b) Image after 10 generations, (c) Image after 20 generations, (d)Final segmented image after 31 generations

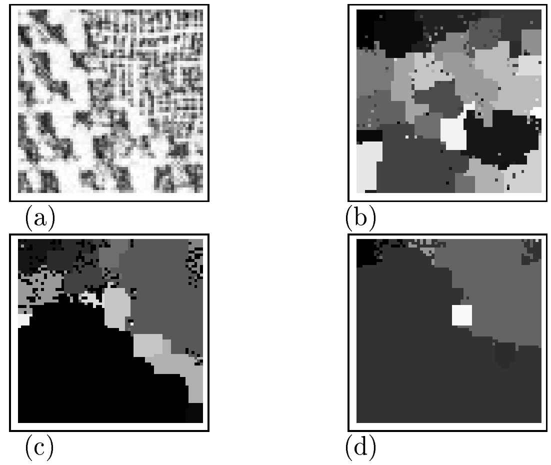

Fig. 3. Unsupervised segmentation of real textured images of size (64×64) with two textures; (a) original image, (b)image after 10 generations, (c)image after 20 generations, (d) final segmented image after 25 generations

Acknowledgment

This work is supported by the Research and Development project on Real time Signal and Image Processing using Soft Computing approach funded by Ministry of Human Resources, Government of India.

References

[1] T. R. Reed and H. J. M. du Buf, "A Review of Recent Texture Segmentation and Feature Extraction Techniques," *CVGIP: Image Understanding*, vol. 57, no.3, pp. 359-372, 1993.

[2] R. C. Dubes and A. K. Jain, "Random field models in image analysis," *J. Appl. Statistics*, vol. 16, no. 2, pp 131-163, 1989.

[3] F. S. Cohen and D. B. Cooper, "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Non causal Markovian Random Fields ," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 2, pp. 195-219, 1987.

[4] C, D'Elia, G. Poggi, G. Scarpa, "A tree-structured Markov random field model for Bayesian image segmentation," *IEEE Trans. Image processing*, vol. 12, no. 10, pp. 1259-1273, 2003.

[5] P. Andrey and P. Tarroux, "Unsupervised Segmentation of Markov Random Field Modeled Textured Images Using Selecetionist Relaxation ," *IEEE Trans. Pattern Analysis and*

Machine Intelligence, vol. 20, no. 3, pp. 252-262, 1998.

[6] Suchendra M. Bhandarkar and Hui Zhang, "Image Segmentation Using Evolutionary Computation ," *IEEE Trans. on Evolutionary Computation*, vol. 3,no. 1, pp. 100-132, 1999.

[7] Eric Cantu-Paz,"Designing Efficient and Accurate Parallel Genetic Algorithms" ,*Ph.D. Dissertation, Illinois Genetic Algorithm Laboratory, UIUC,USA*, 1999.

[8] Eric Cantu-Paz,"Migration Policies and Takeover Times in Parallel Genetic Algorithms",Technical Report, IlliGAL Report No. 99008, Jan. 199, Illinois Genetic Algorithm Lab, UIUC, USA.