Constrained Ant Colony Optimization for Data Clustering

Shu-Chuan Chu[1,3], John F. Roddick[1], Che-Jen Su[2], and Jeng-Shyang Pan[2,4]

[1] School of Informatics and Engineering,
Flinders University of South Australia,
GPO Box 2100, Adelaide 5001, South Australia
roddick@infoeng.flinders.edu.au

[2] Department of Electronic Engineering,
Kaohsiung University of Applied Sciences
Kaohsiung, Taiwan
jspan@cc.kuas.edu.tw

[3] National Kaohsiung Marine University
Kaohsiung, Taiwan

[4] Department of Automatic Test and Control,
Harbine Institute of Technology
Harbine, China

Abstract. Processes that simulate natural phenomena have successfully been applied to a number of problems for which no simple mathematical solution is known or is practicable. Such meta-heuristic algorithms include genetic algorithms, particle swarm optimization and ant colony systems and have received increasing attention in recent years.

This paper extends ant colony systems and discusses a novel data clustering process using Constrained Ant Colony Optimization (*CACO*). The *CACO* algorithm extends the Ant Colony Optimization algorithm by accommodating a quadratic distance metric, the *Sum of K Nearest Neighbor Distances* (*SKNND*) metric, constrained addition of pheromone and a shrinking range strategy to improve data clustering. We show that the *CACO* algorithm can resolve the problems of clusters with arbitrary shapes, clusters with outliers and bridges between clusters.

1 Introduction

Inspired by the food-seeking behavior of real ants, the ant system [1] and ant colony system [2] algorithms have demonstrated themselves to be efficient and effective tools for combinatorial optimization problems. In simplistic terms, in nature, a real ant wandering in its surrounding environment will leave a biological trace - pheromone - on its route. As more ants take the same route the level of this pheromone increases with the intensity of pheromone at any point biasing the path-taking decisions of subsequent ants. After a while, the shorter paths will tend to possess higher pheromone concentration and therefore encourage subsequent ants to follow them. As a result, an initially irregular path from nest to food will eventually focus to form the shortest path or paths. With

C. Zhang, H.W. Guesgen, W.K. Yeap (Eds.): PRICAI 2004, LNAI 3157, pp. 534–543, 2004.

appropriate abstractions and modifications, these natural observations have led to a successful computational model for combinatorial optimization. The ant system and ant colony system algorithms [1, 2] have been applied successfully in many difficult applications such as the quadratic assignment problem [3], data mining [4], space-planning [4], job-shop scheduling and graph coloring [5]. A parallelised ant colony system has also been developed by the authors [6, 7].

Clustering is an important technique that has been studied in various fields with applications ranging from similarity search, image compression, texture segmentation, trend analysis, pattern recognition and classification. The goal of clustering is to group sets of objects into classes such that similar objects are placed in the same class while dissimilar objects are placed in separate classes. Substantial work on clustering exists in both the statistics and database communities for different domains of data [8–18].

The Ant Colony Optimization with Different Favor (*ACODF*) algorithm [19] modified the Ant Colony Optimization (*ACO*) [2] to allow it to be used for data clustering by adding the concept of simulated annealing [20] and the strategy of tournament selection [21]. It is useful in partitioning the data sets for those with clear boundaries between classes, however, it is less suitable when faced with clusters of arbitrary shape, clusters with outliers and bridges between clusters.

An advanced version of the *ACO* algorithm, termed the Constrained Ant Colony Optimization (*CACO*) algorithm, is proposed here for data clustering by adding constraints on the calculation of pheromone strength. The proposed *CACO* algorithm has the following properties:

- It applies the quadratic metric combined with the *Sum of K Nearest Neighbor Distances* (*SKNND*) metric to be instead of the Euclidean distance measure.
- It adopts a constrained form of pheromone updating. The pheromone is only updated based on some statistical distance threshold.
- It utilises a reducing search range.

2 Constrained Ant Colony Optimization

Ant Colony Optimization with Different Favor (*ACODF*) applies *ACO* for use in data clustering. The difference between the *ACODF* and *ACO* is that each ant in *ACODF* only visits a fraction of the total clustering objects and the number of visited objects decreases with each cycle. *ACODF* also incorporates the strategies of simulated annealing and tournament selection and results in an algorithm which is effective for clusters with clearly defined boundaries. However, *ACODF* does not handle clusters with arbitrary shapes, clusters with outliers and bridges between clusters well. In order to improve the effectiveness of the clustering the following four strategies are applied:

Strategy 1: While the Euclidean distance measure is used in conventional clustering techniques such as in the *ACODF* clustering algorithm, it is not suitable for clustering non-spherical clusters, (for example, a cluster with a slender shape). In this work we therefore opt for a quadratic metric [22] as the distance measure. Given an object at position O and objects X_i, $i = 1, 2, \ldots, T$, (T is the total number of objects), the quadratic metric between the current object O and the object X_m can be expressed as

$$D_q(O, X_m) = (O - X_m)^t W^{-1} (O - X_m) \tag{1}$$

where $(O - X_m)$ is an error column vector and W is the covariance matrix given as

$$W = \frac{1}{T} \sum_{i=1}^{T} (X_i - \bar{X})(X_i - \bar{X})^t \tag{2}$$

and $\bar{X}$ is the mean of X_i, $i = 1, 2, \ldots, T$ defined as

$$\bar{X} = \frac{1}{T} \sum_{i=1}^{T} X_i \tag{3}$$

W^{-1} is the inverse of covariance matrix W.

Strategy 2: We use the *Sum of K Nearest Neighbor Distances* (*SKNND*) metric in order to distinguish dense clusters more easily. The example shown in Figure 1 shows an ant located at A which will tend to move toward C within a dense cluster rather than object B located in the sparser region. By adopting *SKNND*, as the process iterates, the probability for an ant to move towards the denser clusters increases. This strategy can avoid clustering errors due to bridges between clusters.

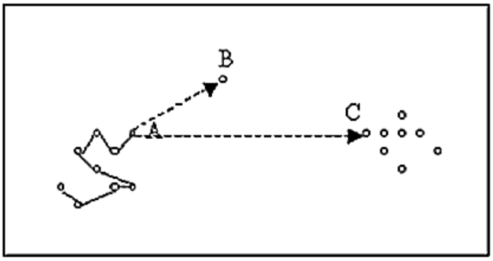

Fig. 1. Using *SKNND*, ants tend to move toward objects located within dense clusters.

Strategy 3: As shown in Figure 1, as a result of strategy 2, ants will tend to move towards denser clusters. However, the pheromone update is inversely proportional to the distance between the visited objects for conventional search formula [2] and the practical distance between objects A and C could be farther than that between objects A and B reducing the pheromone level and causing a clustering error. In order to compensate for this, a statistical threshold for the k^{th} ant is adopted as below.

$$L^k_{ts} = AvgL^k_{path} + StDevL^k_{path} \tag{4}$$

where $AvgL^k_{path}$ and $StDevL^k_{path}$ are the average of the distance and the standard deviation for the route of the visited objects by the k^{th} ant expressed as

$$AvgL^k_{path} = \frac{\sum L^k_{ij}}{E}, \quad if\ (X_i, X_j)\ path\ visited\ by\ the\ k^{th}\ ant \tag{5}$$

$$StDevL^k_{path} = \sqrt{\frac{\sum (L^k_{ij} - AvgL^k_{path})^2}{E}}, \tag{6}$$

$$if\ (X_i, X_j)\ path\ visited\ by\ the\ k^{th}\ ant$$

where E is the number of paths visited by the k^{th} ant. We may roughly consider objects X_i and X_j to be located in different clusters if $L^k_{ij} > L^k_{ts}$. The distance between objects X_i and X_j cannot be added into the length of the path and the pheromone cannot be updated between the objects.

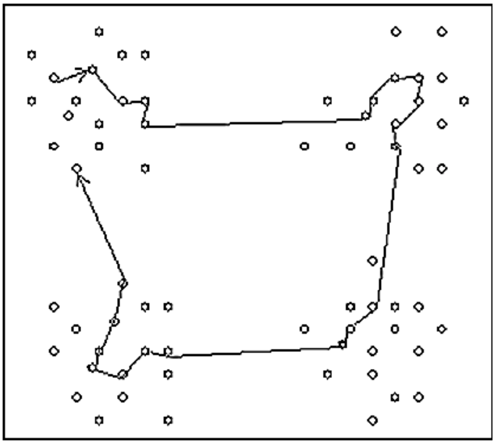

Fig. 2. Conventional search route.

Strategy 4: The conventional search formula [2] between objects r and s is not suitable for robust clustering as object s represents all un-visited objects resulting in excessive computation and a tendency for ants to jump between dense clusters as shown in Figure 2. In order to improve clustering speed and eliminate this jumping phenomenon, the conventional search formula [2] is modified to be

$$P_k(r,s) = \begin{cases} \frac{[\tau(r,s)]\cdot[D_q(r,s)]^{-\beta}\cdot[SKNND(s)]^{-\gamma}}{\sum_{u \in J_k^{N_2}(r)} [\tau(r,u)]\cdot[D_q(r,u)]^{-\beta}\cdot[SKNND(u)]^{-\gamma}} & , \quad if\ s \in J_k^{N_2}(r) \\ 0 & , \quad otherwise \end{cases} \tag{7}$$

where $J_k^{N_2}(r)$ is used to shrink the search range to the N_2 nearest un-visited objects. N_2 is set to be some fraction of the object (in our experiments we used 10%), $D_q(r,s)$ is the quadratic distance between objects r and s. $SKNND(s)$ is the sum of the distances between object s and the N_2 nearest objects. β and γ are two parameters which determine the relative importance of pheromone level versus the quadratic distance and the Sum of N_2 Nearest Neighbor Distance, respectively. We have found that setting β to 2 and γ to between 5 and 15 results in robust performance. As shown in Figure 3, the jumping phenomenon is eliminated after using the shrinking search formula.

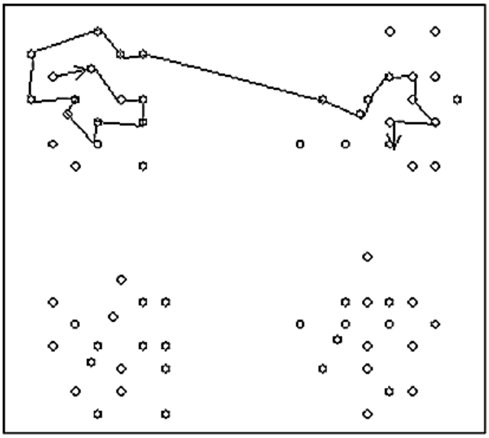

Fig. 3. Shrinking search route using Eq. (7).

The Constrained Ant Colony Optimization algorithm for data clustering can be expressed as follows:

Step 1: Initialization
Randomly select the initial object for each ant. The initial pheromone τ_{ij} between any two objects X_i and X_j is set to be a small positive constant τ_0.

Step 2: Movement
Let each ant moves to N_1 objects only using Eq. (7). In our initial experiments, N_1 was set to be 1/20 of the data objects.

Step 3: Pheromone Update
Update the pheromone level between objects as

$$\tau_{ij}(t+1) = (1-\alpha)\tau_{ij}(t) + \Delta\tau_{ij}(t+1) \tag{8}$$

$$\Delta\tau_{ij}(t+1) = \sum_{k=1}^{T} \Delta\tau_{ij}^{k}(t+1) \tag{9}$$

$$\Delta\tau_{ij}^{k}(t+1) = \begin{cases} \frac{Q}{L_k} & , \ if\ ((i,j) \in route\ done\ by\ ant\ k,\ and\ L_{ij}^{k} < L_{ts}^{k} \\ 0 & , \ otherwise \end{cases} \tag{10}$$

where τ_{ij} is the pheromone level between objects X_i and X_j, T is the total number of clustering objects, α is a pheromone decay parameter and Q is a constant and is set to 1. L_k is the length of the route after deleting the distance between object X_i and object X_j in which $L_{ij}^k > L_{ts}^k$ for the k^{th} ant.

Step 4: Consolidation

Calculate the average pheromone level on the route for all objects as

$$Avg\tau = \frac{\sum_{i,j \in E} \tau_{ij}}{E} \tag{11}$$

where E is the number of paths visited by the k^{th} ant. Disconnect the path between two objects if the pheromone level between these two objects is smaller than $Avg\tau$. All the objects thus connected together are deemed to be in the same cluster.

3 Experiments and Results

The experiments were carried out to test the performance of the data clustering for Ant Colony Optimization with Different Favor (*ACODF*), *DBSCAN* [14], *CURE* [11] and the proposed Constrained Ant Colony Optimization (*CACO*). Four data sets, Four-Cluster, Four-Bridge, Smile-Face and Shape-Outliers were used as the test material, consisting of 892, 981, 877 and 999 objects, respectively.

In order to cluster a data set using *CACO*, N_1 and γ are two important parameters which will influence the clustering results. N_1 is the number of objects to be visited in each cycle for each ant. If N_1 is set too small, the ants cannot finish visiting all the objects belonged to the same cluster resulting in a division of slender shaped cluster into several sub-clusters. Our experiments indicated that good experimental results were obtained by setting N_1 to $\frac{1}{20}$. γ also influences the clustering result for clusters with bridges or high numbers of outliers. We found that γ set between 5 and 15 provided robust results. The number of ants is set to 40.

DBSCAN is a well-known clustering algorithm that works well for clusters with arbitrary shapes. Following the recommendation of *Ester et al.*, *MinPts* was fixed to 4 and ϵ was changed during the experiments. *CURE* produces high-quality clusters in the existence of outliers, allowing complex shaped clusters and different size. We performed experiments with shrinking factor is 0.3 and the number of representative points as 10, which are the default values recommended by *Guha et al.* (1998).

All the experiments demonstrate *CACO* algorithm can correctly identifies the clusters. For the reason of saving the space, we only describe the last experiment to partition the shape-outliers data set. *ACODF* algorithm cannot correctly partition the Shape-Outliers data set shown in Figure 4. Figure 5 shows the clusters found by *DBSCAN*, but it also makes a mistake in that it has fragmented the clusters in the right-side 'L'-shaped cluster. Figure 6 shows that

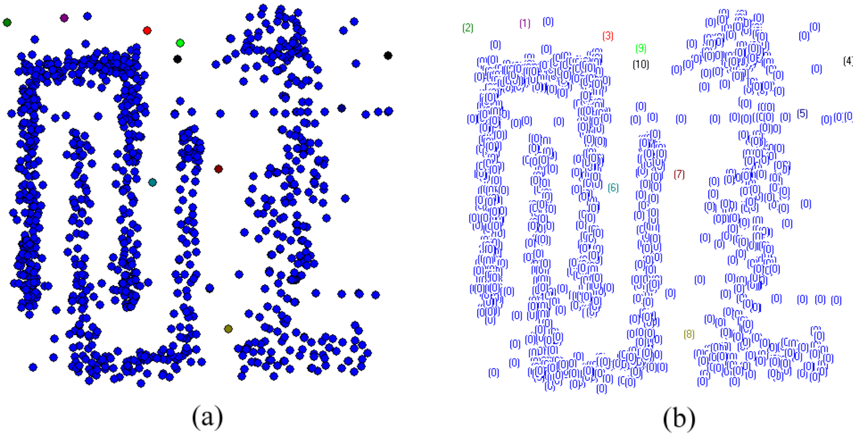

Fig. 4. Clustering results of Shape-Outliers by *ACODF* algorithm. (a) cluster represented by colour, (b) cluster represented by number.

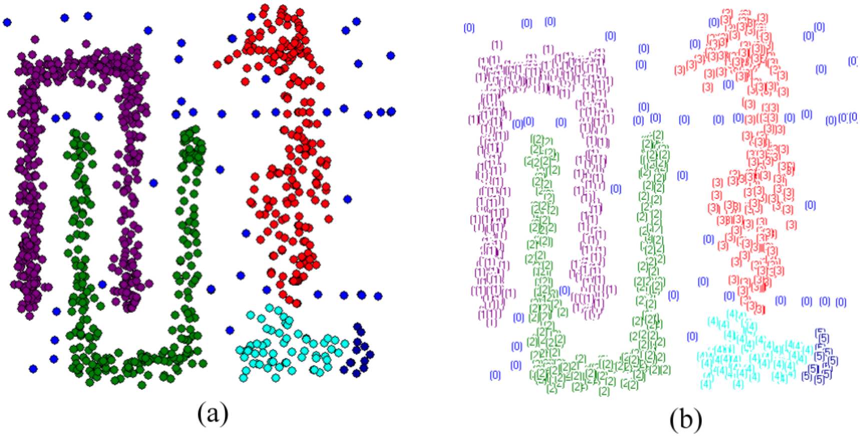

Fig. 5. Clustering results of Shape-Outliers by *DBSCAN* algorithm. (a) cluster represented by colour, (b) cluster represented by number.

CURE fails to perform well on Shape-Outliers data set, with the clusters fragmented into a number of smaller clusters. Looking at Figure 7, we can see that *CACO* algorithm correctly identifies the clusters.

4 Conclusions

In this paper, a new Ant Colony Optimization based algorithm, termed Constrained Ant Colony Optimization (*CACO*), is proposed for data clustering. *CACO* extends Ant Colony Optimization through the use of a quadratic metric, the *Sum of K Nearest Neighbor Distances* metric, together with constrained addition of pheromone and shrinking range strategies to better partition data sets

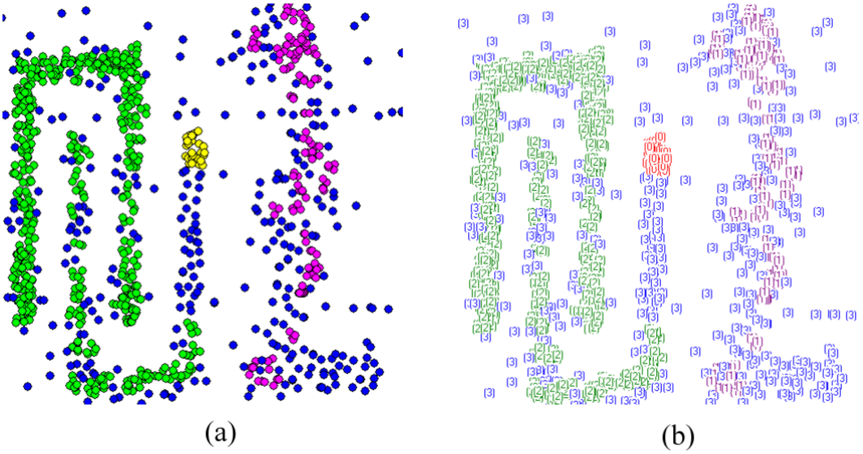

Fig. 6. Clustering results of Shape-Outliers by *CURE* algorithm. (a) cluster represented by colour, (b) cluster represented by number.

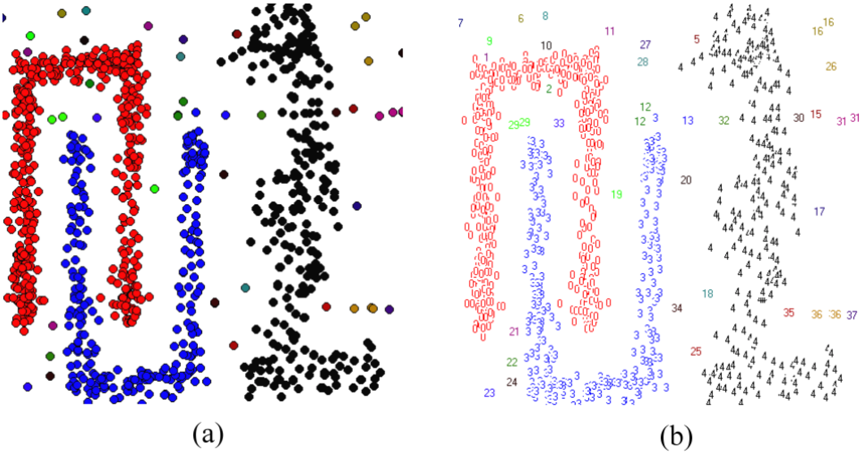

Fig. 7. Clustering results of Shape-Outliers by *CACO* algorithm. (a) cluster represented by colour, (b) cluster represented by number.

with clusters with arbitrary shape, clusters with outliers and outlier points connecting clusters. Preliminary experimental results compared with the *ACODF*, *DBSCAN* and *CURE* algorithms, demonstrate the usefulness of the proposed *CACO* algorithm.

References

1. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Trans. on Systems, Man, and Cybernetics-Part B: Cybernetics 26 (1996) 29–41

2. Dorigo, J.M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. on Evolutionary Computation 1 (1997) 53–66
3. Maniezzo, V., Colorni, A.: The ant system applied to the quadratic assignment problem. IEEE Trans. on Knowledge and Data Engineering 11 (1999) 769–778
4. Parpinelli, R.S., Lopes, H.S., Freitas, A.A.: Data mining with an ant colony optimization algorithm. IEEE Trans. on Evolutionary Computation 6 (2002) 321–332
5. Bland, J.A.: Space-planning by ant colony optimization. International Journal of Computer Applications in Technology 12 (1999) 320–328
6. Chu, S.C., Roddick, J.F., Pan, J.S., Su, C.J.: Parallel ant colony systems. In Zhong, N., Raś, Z.W., Tsumoto, S., Suzuki, E., eds.: 14th International Symposium on Methodologies for Intelligent Systems. Volume 2871., Maebashi City, Japan, LNCS, Springer-Verlag (2003) 279–284
7. Chu, S.C., Roddick, J.F., Pan, J.S.: Ant colony system with communication strategies. Information Sciences (2004) (to appear)
8. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: 5th Berkeley symposium on mathematics, statistics and Probability. Volume 1. (1967) 281–296
9. Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis. John Wiley and Sons, New York (1990)
10. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: An efficient clustering method for very large databases. In: ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, Montreal, Canada (1996) 103–114
11. Guha, S., Rastogi, R., Shim, K.: CURE: an efficient clustering algorithm for large databases. In: ACM SIGMOD International Conference on the Management of Data, Seattle, WA, USA (1998) 73–84
12. Karypis, G., Han, E.H., Kumar, V.: CHAMELEON: a hierarchical clustering algorithm using dynamic modeling. Computer 32 (1999) 32–68
13. Ganti, V., Gehrke, J., Ramakrishnan, R.: CACTUS – clustering categorical data using summaries. In Chaudhuri, S., Madigan, D., eds.: Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, ACM Press (1999) 73–83
14. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In Simoudis, E., Han, J., Fayyad, U., eds.: Second International Conference on Knowledge Discovery and Data Mining, Portland, Oregon, AAAI Press (1996) 226–231
15. Sheikholeslami, G., Chatterjee, S., Zhang, A.: WaveCluster: A multiresolution clustering approach for very large spatial databases. In: 1998 International Conference Very Large Data Bases (VLDB'98), New York (1998) 428–439
16. C, A.C., S, Y.P.: Redefining clustering for high-dimensional applications. IEEE Trans. on Knowledge and Data Engineering 14 (2002) 210–225
17. Estivill-Castro, V., Lee, I.: AUTOCLUST+: Automatic clustering of point-data sets in the presence of obstacles. In Roddick, J.F., Hornsby, K., eds.: International Workshop on Temporal, Spatial and Spatio-Temporal Data Mining, TSDM2000. Volume 2007., Lyon, France, LNCS, Springer-Verlag (2000) 133–146
18. Ng, R.T., Han, J.: Clarans: A method for clustering objects for spatical data mining. IEEE Transactions on Knowledge and Data Engineering 14 (2002) 1003–1016
19. Tsai, C.F., Wu, H.C., Tsai, C.W.: A new data clustering approach for data mining in large databases. In: International Symposium on Parallel Architectures, Algorithms and Networks, IEEE Press (2002) 278–283

20. Kirkpatrick, S., Gelatt, J.C.D., Vecchi, M.P.: Optimization by simulated annealing. Science 220 (1983) 671–680
21. 21, A.: Genetic algorithms for function optimization. PhD thesis, University of Alberta, Edmonton, Canada (1981)
22. Pan, J.S., McInnes, F.R., Jack, M.A.: Bound for minkowski metric or quadratic metric applied to VQ codeword search. IEE Proc. Vision Image and Signal Processing 143 (1996) 67–71