Pergamon

Pattern Recognition, Vol. 30, No. 2, pp. 261–272, 1997
Copyright © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd
Printed in Great Britain. All rights reserved
0031-3203/97 $17.00+.00

PII:S-0031-3203(96)00079-9

PARAMETRIC AND NON-PARAMETRIC UNSUPERVISED CLUSTER ANALYSIS

STEPHEN J. ROBERTS*
Department of Electrical and Electronic Engineering, Imperial College of Science, Technology and Medicine,
University of London, London SW7 2BT, U.K.

(*Received* 12 *October* 1995; *in revised form* 10 *May* 1996; *received for publication* 28 *May* 1996)

Abstract—Much work has been published on methods for assessing the probable number of clusters or structures within unknown data sets. This paper aims to look in more detail at two methods, a broad parametric method, based around the assumption of Gaussian clusters and the other a non-parametric method which utilises methods of scale-space filtering to extract robust structures within a data set. It is shown that, whilst both methods are capable of determining cluster validity for data sets in which clusters tend towards a multivariate Gaussian distribution, the parametric method inevitably fails for clusters which have a non-Gaussian structure whilst the scale-space method is more robust. Copyright © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Cluster analysis   Maximum likelihood methods   Scale-space filtering
Probability density estimation

1. INTRODUCTION

Most scientific disciplines generate experimental data from an observed system about which we may have little understanding of the data generating function. The notion that complexity may be (at least partially) analysed by breaking it down into simpler substructures is deeply rooted in (western) scientific culture. For whatever reason, however, it has been commented that there appear to be more algorithms for clustering data than data to analyse! I do not dispute this—indeed the fact that data structure is multifarious will inevitably prompt the development of a large number of clustering approaches. Commenting on a survey of clustering techniques, Jain(1) states that (up to 1982) some 40 books alone had been published on the subject. It is important that this breadth of approach is acknowledged, as the problem is complex and many methods must often be tried from the analysis "toolbox" before data structure may be inferred. Excellent reviews of many methods may be found in, for example, Jain,(1) Jain and Dubes,(2) Hartigan(3) and Everitt.(4)

On a broad, descriptive level cluster analysis algorithms can be broken into two distinct phases.[1] Firstly, a model fitting phase, whereby some *partition hypothesis* of complexity K, $\mathscr{H}_K$ say, is "optimally" fitted to the data set. Secondly, a model validation phase, whereby the set of $\{\mathscr{H}_K\}$ are assessed according to some cluster validity criterion and the "optimal" partition hypothesis selected.

Jain(1) makes the distinction between *hierarchical* and *partitional* fitting methods. For the most part, the former act by partitioning the data set into successively fewer structures, based upon merging structures which have sufficient similarity. Such a method gives rise to a *dendogram* which, amongst other properties, details the number of structures obtained as a function of some merging threshold. With a threshold of zero the number of structures equals the number of data and a high threshold partitions the entire data set as a single cluster. Observation of the dendogram linkage may then be used to select an appropriate number of structures within the data (this is often based upon a partition's "lifetime"—the range over which a partitioning is stable with respect to changes in the merging threshold). The major drawback of the hierarchical approach is that the entire dendogram is sensitive to (possibly erroneous) previous cluster merging, i.e. data are not permitted to change cluster membership once assignment has taken place.

Partitional methods typically start with a data partitioning into a small number of clusters and increase the number of partitions into which the data is divided. The precise partitioning is performed so as to minimize or maximize some objective function. A datum is, furthermore, free to change its partition membership in such a scheme. The precise choice of objective function clearly has a very strong bearing on the partitioning of the data; indeed the most popular, the total square error between $\mathscr{H}_K$ and the data, implies that the data will be modelled by the fitting of hyper-ellipsoidal clusters, and whether we may assume that clusters (if they exist) are multivariate Gaussian distributed is an open question for many data sets. Many methods for ranking the validity

* E-mail: s.j.roberts@ic.ac.uk.

[1] One could argue that, in the case of Bayesian inference methods, these two phases are so interelated as to form a single methodology, however.

of data models of differing complexity (number of partitions) have been proposed. Most, however, rely (implicitly or explicitly) upon estimates of within- and between-cluster scatter matrices.[1,2,5] The major problem with this approach, of course, is that if data do not conform to the assumptions made by the technique then the latter may impose structure on the data and not disclose the "true" structure.[1] This is also the case when clusters have widely differing numbers of members. The objective function may be improved by artificially splitting a cluster with a large number of members more than recognising one with a small number.[1] The use of "fuzzy" cluster methods (whereby a datum's membership may be distributed over many clusters)[6] may be incorporated into the genre of partitional methods with relative ease and has proved popular.[7–12] Such an extension to the modelling process is certainly more representative of "real" data. There are still, however, difficulties encountered. The "optimal" number of clusters must be estimated, the location and shape of the clusters is invariably unknown and there may be a large variability in the number of data members in each cluster.[7]

This paper looks in detail at two methods of *unsupervised* clustering (the "optimal" number of partitions is unknown *a priori*). The first is a partitional method utilising a maximum likelihood algorithm; this is reduced to the well-known *K*-means algorithm as well. The second method may be seen as falling within the hierarchical clustering genre or as a method of scale-space (multiresolution) parameter estimation.[13] Results from both methods are compared on test data and the scale-space method on examples from image and signal processing.

2. MAXIMUM LIKELIHOOD AND *K*-MEANS ALGORITHMS

2.1. *Theory*

We consider the case of a data set, $\chi=\{\boldsymbol{x}_i\}$ where $\chi\subset\Re^d$. Let the distribution of data in χ form a probability density function denoted by $p(\chi)$. We may estimate this density function, in the limit, by considering it as a weighted combination of all possible data models, $\mathscr{M}$, each one of which is specified by a parameter vector $\boldsymbol{\theta}^{\mathscr{M}}$

$$p(\chi) = \int\limits_{\mathscr{M}} \int\limits_{\boldsymbol{\theta}^{\mathscr{M}}} p(\chi|\mathscr{M},\boldsymbol{\theta}^{\mathscr{M}})p(\mathscr{M},\boldsymbol{\theta}^{\mathscr{M}})\,\mathrm{d}\boldsymbol{\theta}^{\mathscr{M}}\mathrm{d}\mathscr{M}. \tag{1}$$

Such a complex specification is normally constrained such that we look only at a particular class of models, such as those with the property of universal approximation, the Gaussian mixture model (GMM) being a popular choice. We may then specify each GMM by a single parameter, *K*, which describes the complexity of the model (the number of Gaussian kernels). If, furthermore, we assume, as is common practise, that the probability distribution of the within-model free parameters, specified by $\boldsymbol{\theta}^K$, is dominated by a single, most probable, solution, $\boldsymbol{\theta}^K_{\mathrm{MP}}$, then equation (1) reduces to

$$p(\chi) = \sum_K p(\chi|K,\boldsymbol{\theta}^K_{MP})p(K,\boldsymbol{\theta}^K_{MP}). \tag{2}$$

For ease of notation we assume that, for every model specified by *K*, dependence upon $\boldsymbol{\theta}^K_{\mathrm{MN}}$ is implied. Bayes' theorem then states that (dropping the $\boldsymbol{\theta}$ terms)

$$p(K|\chi) = \frac{p(\chi|K)p(K)}{p(\chi)}, \tag{3}$$

where the evidence term, $p(\chi)$, is given by equation (2). As the number of Gaussian kernels, *K*, specifies the number of data partitions, we may use $p(K|\chi)$ as a partition validation measure.

2.2. *Parameter estimation*

The likelihood term in equation (3), if we assume *N* independent data samples, may be written as

$$p(\chi|K) = \prod_{i=1}^{N} p(\boldsymbol{x}_i|K). \tag{4}$$

From Bayes' theorem for mixtures we may write the above as

$$p(\chi|K) = \prod_{i=1}^{N}\left[\sum_{k=1}^{K} p(\boldsymbol{x}_i|k)p(k)\right], \tag{5}$$

where $p(k)$ is the *a priori* probability of the *k*th kernel in the GMM. The within-model parameter vector, $\boldsymbol{\theta}^K_{\mathrm{MN}}$, however, still remains to be estimated for each model. The maximum likelihood approach seeks to maximise the logarithm of equation (4), namely

$$\sum_{i=1}^{N} \log p(\boldsymbol{x}_i|K). \tag{6}$$

This maximisation is subject to the constraint that

$$\sum_{k=1}^{K} p(k) = 1. \tag{7}$$

If we let the free parameters of each component, *k*, of the GMM be specified by a parameter vector $\boldsymbol{\theta}_k$ then combining equations (5) and (6) we obtain, for the *k*th component:[14,15]

$$\sum_{i=1}^{N} p(k|\boldsymbol{x}_i)\frac{\partial}{\partial\theta_k}\log p(\boldsymbol{x}_i|k,\theta_k) = 0. \tag{8}$$

For a GMM, each kernel is a multivariate Gaussian whose free parameters are completely specified by its mean, $\boldsymbol{\mu}_k$, and covariance matrix, $\boldsymbol{\Sigma}_k$. Hence

$$\begin{aligned} p(\boldsymbol{x}_i|k,\boldsymbol{\theta}_k) &= p(\boldsymbol{x}_i|k,\boldsymbol{\mu}_k,{}_k) \\ &= \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_k|^{1/2}} \\ &\quad\times \exp\left[-\frac{1}{2}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)^{\mathrm{T}}\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)\right]. \end{aligned} \tag{9}$$

Combining equations (7–12) gives[14,15]

$$\hat{p}(k) = \frac{1}{N}\sum_{i=1}^{N} \hat{p}(k|\boldsymbol{x}_i),$$
$$\hat{\boldsymbol{\mu}}_k = \frac{\sum_{i=1}^{N} \hat{p}(k|\boldsymbol{x}_i)\boldsymbol{x}_i}{\sum_{i=1}^{N} \hat{p}(k|\boldsymbol{x}_i)},$$
$$\hat{\boldsymbol{\Sigma}}_k = \frac{\sum_{i=1}^{N} \hat{p}(k|\boldsymbol{x}_i)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k)^{\mathrm{T}}}{\sum_{i=1}^{N} \hat{p}(k|\boldsymbol{x}_i)}. \quad (10)$$

Solution to equation (10) requires a non-linear optimisation algorithm. Solutions may be estimated using batch updating algorithms[7,11] or by means of iterative variants of the Expectation-Maximisation (EM) algorithm,[16] full details of which may be found in references (14,15) for example (see also Section 2.4). If we allow the set of posterior probabilities, $\hat{p}(k|\boldsymbol{x}_i)$, to collapse to the set $p'(k|\boldsymbol{x}_i)$ such that

$$p'(k|\boldsymbol{x}_i) = \begin{cases} 1 & \text{if } k = \arg\max_k\{\hat{p}(k|\boldsymbol{x}_i)\}, \\ 0 & \text{otherwise,} \end{cases} \quad (11)$$

then the above maximum-likelihood solution becomes a standard K-means approach. Solutions to the latter, like the ML approach, may be obtained using batch or incremental algorithms.

2.3. *Cluster validity*

The use of either the K-means or the ML approach imposes the implicit assumption of hyper-ellipsoidal clusters on the data model. Most cluster validity measures are based upon estimates of the kernel covariance matrices for a given model complexity (number of kernels in the GMM).[1] This paper follows a proposal made in reference (7) to utilise the "fuzzy hypervolume", V, of the data partitioning. For a GMM with K components this is given as

$$V_K = \sum_{k=1}^{K} |\boldsymbol{\Sigma}_k|^{1/2}. \quad (12)$$

Gath and Geva[7] choose the partition model which has minimum V_K. We will, however, allow V_K to act as a penalty term, such that those data models with large values of V_K have correspondingly low prior probabilities. By setting the prior as

$$p(K) = \frac{\nu}{V_K}, \quad (13)$$

(where ν is a normalising factor to ensure that $\Sigma\, p(K)$=1) and combining equations (3) and (13), we obtain

$$p(K|\chi) = \frac{\nu p(\chi|K)}{p(\chi)V_K}. \quad (14)$$

As both the evidence term, $p(\chi)$, and ν are constant over all models, ranking models according to their posterior on the data set is equivalent to ranking on a "likelihood density" term, $\rho(K)$ given by

$$\rho(K) = \frac{p(\chi|K)}{V_K}. \quad (15)$$

We note that the above formalism may be applied to both the ML and K-means partitioning methods.

2.4. *Notes on implementation*

All data sets in the paper were normalised to zero mean and unit variance. This is especially important in the case of the K-means algorithm, which relies upon the unweighted L_2 norm.

Solutions to both the ML and K-means partitionings were obtained using iterative (stochastic gradient-descent) algorithms. For the ML method, the iterative solutions to equation (10) are given as[15]

$$\hat{p}_k^{(t+1)} = \hat{p}_k^{(t)} + \alpha^{(t)}(p(k|\boldsymbol{x}^{(t)}) - \hat{p}_k^{(t)}),$$
$$\hat{\boldsymbol{\mu}}_k^{(t+1)} = \frac{\hat{\boldsymbol{\mu}}_k^{(t)} + \alpha^{(t)}[p(k|\boldsymbol{x}^{(t)})\boldsymbol{x}^{(t)} - \hat{\boldsymbol{\mu}}_k^{(t)}]}{(1-\alpha^{(t)}) + \alpha^{(t)}p(k|\boldsymbol{x}^{(t)})},$$
$$\hat{\boldsymbol{\Sigma}}_k^{(t+1)} = \frac{\hat{\boldsymbol{\Sigma}}_k^{(t)} + \alpha^{(t)}[p(k|\boldsymbol{x}^{(t)})(\boldsymbol{x}^{(t)} - \hat{\boldsymbol{\mu}}_k^{(t)})(\boldsymbol{x}^{(t)} - \hat{\boldsymbol{\mu}}_k^{(t)})^{\mathrm{T}} - \hat{\boldsymbol{\Sigma}}_k^{(t)}]}{(1-\alpha^{(t)}) + \alpha^{(t)}p(k|\boldsymbol{x}^{(t)})}, \quad (16)$$

where $\boldsymbol{x}^{(t)}$ is the tth randomly-chosen member of χ to be presented to the algorithm and $\alpha^{(t)}$ is a decreasing adaption parameter chosen to be of the form

$$\alpha^{(t)} = \frac{\alpha^{(t)}}{1+t} \quad (17)$$

to satisfy the Munro-Robbins convergence equations.[15]

For the K-means algorithm, the above equation for the adaption of the mean of the kth kernel reduces to

$$\boldsymbol{\mu}_k^{(t+1)} = \begin{cases} \boldsymbol{\mu}_k^{(t)} + \alpha^{(t)}(\boldsymbol{x}^{(t)} - \boldsymbol{b}\mu_k^{(t)}) & \text{if } \boldsymbol{\mu}_k^{(t)} \text{ has smallest } L_2 \text{ norm to } \boldsymbol{x}^{(t)}, \\ \boldsymbol{\mu}_k^{(t)} & \text{otherwise.} \end{cases} \quad (18)$$

It is noted that, in many applications of the K-means algorithm, $\alpha^{(t)}$ is chosen to be a constant adaption gain $0 \le \alpha \le 1$. Whilst this means that the estimates of the kernel means will be *inconsistent* (their variance with t does not go to zero as $t \rightarrow \infty$) they do converge, however, *in the mean*, to the expected values, and are hence unbiased estimators. The exact choice of α should be made from knowledge of the eigen spectrum of the data covariance matrix, but for most practical applications a value of α=0.01 gives good results (assuming a unit variance data set).

It is computationally easier to implement diagonal covariance matrices in both the ML and K-means approaches. This, however, constrains the model even further. In the ML approach the covariance matrices are updated at each sample. We may, however, ease the computational burden of constantly inverting matrices by utilising the Woodbury identity. For the K-means algorithm, once the means, $\boldsymbol{\mu}_k$ have been calculated, we may estimate both $\boldsymbol{\Sigma}_k$ and $p(k)$ from equation (10) and equation (11).

3. SCALE-SPACE CLUSTERING

The major problem with the above approaches is that cluster validity makes a strong assumption about the *parametric* form of the data. For many applications this is undesirable. Furthermore, solutions to local centroid (mean) estimation are not robust in the presence of noise. Spann and Wilson[13] consider a paradigm shift in the evaluation of the clustering problem in which they require estimates of valid structure within a data set to be robust both to the presence of noise and to spatial scale changes of the data. We expand upon this methodology in this paper.

3.1. Theory

We consider once more a data set, χ, and make a non-parametric estimate of the probability density function of $\boldsymbol{x} \in \chi$ as the weighted combination of a set of basis functions (kernels), f, sited at each $\boldsymbol{x}_i \in \chi$.

$$\hat{p}(\boldsymbol{x}) = \sum_{i=1}^{N} w^{(i)} f^{(i)}(\boldsymbol{x}), \tag{19}$$

where the superscript (i) implies the basis function is sited at $\boldsymbol{x}_i$. If we take the Parzen-windows approach,[17] whereby the weighting of each basis function is independent of its position we may write equation (19) as

$$\hat{p}(\boldsymbol{x}) = w \sum_{i=1}^{N} f^{(i)}(\boldsymbol{x}), \tag{20}$$

whence w becomes a normalising factor ensuring that $\hat{p}(\boldsymbol{x})$ integrates to unity. In the case of noiseless data, data clusters may be defined as peaks in the probability density function and hence the number of clusters, and their centroids, may be evaluated from the peaks of equation (20) or the positive–negative zero-crossings of its spatial derivative.

In the more realistic case, where noise exists in the data set, this peak detection fails to detect genuine data structure as the noise generates false peaks in $\hat{p}(\boldsymbol{x})$. Equation (20) then becomes

$$\hat{p}(\boldsymbol{x}) = w \sum_{i=1}^{N} f^{(i)} + \zeta(\boldsymbol{x}), \tag{21}$$

where $\zeta(\boldsymbol{x})$ is a noise process. Estimating the positions of the extrema of $\hat{p}(\boldsymbol{x})$ is equivalent to finding the set of zero-crossings of $\partial \hat{p}(\boldsymbol{x})/\partial \boldsymbol{x}$. If we wish to provide some noise reduction to the estimates, however, some form of spatial smoothing must be performed. If the smoothing function is specified in the spatial domain by a filter kernel $\kappa_s(\boldsymbol{x})$ where s is a scale parameter, then, as the convolution and differentiation operators commute, we may rewrite the problem as one of seeking the zero-crossings of

$$\begin{aligned}\hat{p}(\boldsymbol{x}) * \frac{\partial}{\partial \boldsymbol{x}} \kappa_s(\boldsymbol{x}) &= \hat{p}(\boldsymbol{x}) * h_s(\boldsymbol{x}) \\ &= \left\{ w \sum_{i=1}^{N} f^{(i)}(\boldsymbol{x}) + \zeta(\boldsymbol{x}) \right\} * h_s(\boldsymbol{x}).\end{aligned} \tag{22}$$

It may be shown[13] that the "optimal" (Wiener) filter, $h_s(\boldsymbol{x})$, for the problem, under the assumption that $\zeta(\boldsymbol{x})$ is white, is the derivative of a symmetric low-pass filter kernel. A full proof is given in reference (13).

Furthermore, we require the zero-crossings of equation (22) to correspond to a maximum likelihood solution of a local cluster centroid. Operating $\kappa_s(\boldsymbol{x})$ on equation (20) we obtain

$$\hat{p}_s(\boldsymbol{x}) = w \sum_{i=1}^{N} f^{(i)}(\boldsymbol{x}) * \kappa_s(\boldsymbol{x}). \tag{23}$$

Letting $f^{(i)}(\boldsymbol{x}) * \kappa_s(\boldsymbol{x}) = \phi_s^{(i)}(\boldsymbol{x})$, the above is written as

$$\hat{p}_s(\boldsymbol{x}) = w \sum_{i=1}^{N} \phi_s^{(i)}(\boldsymbol{x}) \tag{24}$$

which may be regarded as an estimate of $p(\boldsymbol{x})$, smoothed at scale s, based upon a new set of basis functions, $\phi_s^{(i)}(\boldsymbol{x})$.

We now consider the form of a local maximum-likelihood solution to the position of the kth cluster centroid, given by equation (10) and rewritten as

$$\sum_{i=1}^{N} (\boldsymbol{x}_i - \boldsymbol{\mu}_k) \hat{p}(k|\boldsymbol{x}_i) = 0. \tag{25}$$

Taking the spatial derivative of equation (24) and setting it equal to zero gives

$$w \sum_{i=1}^{N} \frac{\partial}{\partial \boldsymbol{x}} \phi_s^{(i)}(\boldsymbol{x}) = 0. \tag{26}$$

The solutions to this equation occur at a series of points and we consider the kth maxima solution, $\boldsymbol{x} = \boldsymbol{\mu}_k$, say. equations (25) and (26) may be equated if we allow

$$w \phi_s^{(i)}(\boldsymbol{\mu}_k) = \hat{p}(k|\boldsymbol{x}_i) \tag{27}$$

and

$$\frac{\partial}{\partial \boldsymbol{x}} \phi_s^{(i)}(\boldsymbol{\mu}_k) = (\boldsymbol{x}_i - \boldsymbol{\mu}_k) \phi_s^{(i)}(\boldsymbol{\mu}_k). \tag{28}$$

Without loss of generality we may consider the original basis functions, f, to be symmetric low-pass filter kernels, whence we must constrain $\phi = f * \kappa$ to be a symmetric low-pass filter kernel also. This constraint, along with equations (27) and (28) may be satisfied by a Gaussian filter

$$\phi_s^{(i)}(\boldsymbol{x}) = \frac{1}{s^d} \exp\left\{ -\frac{|\boldsymbol{x} - \boldsymbol{x}_i|^2}{2s^2} \right\}, \tag{29}$$

whence the normalising factor, w, becomes

$$w = \frac{1}{N(2\pi)^{d/2}}. \tag{30}$$

3.2. Scale-space evolution

We, however, still require a methodology for determining "valid" data structure from the sets of maximal solutions to equation (26). In this section we look, from a primarily theoretical perspective, at the evolution of these solution sets with scale. If we regard the scale as a merging threshold, then the scale-space evolution of peaks in the smoothed density function, equation (24), may be regarded as a form of *dendogram*, akin to that formed from a hierarchical partitioning method. Unlike the latter, however, each datum is not assigned to a partition as the dendogram evolves, thus removing the largest problem with the hierarchical approach. We may, however, utilize the scale-space dendogram to determine partitional structure within the data set. We require that "true" clusters in the data set be stable over a range of scales. We note that other authors have proposed a similar requirement.[13] This stability may be related to the partitional "lifetime" of hierarchical clustering methods.

In order to construct a meaningful scale-space dendogram, we require that, as scale increases, estimated clusters *merge* and do not split (thus the number of solutions to equation (26) is non-increasing with scale). This requirement is depicted in Fig. 1. We may state this requirement more formally as

Theorem 1. Let $\pi(s)$ represent the number of solutions to equation (26) at scale s. For $\phi_s^{(i)}(\boldsymbol{x})$ of equation (29), if $s_1 < s_2$ then $\pi(s_1) \geq \pi(s_2)$.

Proof. We consider an arbitrary function of scale space, $h(\boldsymbol{x},s)$ say. Consider the set of curves formed by solution of $(\partial/\partial\boldsymbol{x})h(\boldsymbol{x},s) = 0$. These solutions may be equated to solutions of equation (26) by letting

$$h(\boldsymbol{x},s) = w\sum_{i=1}^{N}\phi_s^{(i)}(\boldsymbol{x}). \tag{31}$$

Parameterising the evolution of a curve in scale-space by an arbitrary parameter, t say, we may write[18]

$$\frac{\mathrm{d}h}{\mathrm{d}t} = \frac{\partial h}{\partial \boldsymbol{x}}\cdot\frac{\partial \boldsymbol{x}}{\partial t} + \frac{\partial h}{\partial s}\frac{\partial s}{\partial t} = \sum_{n=1}^{d} = \frac{\partial h}{\partial x_n}\frac{\partial x_n}{\partial t} + \frac{\partial h}{\partial s}\frac{\partial s}{\partial t}. \tag{32}$$

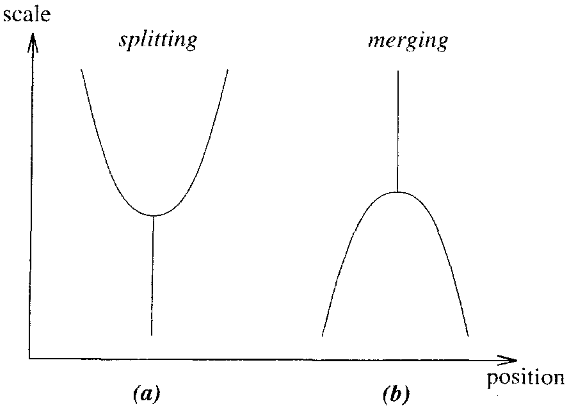

Fig. 1. Forbidden (a) and allowed (b) evolution of scale-space solution sets.

At a point of merging or splitting (as in Fig. 1) so $\partial h/\partial t = 0$ and setting $t = x_l$ (the lth component of $\boldsymbol{x}$) we obtain

$$\frac{\partial h}{\partial x_l} = -\frac{\partial h}{\partial s}\frac{\partial s}{\partial x_l} \tag{33}$$

as $(\partial x_a/\partial x_b) = 0$ for $a \neq b$. Operating equation (33) by $(\partial/\partial x_q)$ we obtain

$$\frac{\partial^2 s}{\partial x_l \partial x_q} = \left(-\frac{\partial^2 h}{\partial x_l \partial x_q} - \frac{\partial s}{\partial x_l}\frac{\partial h}{\partial s \partial x_q}\right)\left(\frac{\partial h}{\partial s}\right)^{-1}. \tag{34}$$

At the turning points of the scale-space curve so $(\partial s/\partial x_l) = 0$ for all l, hence

$$\frac{\partial^2 s}{\partial x_l \partial x_q} = \left(-\frac{\partial^2 h}{\partial x_l \partial x_q}\right)\left(\frac{\partial h}{\partial s}\right)^{-1}. \tag{35}$$

Equation (35) may be expressed, for all l,q, as a Hessian matrix equation

$$\boldsymbol{H}_s = -\boldsymbol{H}_h\left(\frac{\partial h}{\partial s}\right)^{-1}. \tag{36}$$

We may diagonalise equation (36), without loss of generality, by use of a co-ordinate transform $\mathscr{T}$, say, based upon the matrix of normalised eigenvectors of $\boldsymbol{H}_s$ and we denote the new co-ordinate frame by the use of a prime ($'$), hence

$$\boldsymbol{H}_s' = -\boldsymbol{H}_h'\left(\frac{\partial h}{\partial s}\right)^{-1}. \tag{37}$$

If the scale-space turning points are to be maxima (Fig. 1 (b)) then we require each element along the trace of $\boldsymbol{H}_s'$ to be negative, hence for each $l \in [1,d]$,

$$\frac{\partial^2 h}{\partial x_l'^2} = k\frac{\partial h}{\partial s}, \tag{38}$$

where k is an arbitrary *positive* constant. Other authors[18] note that equation (38) is the heat equation, for which a Gaussian is the Green's function. Application of the co-ordinate transform on equation (31) gives

$$h(\boldsymbol{x}',s) = \mathscr{T}\left\{w\sum_{i=1}^{N}\phi_s^{(i)}(\boldsymbol{x})\right\} = w\sum_{i=1}^{N}\mathscr{T}\{\phi_s^{(i)}(\boldsymbol{x})\}. \tag{39}$$

As ϕ is a *symmetric* basis function, i.e. it is functionally dependent upon $|\boldsymbol{x}-\boldsymbol{x}_i|^2$ so it is unaffected by rotation of the co-ordinate frame from $\boldsymbol{x} \rightarrow \boldsymbol{x}'$, but we must introduce a *positive* scaling factor, k_1 say, (positive as the functional form of ϕ is that of an L_2 norm) which accommodates possible changes in the scaling of the axes, hence

$$\mathscr{T}\{\phi_s^{(i)}(\boldsymbol{x})\} = k_1\phi_s^{(i)}(\boldsymbol{x}'). \tag{40}$$

Combining equations (38–40) gives

$$\frac{\partial^2}{\partial x_l'^2}\left\{wk_1\sum_{i=1}^{N}\phi_s^{(i)}(x_l')\right\} = kk_1\frac{\partial}{\partial s}\left\{w\sum_{i=1}^{N}\phi_s^{(i)}(x_l')\right\}. \tag{41}$$

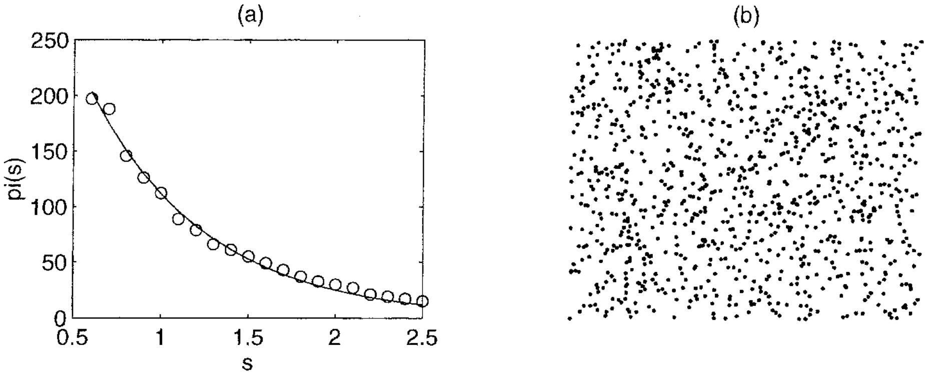

Fig. 2. (a) Decay of $\pi(s)$ with scale and (b) "random" data set.

As the summation and differential operators commute, we obtain, for each (i)

$$\frac{\partial^2 \phi_s^{(i)}(x_l')}{\partial x_l'^2} = k\frac{\partial \phi_s^{(i)}(x_l')}{\partial s}. \tag{42}$$

If

$$\phi_s^{(i)}(x_l') = \frac{1}{s}\exp\left\{\frac{(x_l' - x_l'^{(i)})^2}{2s^2}\right\} \tag{43}$$

then

$$\frac{\partial^2 \phi_s^{(i)}(x_l')}{\partial x_l'^2} = s\frac{\partial \phi_s^{(i)}(x_l')}{\partial s} \tag{44}$$

which equates to equation (42) if k=s. As s is positive, each scale-space turning point is a maxima of the form depicted in Fig. 1(b). The number of turning points, therefore, cannot increase with scale, hence if s_1<s_2 then $\pi(s_1)\geq\pi(s_2)$.

3.3. *Cluster validity*

We consider the case of a datum $\boldsymbol{x}_i$. If there exists some $\boldsymbol{x}_j$ such that $|\boldsymbol{x}_i - \boldsymbol{x}_j| < 2s$ then $\boldsymbol{x}_i, \boldsymbol{x}_j$ will share a common local maxima solution to equation (24). If the set of $\boldsymbol{x}_i$ are uniformly randomly distributed, then the set of $|\boldsymbol{x}_i - \boldsymbol{x}_j|$ are also uniformly randomly distributed. Under these circumstances we expect the number of maxima to decay with scale according to

$$\pi(s) = \pi(0)\exp(-\beta s), \tag{45}$$

where β is a positive constant (related to the dimensionality of the data space). Figure 2(a) shows the decay of $\pi(s)$ with s for the random set of $\boldsymbol{x}_i$ shown in Fig. 2(b). For "random" data, i.e. where no structure is discernable, we expect $\pi(s_1)$>$\pi(s_2)$ for s_1<s_2 for all s_1, s_2. If, however, "valid" data structure exists, then (by our definition) it is stable over a range of scales $s_a \longrightarrow s_b$ such that $\pi(s_a)$=$\pi(s_c)$=$\pi(s_b)$ for all $c \in]a, b[$.

3.4. *Classification*

We now assume that some "significant" partitioning exists at scale s^*; we wish, therefore, to assign a membership function to each datum, $\boldsymbol{x}_i$, to each of the $\pi(s^*)$ partitions.

We consider a simple analogy, by which we "invert" the smoothed function $\hat{p}_s * (\boldsymbol{x})$ so producing a hypersurface with exactly $\pi(s^*)$ minima. "Hard" classification of each datum then consists of evaluating which *basin* of attraction each $\boldsymbol{x}_i$ belongs to. To evaluate this we construct a likelihood function for $\boldsymbol{x}_i$ conditioned on the kth partition ("basin of attraction") of the form

$$p(\boldsymbol{x}_i|C_k) = \frac{1}{2}(1 + \langle\nabla\hat{p}(\boldsymbol{x}_i)\rangle \cdot \langle\boldsymbol{d}_{i,k}\rangle)\exp(-|\boldsymbol{d}_{i,k}|^2/2(s^*)^2), \tag{46}$$

where $\nabla\hat{p}(\boldsymbol{x}_i)$ is the gradient operation on $\hat{p}_{s^*}$ at $\boldsymbol{x}_i$, $\boldsymbol{d}_{i,k} = \boldsymbol{\mu}_k - \boldsymbol{x}_i$ and $\langle\boldsymbol{a}\rangle \cdot \langle\boldsymbol{b}\rangle$ represents the inner product between two normalised vectors. Equation (46) represents the product of two probabilistic penalty terms, the first of which assesses the orientation of the vector between $\boldsymbol{x}_i$ and the kth maxima given an estimate of the PDF at scale s^* and the second acts as to penalise L_2 distance from the kth maxima. The "membership function" for each $\boldsymbol{x}_i$ on C_k is given by the *a posteriori* probability which is obtained from equation (46) via Bayes' theorem and a choice for the *a priori* probability of each partition (normally all assumed to be equal and given by $1/\pi(s^*)$).

4. RESULTS

4.1. *Test data*

We first show the use of all the partitional methods presented in this paper on a simple data set consisting of five Gaussian clusters each of 1000 data. Each $\boldsymbol{x}$ is five-dimensional, but for ease of visualisation the data is projected to a 2-D space using the *Sammon mapping*.[19]

Figure 3(a) shows the data set used in this experiment. Figure 3(b) shows the variation of the *likelihood density* parameter, as defined in equation (15), for both the ML

[2] We note that it is easily possible to look, statistically, for departures of $\pi(s)$ from the form of equation (45)—this is not, however, explored in this paper.[2]

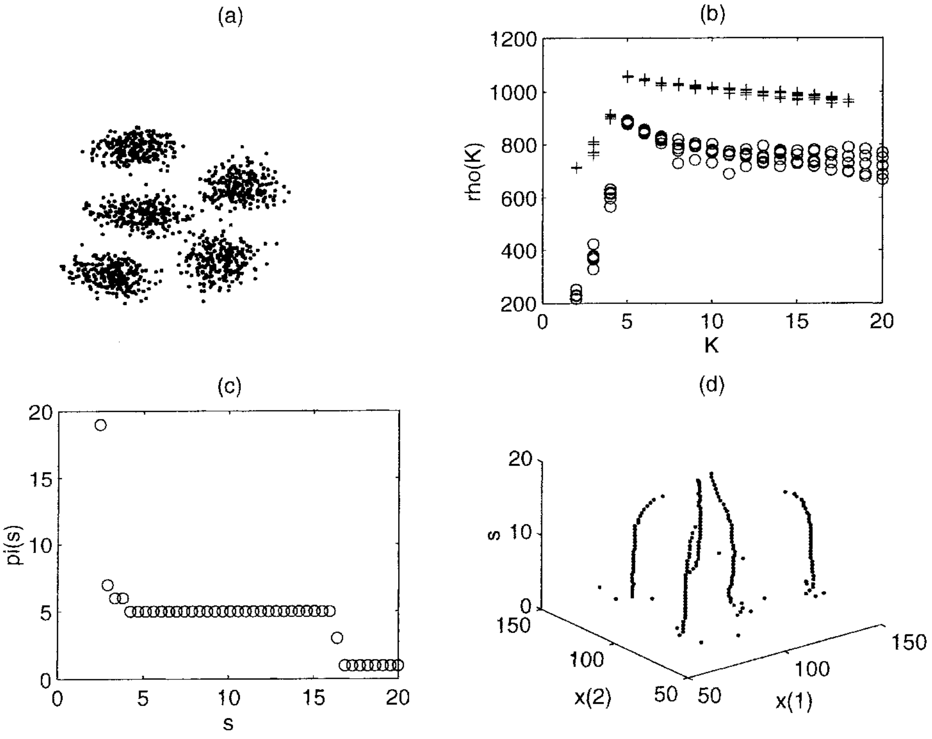

Fig. 3. (a) Simple data set of five Gaussian, (b) (K) curves for ML (o) and K-means algorithms $(+)$, (c) $\pi(s)$ vs s and (d) scale-space evolution of cluster centroids.

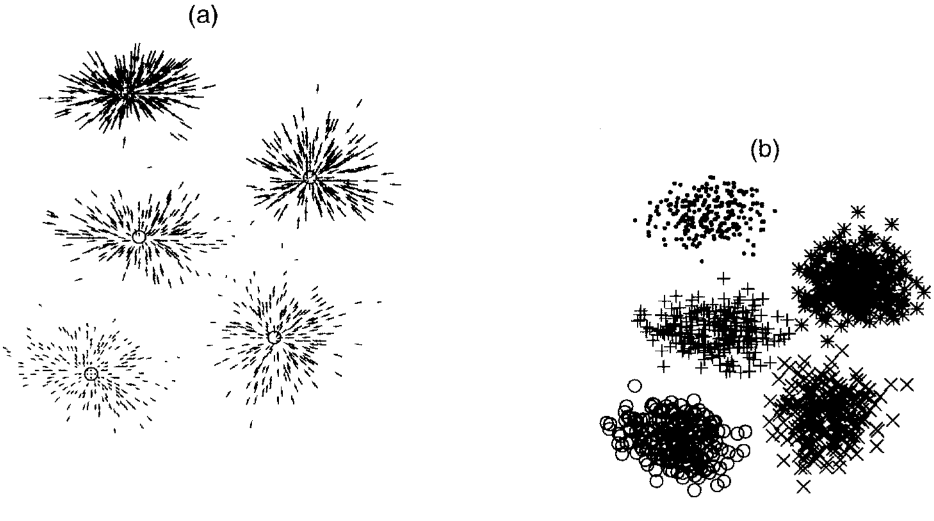

Fig. 4. (a) Partition centroids (circles) and gradient and (b) classification.

(o) and K-means algorithms $(+)$ with K, the number of Gaussians in the mixture models. Fig. 3(c) of the same figure shows the evolution with scale, s, of the number of maxima, $\pi(s)$, of the smoothed density function, and Fig. 3(d) the evolution with scale, s, of the spatial locations of these points of maxima.

Figure 4(a) shows the positions of the cluster centroids at $s=10$ (circles) and the gradient function for each x_i. A "hard" classification into each of the five classes is shown in Fig. 4(b) of the same figure. We note that, for this simple example, all algorithms determined the "correct" number of partitions.

The next example we give is of a series of three non-Gaussian clusters, as shown in Fig. 5(a). As before, Fig. 5(b) shows $\rho(K)$ vs K for the ML and K-means

[3] Note that a series of five runs, each with a different random number seed, have been evaluated.[3]

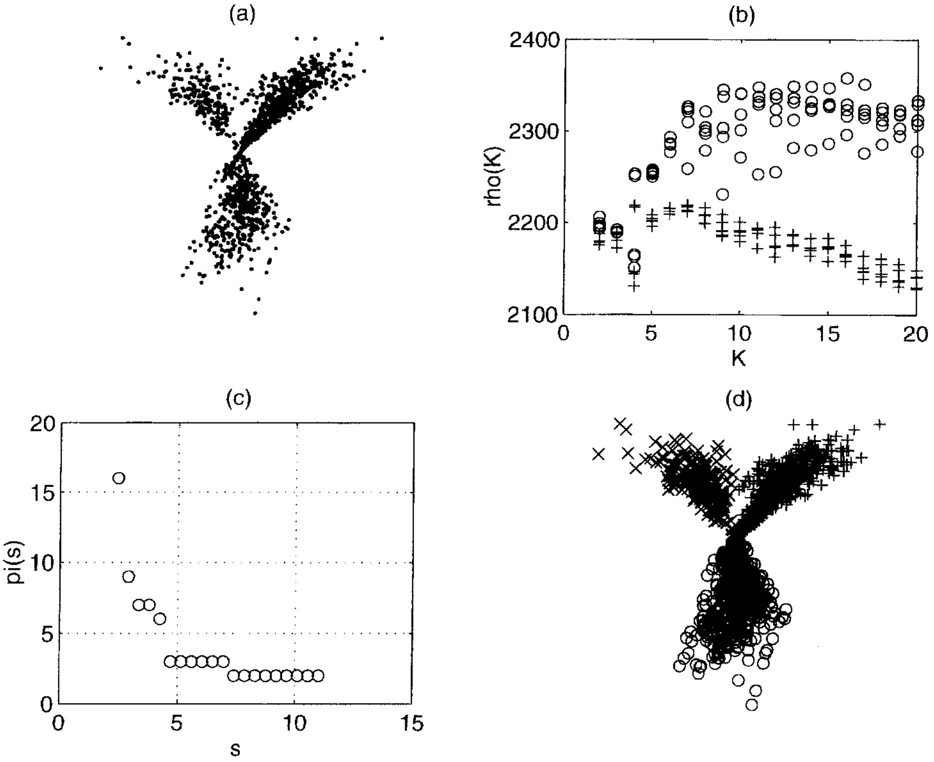

Fig. 5. (a) Data set of three non-Gaussian clusters, (b) (K) curves for ML (o) and K-means algorithms $(+)$, (c) $\pi(s)$ vs s and (d) classification.

algorithms, we note however, that the "correct" structure (K=3) is not determined by either algorithm. Figure 5(c), of $\pi(s)$ vs s, shows a clear partitioning for the scale-space method at $\pi(s)$=3. Figure 5(d) of the same figure shows the resultant partitioning for s=6. It is clear, from this example, that the hyper-ellipsoidal assumption of the ML and K-means algorithms leads to misleading results (it appears that a value of K=7 gives "optimal" partitioning for the K-means algorithm, and a clear partitioning is not obtained with the ML algorithm).

4.2. Partitioning of vibromyography signals

The vibromyogram (VMG) is a non-invasive measurement of muscle sounds. It contains information regarding muscular activity as a function of force and is of importance in the assessment of disability and muscle tremor.[20] In a separate study, the time-domain VMG signal was parameterised over 0.1-second segments using an 8th-order AR model (parameters estimated using the Burg algorithm[21]). Figure 6(a) shows a plot of the first two partial correlation (reflection) coefficients for data accumulated from one subject over eight muscle-force levels. We see that there is a clustering of the data, but that a simple Gaussian description would not be appropriate. Figure 6(b) shows $\pi(s)$ vs s and from this plot we see that a partitioning into $\pi(s)$=4 is significant. Figure 6(c) of the same figure shows the data partitioning for s=6 and Fig. 6(d) the time course of the classification for the subject. As the time index progresses the subject is exerting increasing muscle force levels and Fig. 6(d) clearly shows changes in "state" as this force increases. It is interesting to note that the plot shows changes in "state" over low to mid range force, but a return to previously visited "states" at the higher force levels (large time index). This corresponds well to current speculations regarding recruitment of fibres within muscles; work in this area is ongoing.

4.3. Segmentation of simple texture images

Variation in *texture* often provides important information regarding the boundaries of objects in an image and much effort has been directed at methods of segmenting images on a textural basis. One of the primary problems is that the number of segments within the image is unknown (in the majority of cases) *a priori*. To this end, much effort has been directed at the application of unsupervised clustering methods to features extracted from images.[22–26] It is unreasonable to expect, however, for feature-space data sets, constructed from the image, to have a simple hyper-ellipsoidal cluster structure, and hence the use of algorithms such as K-means may be inappropriate.

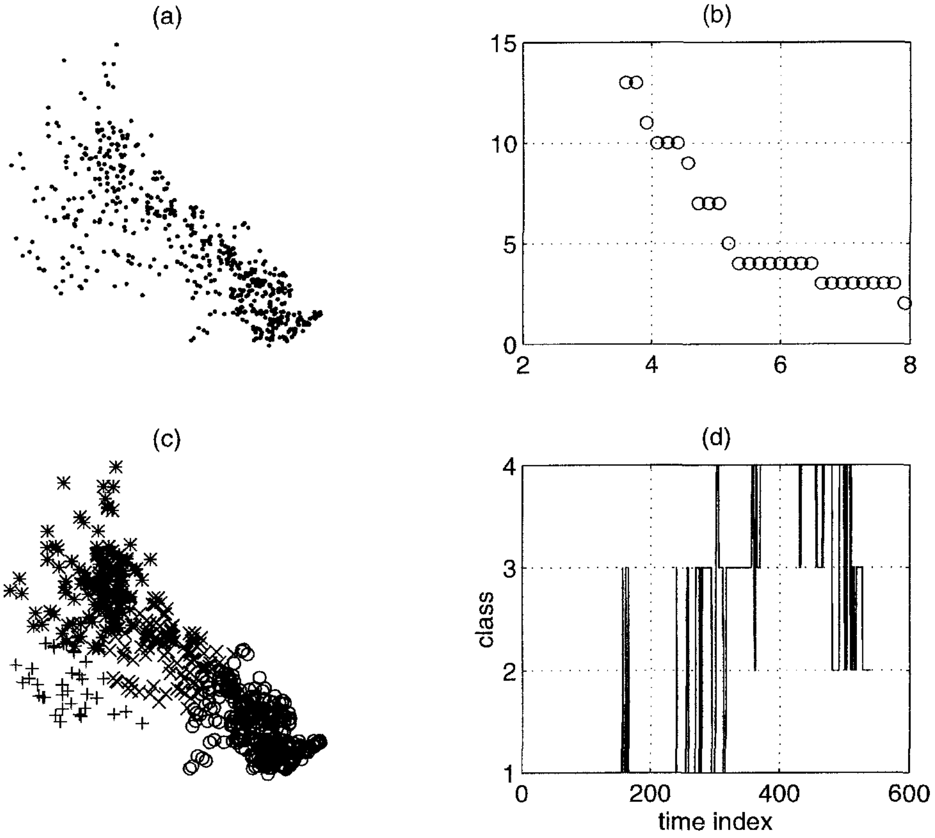

Fig. 6. (a) VMG data—first two reflection coefficients from AR model, (b) decay of $\pi(s)$ with s, (c) data partitioning and (d) time course of classification (larger time index corresponds to higher muscle forces).

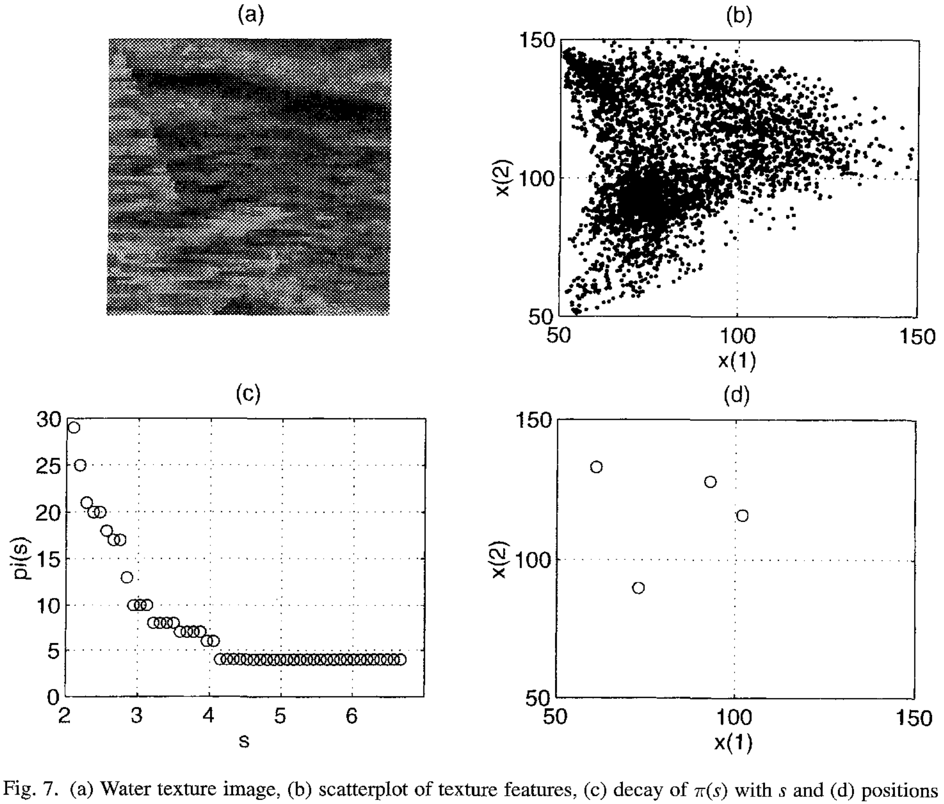

Fig. 7. (a) Water texture image, (b) scatterplot of texture features, (c) decay of $\pi(s)$ with s and (d) positions of maxima of the smoothed PDF.

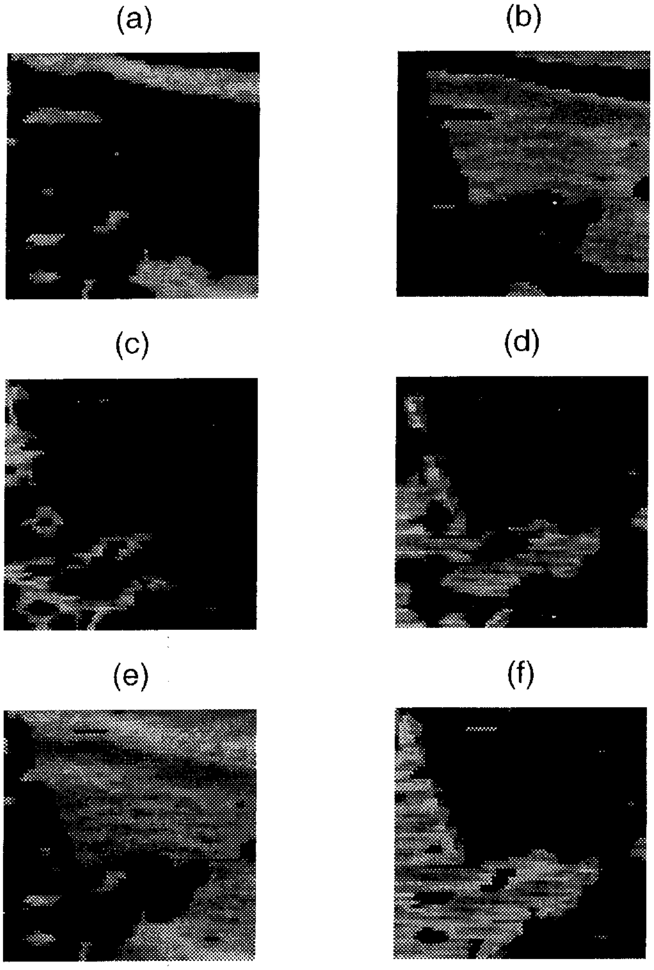

Fig. 8. Masking of the "water" image using *a posteriori* probabilities for four partitions—(a) to (d) respectively. Plots (e) and (f) show the masking of the image using a two-fold partitioning of the image.

We take two simple examples in this paper and investigate image segmentation based upon well-known textural measures. The first test image consists of a 64×64 grey-scale (8-bit) image manufactured from two Brodatz macro-texture images of water ("finer" and "coarser"). The boundary between the two macro-texture regions takes the form of a section of a "star" and may be seen most clearly at the top left-hand corner of the image in Fig. 7(a). Two texture measures are evaluated from a 7×7 sliding mask applied to the image. Firstly *correlation* estimated from the co-occurrence matrix[27] and secondly the first grey-scale moment of the *grey-scale run length matrix* (GSRLM) as proposed in reference (28). Full details of both these texture measures may be found in the given references.

Figure 7(b) shows the 2-D scatter plot obtained from the texture features. Application of the scale-space method described in this paper gives a decay of $\pi(s)$ with scale as shown in Fig. 7(c) of the same figure. Note that, although there are "ledges" in the decay curve prior to $\pi(s)=4$, the latter forms a very scale-robust partitioning of the data set. Figure 7(d) shows the resultant points of maxima in the smoothed density function created from the data set.

Figure 8(a)–(d) present the masking of the image by the partition *a posteriori* probabilities. Comparison of these plots to the original image shows a realistic partitioning of textural regions. If, however, we follow the evolution of the four partitions as they merge at larger scales we observe that a two-fold partitioning of the data is obtained whereby partitions 1 and 2 (Fig. 8(a) and (b)) merge and partitions 3 and 4 (Fig. 8(c) and (d)) merge. The resultant partition probabilities, used so as to mask the image, are shown in Fig. 8(e) and (f) of the same figure. It is clear from the latter that the basic structure of the image's textural regions is recovered. One of the benefits, therefore, of the proposed scale-space method is that the evolution of partitions is mapped and provides detailed information about the merging and splitting of image segments.

The second test image is the (well-known) "house" image, configured here as a 128×128 grey-scale image

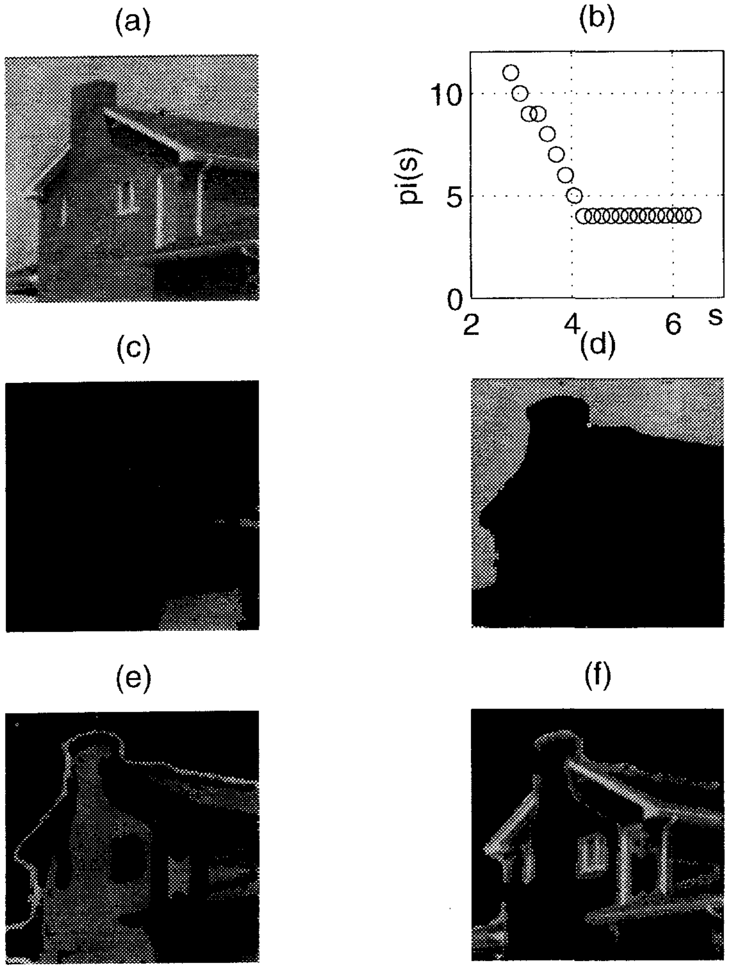

Fig. 9. (a) "House" image, (b) decay of $\pi(s)$ with s, (c)–(f) masking of the "house" image using a posteriori probabilities for four partitions, respectively.

and shown in Fig. 9(a). Two of the Laws microstructure texture measures(27,29,30) were obtained for each pixel within the image calculated via application of the following 3×3 masks:

$$L_1=\begin{bmatrix}1&2&1\\2&4&2\\1&2&1\end{bmatrix}L_3=\begin{bmatrix}-1&2&-1\\-2&4&-2\\-1&2&-1\end{bmatrix}.$$

Figure 9(b) shows the decay of $\pi(s)$ with s and a clear partitioning is obtained for $\pi(s)=4$. Figure 9(c)–(f) show the masking of the image with the *a posteriori* probabilities from each partition respectively. Note, for example, that Fig. 9(f) shows regions of "detailed" structure within the image, i.e. windows, guttering etc.

5. CONCLUSIONS

Two separate methods have been investigated in detail within this paper. First, a parametric method of Gaussian mixture modelling (GMM) achieved using both maximum-likelihood and K-means algorithms; second, a method of scale-space parameter estimation based upon successive smoothings using a Gaussian kernel function. For both methodologies a cluster validity criterion is introduced, the concept of the *likelihood density* for the parametric methods and that of "significant" scale robustness in the decay of the parameter $\pi(s)$ with scale for the scale-space method. It has been shown that the scale-space method is more robust in cases where hyper-ellipsoidal partitions may not be assumed. Furthermore, the scale-space method is not apt to consume structures with a small number of exemplars as part of a densely populated structure, as the Gaussian mixture techniques tend to do. It is, therefore, well suited to the task of preserving the structure and integrity of small outlying structures within a data set. The overall analysis of such outliers is a complex one (as described in reference (31)), but it is felt that preservation of such structures is important in a primary partitioning phase even if they are subsequently removed after examining the spectrum of estimated partition priors. Furthermore, it is noted that much information regarding structures in a data set is also obtained from the merging pattern the structures trace in scale-space.

As must have been noted, no doubt, the data sets examined in this paper are of low dimensionality. Although the GMM methods are easily extended to higher dimensional spaces, visualisation, interpretation and (perhaps most importantly) verification of the results is difficult. Similarly, the scale-space method may, in principle, be extended to a data set in an arbitrary dimensional space, but the complexity of determining the number and location of the turning points of $\hat{p}_s(\boldsymbol{x})$ rises rapidly with the space's dimension. Although this may be a disadvantage for some data sets, most high-dimensional data sets may be mapped (with, naturally, some loss) onto a low-dimensional manifold using, for example, Kohonen's topographic map(32) or Sammon's non-linear mapping.(19) It is noted that more efficient algorithms than a simple grid search exist for evaluating the zeroes of multi-dimensional functions(33) and their use is an area of future implementation research.

Acknowledgements—The author would like to thank Alan Outten for providing data and information regarding the VMG and Joao Batista for providing the texture data and for helpful discussions. Thanks also to Ioannis Matalas for comments on the draft manuscript.

REFERENCES

1. A. K. Jain, *Classification, Pattern Recognition and Reduction of Dimensionality*, Krishnaiah and Kanal, eds, Vol. 2, Chapter 2. North-Holland, Amsterdam (1982).
2. A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs, New Jersey (1988).
3. J. A. Hartigan, *Clustering Algorithms*. Wiley, New York (1975).
4. B. Everitt, *Cluster Analysis*. Wiley, New York (1974).
5. R. C. Dubes and A. K. Jain, Validity studies in clustering methodologies, *Pattern Recognition* **11**, 235–254 (1979).
6. L. A. Zadeh, Fuzzy sets, *Inform. Control* **8**, 338–353 (1965).
7. I. Gath and B. Geva, Unsupervised optimal fuzzy clustering, *IEEE Trans. Pattern Analysis Mach. Intell.* **11**(7), 773–781 (1989).
8. M. P. Windham, Cluster validity for the fuzzy c-means clustering algorithm, *IEEE Trans. Pattern Analysis Mach. Intell.* **4**(4), 357–363 (1982).
9. R. L. Cannon, J. V. Dave and J. C. Bezdek, Efficient implementation of the fuzzy c-means clustering algorithms, *IEEE Trans. Pattern Analysis Mach. Intell.* **8**(2), 248–255 (1986).
10. J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981).
11. K. Urahama, Unsupervised learning algorithm for fuzzy clustering, *IEICE Trans. Inf. Syst.* **E76-D**(3), 390–391 (1993).
12. R. Lopez de Mantaras and L. Valverde, New results in fuzzy clustering based on the concept of indistinguishability relation, *IEEE Trans. Pattern Analysis Mach. Intell.* **10**(5), 754–757 (1988).
13. R. Wilson and M. Spann, A new approach to clustering, *Pattern Recognition* **23**(12), 1413–1425 (1990).
14. H. G. C. Tråvén A neural network approach to statistical pattern classification by "Semiparametric" estimation of probability density functions, *IEEE Trans. Neural Networks* **2**(3), 366–377 (1991).
15. S. Roberts and L. Tarassenko, A probabilistic resource allocating network for novelty detection, *Neural Computation* **6**, 270–284 (1994).
16. A. P. Dempster, N. M. Laird and D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. Roy. Statist. Soc.* **39**(1), 1–38 (1977).
17. E. Parzen, On estimation of a probability density function and mode, *Ann. Math. Stat.* **33**, 1065–1076 (1962).
18. A. L. Yuille and T. A. Poggio, Scaling theorems for zero crossings, *IEEE Trans. Pattern Analysis Mach. Intell.* **8**(1), 15–25 (1986).
19. J. W. Sammon, A nonlinear mapping for data structure analysis, *IEEE Trans. Computers* **18**(5), 401–409 (1969).
20. A. Outten, Analysis of the vibromyogram in the assessment of brain injured patients, Technical Report, Imperial College, University of London (1995).
21. N. Andersen, Comments on the performance of maximum entropy algorithms, *IEEE Proc.* **66**(11), 1581–1582 (1978).
22. A. K. Jain and F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, *Pattern Recognition* **24**(12), 1167–1183 (1991).
23. T. R. Reed and J. M. H. du Buf, A review of recent texture segmentation and feature extraction techniques, *CVGIP: Image Understanding* **57**(3), 359–372 (1993).
24. M. Spann and R. Wilson, A quad-tree approach to image segmentation which combines statistical and spatial information, *Pattern Recognition* **18**(3/4), 257–269 (1985).
25. G. B. Coleman and H.C. Andrews, Image segmentation by clustering, *IEEE Proc.* **67**(5), 773–785 (1979).
26. L. O. Hall, A. M. Bensail, L. P. Clark, R. P. Velthvizen, M. S. Silbiger and J. C. Bezdek, A comparison of neural network and fuzzy clustering techniques in segmenting MR images of the brain, *IEEE Trans. Neural Networks* **3**(5), 672–682 (1992).
27. R. M. Haralick and L.M. Shapiro, *Computer and Robot Vision*, Vol. 1. Addison-Wesley, Reading, Massachusetts (1993).
28. H. H. Loh, J. G. Leu and R. C. Luo, The analysis of natural textures using run length features, *IEEE Trans. Indus. Electr.* **35**(2), 323–328 (1988).
29. W. K. Pratt, *Digital Image Processing*. Wiley, New York (1991).
30. K. I. Laws, Textured image segmentation, Technical Report USCIPI 940, University of Southern California, January (1980).
31. R. J. Beckman and R. D. Cook, Outlier...........s, *Technometrics* **25**(2), 119–149 (1983).
32. T. Kohonen, Self-organized formation of topographically correct feature maps, *Biol. Cybernetics* **43**, 59–69 (1982).
33. W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in C*. Cambridge University Press, Cambridge (1991).

About the Author—STEPHEN J. ROBERTS graduated from Oxford University with a degree in physics. He worked in an industrial research department before returning to Oxford to undertake research towards the degree of D.Phil. in the area of artificial neural networks applied to medical data analysis. He subsequently held postdoctoral positions in neural network research and was lecturer in Engineering Science at St. Hugh's College, Oxford, for three years prior to his appointment as lecturer in the Department of Electrical and Electronic Engineering at Imperial College in 1994. His research interests include neural networks, scale-space methods, image and signal processing, Bayesian methods, machine learning and AI.