A NEW APPROACH TO CLUSTERING

R. WILSON* and M. SPANN†

* University of Warwick, Coventry, UK; † University of Birmingham, Birmingham, UK

(Received 19 December 1988; in revised form 6 February 1990; received for publication 26 April 1990)

Abstract—Estimation theory is used to derive a new approach to the clustering problem. The new method is a unification of centroid and mode estimation, achieved by considering the effect of spatial scale on the estimator. The result is a multiresolution method which spans a range of spatial scales, giving enhanced robustness both to noise in the data and to changes of scale in the data, by using comparison between scales as a test of cluster validity. Iterative and non-iterative algorithms based on the new estimator are presented and are shown to be more accurate than simple scale-space filtering in identifying and locating the cluster centres from noisy test data. Results from a wide range of applications are used to illustrate the power and versatility of the new method.

Clustering Scale-space Multi-resolution Estimation

1. INTRODUCTION

The classification of N-dimensional data is a widely occurring problem, with applications ranging from data compression to pattern recognition. When the statistics of the data are known a priori, techniques ranging from Bayesian methods to maximum likelihood techniques can be employed. In cases where the statistics are known in parametric form, estimation of the parameters is possible, e.g. refs 1 and 2.

However, there are many applications where such a priori information is unavailable and (or) the data set is not large enough to make reliable estimates of class parameters. In the worst case, not even the number of classes is known. Such problems are typical in applications such as image segmentation, for example references.3,4

One approach to such applications is clustering. Clustering algorithms span a wide range of disparate techniques, ranging from spanning trees to least squares methods.5,6 The bewildering variety of clustering methods reflects in part the diversity of problem domains and in part the lack of a rigorous basis for the problem: clustering is most often applied in problems where no clear mathematical formulation exists. Indeed many algorithms require no more than the existence of a set of data points in some suitable metric space, although some also require knowledge of the number of classes.6 Indeed many of these methods can be applied to a given problem, there is in general no guarantee that any two will produce consistent answers, leading to the vexed issue of "cluster validity".5,6 Indeed one rule of thumb for establishing the validity of the clusters produced by one method is to compare them with those produced by a second method. This is clearly an unsatisfactory state of affairs.

One way out of this difficulty is to take a step back towards statistical theory and formulate the problem as one of estimating the means or modes of a mixed probability density p(x). Even this is a non-trivial problem, however, as it confronts the user with a choice between an essentially global estimate of the mean—the centroid—or one which is inherently local—the mode. It is clearly desirable, moreover, that the estimate should be robust both to changes of spatial scale and to the presence of the noise which inevitably accompanies the collection of data.

A solution to all of these difficulties can be found by considering a multiresolution approach to the problem. A general form of estimator, which is intermediate in scale between the global, centroid estimator and the simplest form of mode estimator—a peak detector—is derived in the next section, using heuristic arguments. These are strengthened by application of a Wiener filtering approach,7 which is presented in the appendix. It is shown that the new estimator can be seen equally well as a windowed form of centroid estimator or as a peak detector operating on smoothed data. Moreover, by comparing and combining estimates from different scales, it is possible to give the final estimate of cluster positions a significant degree of robustness to noise in the data and changes of scale. The use of a multiresolution approach also provides a natural definition of cluster validity.

In terms of implementation, the new estimators are based primarily on the use of linear filtering methods. In this sense, they can be regarded as an extension of the Kernel estimators of probability density8–10 to cope with the problem of unknown cluster scale. Those readers familiar with scale-space methods11,12 might see them as an adaptation of scale-space analysis to the clustering problem. Indeed, the implementations presented in Section 3 are compared in Section 4 with a scale-space method and are shown to give more accurate results. Both a simple signal-detection implementation and an iterative algorithm are presented, the choice between
them depending on the nature of the application and the available hardware. In addition to the test results, the iterative algorithm has been applied to a wide range of clustering problems, including image segmentation, quantisation and line thinning. Results of these applications are also presented to demonstrate the effectiveness and versatility of the new method.

2. CLUSTERING AS AN ESTIMATION PROBLEM

2.1. Towards a definition of clusters

The fundamental assumption behind the work presented here is that clusters represent "essentially isotropic" distributions of some positive quantity, such as mass, energy or probability (frequency), on a continuous real domain. This implies that, to a first approximation, each cluster can be represented by a single position parameter—the "cluster centre". While the problem of non-numerical data can often be overcome, by attaching a uniform mass to each datum, for example, the restriction on topology seems essential, if a description in terms of a single position parameter is to suffice (cf Section 4, however).

It is useful to begin with the simplest case: functions \( f(x) \) on the real line. Clearly some restrictions must be placed on the form of \( f(x) \) if it is to represent a cluster with an unambiguously defined position. Consider first the set \( \mathcal{F}_0 \) of basic cluster functions: those centred at the origin. In conformity with the above notions, a function \( f(x) \in \mathcal{F}_0 \) if:

2.1.1. It is continuous, non-negative and bounded: \( k > f(x) \geq 0 \) \( \forall x \) for some \( k > 0 \).

2.1.2. It is even: \( f(-x) = f(x) \forall x \). This is about the only reasonable interpretation of the isotropy requirement in one dimension.

2.1.3. It has a single peak: \( f'(x) \leq 0 \) \( x > 0 \). Otherwise it might not be considered to represent a single cluster.

2.1.4. It is localised: \( f(x) = 0 \) \( |x| > X \), for some \( X > 0 \).

The need for locality arises from the desire to talk unambiguously about the existence of a number of clusters. While other versions, such as finite moments, are possible, the stated requirement of finite support is technologically the simplest and practically the most meaningful.

The resulting set \( \mathcal{F}_0 \) thus consists of the totality of functions which satisfy these four conditions. Of central importance to the sequel are the closure properties of \( \mathcal{F}_0 \): given that the functions \( f_1(x), f_2(x), \ldots \) belong to \( \mathcal{F}_0 \), what related functions \( g_1(x), g_2(x) \) can be shown to belong in the set? There are two particularly important such properties:

(i) If \( f(x) \in \mathcal{F}_0 \), \( \alpha > 0 \) is some constant and \( g(x) = \alpha f(x) \), then \( g(x) \in \mathcal{F}_0 \). In other words, magnitude scaling of a cluster function results in another cluster function.

(ii) If \( f(x) \in \mathcal{F}_0 \) and \( \alpha > 0 \) is some constant and \( g(x) = f(ax) \) then \( g(x) \in \mathcal{F}_0 \): changing the spatial scale of a cluster does not affect membership of the set \( \mathcal{F}_0 \).

These results follow directly from the definitions. Their significance in the clustering problem is quite simply that these transformations—magnitude scaling and axis scaling—convert any valid clustering problem into another one which is equally valid. It follows that a main aim of the cluster analysis is to obtain results which are as far as possible invariant to such transformations of the data.

2.2. Estimating the positions of clusters

The simplest non-trivial clustering problem consists of finding the centres of \( n \) "well separated" clusters. This amounts to forming the data \( g(x) \) as a sum of suitably shifted basic functions, i.e.

\[
g(x) = \sum_{i=0}^{n-1} a_i f(x - x_i) \quad a_i > 0 \quad 0 \leq i < n
\]  

where to avoid any ambiguity, the distance between adjacent clusters is sufficient to prevent their overlapping

\[
x_i - x_{i-1} \geq X \quad 1 \leq i < n
\]

where \( X > 2X_1 \) and

\[
f(x) = 0 \quad |x| > X_1.
\]

In other words, there are clear gaps between successive clusters. Many clustering algorithms (see for example reference 6) are based on the use of the centroid

\[
\mu = \int_{-\infty}^{\infty} xg(x) \, dx \Big/ \int_{-\infty}^{\infty} g(x) \, dx.
\]

While this is fine if \( n = 1 \), it clearly will not work for \( n > 1 \) because it is a global estimate. For \( g(x) \) of equation (1) it is easy to show that

\[
\mu = \sum_{i=0}^{n-1} \frac{a_i x_i}{\sum_{i=0}^{n-1} a_i}
\]

which bears no useful relation to any \( x_i \) unless \( n = 1 \). The common solution to this problem is to partition the data and to estimate the centroid of each partition. However, when neither \( n \) nor the constants \( a_i \) are known, choice of an appropriate partition is no easy task and there is no guarantee that a given initial partition will lead to the correct results, that is the positions \( x_i, 0 \leq i < n \).

An alternative approach, which is guaranteed to yield the right answers in the above case, is to detect the peaks of \( g(x) \)—the positive-negative zero-crossings of the derivative \( g'(x) \). That there are exactly \( n \) such zero-crossings, at points \( x_i \), follows directly from the definitions 2.1.2 and 2.1.3 and the separation between clusters, equations (2) and (3), which
together imply that
\[ g'(x) \leq 0 \quad 0 \leq i < n \]
where \( x_{-1} = -\infty, x_n = \infty \).

Unfortunately, in a more realistic case, where the data are noisy, this simple technique fails because of the noise sensitivity of the differentiation. Thus if, in place of equation (1), the data are defined by
\[ w(x) = e^{-\frac{x^2}{2}} \]
the two forms are identical
\[ w'(x) = -xw(x). \]

Now if a kernel of the form of equation (12) is convolved with the data, this can be written as
\[ \eta(x) = -\int_{-\infty}^{\infty} yw(y)g(x - y) \, dy \]
which is equivalent, up to normalisation, to the calculation of the centroid of the data windowed by \( w(y) \) centred at point \( x \).

As a final proof of the connection between the two approaches, it is shown in the appendix that for noiseless data in the limit as the window \( w(x) \) becomes arbitrarily narrow, the estimate \( \eta(x) \) tends to the derivative of \( g(x) \). To sum up, whether the estimation procedure is regarded as a "localised" version of centroid estimation or as smoothing the peak detection is a matter of taste: estimation kernels of the form of equations (8) or (12) are intermediate between the ends of a spectrum represented by centroid at one extreme and peak detection at the other. This allows a trade-off to be made between spatial resolution and noise immunity.

Now if the important parameters affecting the selection of the window \( w(x) \)—the exact cluster function \( f(x) \) and noise statistics—are known, then the optimum estimator defines \( w(x) \). In clustering applications, however, this is seldom the case. Selection of the shape of the window is not especially critical and can be based on computational complexity, perhaps supplemented by general arguments, such as the uncertainty principle.\(^{16}\)

Choice of the appropriate spatial scale is essential, however, if satisfactory results are to be obtained (it will be recalled from the previous section that an ideal clustering method would be invariant to the choice of spatial scale). Indeed, this is the case for both the unwindowed centroid of equation (4) and the derivative (equation 6), in the sense that if the cluster centre \( x_0 \) is identified by either method from the continuous (noiseless) data \( g(x) \), then the centre \( x_0/\alpha \) will be identified from the function \( g(\alpha x) \). This result is easily proved by substitution of \( (\alpha x) \) for \( \alpha \) in equations (4) and (6). When a window \( w(x) \) is employed no such scale-invariance exists, even when

\[ h(x) = -xw_1(x) \]
g(x) is continuous. A simple corollary to the result on the limiting behaviour of the windowed estimates (Appendix, Section 3) shows that if the spatially scaled estimate \( \eta_\alpha(x) \) is defined by

\[
\eta_\alpha(x) = \alpha^3 \int_{-x}^{x} y w(\alpha y) g(x + y) \, dy
\]

then there exists, for every \( \delta > 0 \), an \( \alpha > 0 \), such that if \( g(x) \) is continuous, then for any \( \alpha_i \geq \alpha \),

\[
|\eta_{\alpha_i}(x) - \eta_\alpha(x)| < \delta \quad \forall x.
\]

In other words, for small enough windows, the estimates are essentially the same. This result is also proved in the appendix. Similarly, if the data \( g(x) \) is scaled spatially to give \( g(\alpha x) \) and the window \( w(x) \) fixed, the windowed centroid will exhibit the same limiting behaviour for large \( \alpha \). Thus there exists at least a partial invariance to changes of scale.

When noise is present, even this weakened claim cannot be substantiated—rather than being bounded on one side, the range of scales is bounded on both sides, in the following way. On one hand, as above, if the window \( w(\alpha x) \) is at a scale parameter \( \alpha \) which is too small, then the window will be too large and nearby clusters will interfere with each other—more than one cluster will on average be present in a given window. Clearly this depends on the spatial extent of \( w(\alpha x) \), which can be appropriately measured by its variance \( \sigma^2(\alpha) \)

\[
\sigma^2(\alpha) = \int_{-x}^{x} x^2 w(\alpha x) \, dx / \int_{-x}^{x} w(\alpha x) \, dx
\]

or

\[
\sigma(\alpha) = \frac{1}{\alpha} \sigma(1).
\]

On the other hand, too large a value of \( \alpha \) means too high a noise variance. Using the form of estimate \( \eta_\alpha(x) \) of equation (16) guarantees that the estimate gives a constant (independent of \( \alpha \)) response to a linear ramp—a reasonable constraint for a gradient estimator. The corresponding output noise variance resulting from unit input noise is just

\[
v^2(\alpha) = \alpha^6 \int_{-x}^{x} x^2 w^2(\alpha x) \, dx = \alpha^2 v^2(1)
\]

or

\[
v(\alpha) = \alpha^{3/2} v(1)
\]

As might be expected, \( v(\alpha) \) is monotonic increasing with \( \alpha \), while \( \sigma(\alpha) \) is monotonic decreasing. A given window scale can therefore be seen as a particular trade-off between spatial resolution and noise sensitivity. For given data \( g(x) \), there will always be one window scale which represents the lowest uncertainty and there may well be a range of scales for which the spatial resolution is high enough and the noise variance low enough to give satisfactory performance. Equivalently, a given window scale will be optimal—or close to optimal—for a bounded range of scales of data \( g(\alpha x) \). In other words, invariance has been lost and replaced by what amounts to a degree of robustness or insensitivity to changes of scale.

It follows that the clustering procedure cannot be based on a single scale of window. Windows of a wide range of scales must be employed if this important invariance is to be restored. This is the approach adopted below. It shares many features with the multiresolution or scale-space methods currently employed in a range of image and pattern analysis problems. (4, 11-13, 14, 15)

2.3. Using the estimates

The problem remains of using the estimated function \( \eta_\alpha(x) \) to obtain the cluster positions. Two general approaches to this problem have been implemented. The first is an iterative procedure based on the normalised estimate \( \mu_\alpha(x) \) defined by

\[
\mu_\alpha(x) = \eta_\alpha(x) / \alpha^3 \int_{-x}^{x} w(\alpha y) g(x + y) \, dy
\]

Unlike the linear estimate \( \eta_\alpha(x) \), \( \mu_\alpha(x) \) is clearly independent of amplitude scaling: if \( g(x) \) is replaced by \( a g(x) \), \( \mu_\alpha(x) \) is unchanged. In fact, \( \mu_\alpha(x) \) is the centroid of the windowed data. The iterative scheme uses these local centroid estimates to perform a locally adaptive axis scaling. Giving the data \( g(x) \), a sequence of functions \( g^n(x) \) is formed using the rule

\[
g^n(x + \mu_{n-1}(x)) = g^{n-1}(x)
\]

where

\[
g^n_\alpha(x) = g(x)
\]

and

\[
\eta^n_\alpha(x) = \alpha^3 \int_{-x}^{x} w(\alpha y) g^n_\alpha(x + y) \, dy
\]

The nonlinearity of this algorithm has so far prevented any detailed analysis. Some empirical results are presented in Section 4, however.

The alternative is to regard the function \( \mu_\alpha(x) \) (or \( \eta_\alpha(x) \)) as a pattern to be detected. The prototypical pattern is simply from equations (12) and (16)

\[
\eta_\alpha(x) = -x w(\alpha x)
\]

giving a matched filter

\[
m_\alpha(x) = \eta_\alpha(-x) = x w(\alpha x)
\]
be used. Of course, if the linear estimate \( \eta(x) \) is used, this amounts to no more than the convolution of \( g(x) \) with the kernel \( m(x) \) convolved with that of the estimator \( h(x) \), which by the above can be written as

\[
k(x) = \int_{-\infty}^{\infty} w'(y)w'(x + y) \, dy
= w'(x)
\]

(28)

the second derivative of the window \( w(x) \). This approach, when applied over a range of resolutions, has been called scale-space filtering.\(^{(11)}\) It should be pointed out, however, that scale-space filtering, in common with Marr's edge detection,\(^{(17)}\) employs zero-crossings of the convolution of the data with a set of kernels of the form of \( k(\alpha x) \), where \( \alpha > 0 \) is the scale parameter, whereas here a more conventional threshold detection is indicated.

Once the cluster centres have been identified, classification may proceed on the basis of simple distance or, by assuming a parametric (e.g. Gaussian) form for the cluster functions, by a likelihood criterion. For the iterative algorithm, however, this is unnecessary, since all the original points \( x \) for which \( g(x) > 0 \) are automatically mapped onto cluster centres by the algorithm (cf Section 3.1 below).

In each case, the final result of the estimation is for each value of the resolution parameter \( \alpha \geq 1 \), a finite set of \( n(\alpha) \) clusters, each of which may be described by an interval \([ x^l_\alpha, x^u_\alpha ] \) of the real axis and a cluster centre \( x^c_\alpha \), which are renormalised for the scale parameter \( \alpha \), so that a single cluster function \( f(x) \) with centre \( x^c_\alpha \) has, in an ideal case

\[
n(\alpha) = 1, x^c_\alpha = x_f \quad \forall \alpha.
\]

(29)

2.4. Checks on cluster validity

Once the clusters have been identified by one of the above methods, there remains the taxing problem of establishing their validity.\(^{(5)}\) As implied in the above discussion, the effect of changing resolution is one aspect of this. Another is quite simply the internal consistency of the clusters obtained at a given resolution. More precisely, let \( x^c_i(\alpha) \) be the \( i \)-th estimated cluster centre at resolution \( \alpha \) and let \( x^l_\alpha, x^u_\alpha \) be the respective upper and lower bounds. Then the normalised internal consistency measure at \( \alpha \), \( \rho(\alpha) \), is defined to be

\[
\rho(\alpha) = \max\left\{ \frac{1}{n(\alpha)} \int_{x^l_\alpha}^{x^u_\alpha} f(x) \, dx \left| \int_{x^l_\alpha}^{x^u_\alpha} f(x) \, dx \right. \right\} \left( x - x^c_i(\alpha) \right)
\]

(30)

Note that \( 0 \leq \rho(\alpha) \leq 1 \) and that if, for all \( i \), \( x^c_i(\alpha) \) is the centroid of the \( i \)-th cluster then \( \rho(\alpha) = 0 \). Thus a simple thresholding of \( \rho(\alpha) \) is sufficient to determine if all the clusters are internally consistent, in the sense that their estimated centroids correspond to their true centroids.

The second check on cluster validity is based on the observation in Section 2.1 that cluster centres should be invariant to changes of resolution. Thus if the scale parameter \( \alpha \) is changed in an increasing sequence \( \alpha_0, \alpha_1, \alpha_2, \ldots \) with \( \alpha_k > \alpha_{k-1} \), then a clustering will be regarded as consistent if for some \( k \)

\[
n(\alpha_k) = n(\alpha_{k-1})
\]

\[
x^c_i(\alpha_k) = x^c_i(\alpha_{k-1}) \quad i = 1, 2, \ldots, n(\alpha_k).
\]

(31)

These equations express precisely the notion of consistency referred to in Section 2.2. In practice, it has been observed that it is normally sufficient to test only the number of classes \( n(\alpha) \).

2.5. Extension to multidimensional clusters

The extension to \( N \)-dimensional data is a straightforward generalisation of the generic estimator kernel \( h(x) \) of equation (12). A vector filter \( h(x) \) is defined as

\[
h(x) = -x w(r)
\]

(32)

where \( w(\cdot) \) is a window function,

\[
x = (x_1, x_2, \ldots, x_N)^T
\]

(33)

and

\[
r^2 = \sum_{i=1}^{N} x_i^2.
\]

(34)

Convolution of \( h(x) \) with the scalar field \( g(x) \) results in a vector field \( \eta(x) \) of dimension \( N \) with components

\[
\eta_i(x) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_i(v)g(x - v) \, dv
\]

(35)

or, when normalised

\[
\mu_i(x) = \eta_i(x) \left/ \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} w(||v||)g(x - v) \, dv \right.
\]

(36)

The kernel \( h(x) \) is therefore circularly symmetric and is identical in form to the 1-D kernel along any line through the origin. The matched filter kernel \( m(x) \) can be similarly defined, namely

\[
m(x) = x w(r)
\]

(37)

In the \( N - D \) case, however, the scale-space filter, which it may be recalled from Section 2.3 was the convolution of \( h(x) \) with \( m(x) \), is still a scalar filter, since convolution of \( m(x) \) with the vector field \( \eta(x) \) is defined by

\[
e(x) = \prod_{i=1}^{N} \int_{-\infty}^{\infty} m_i(v)\eta_i(x - v) \, dv
\]

(38)
where \( e(x) \) is again a scalar field. By linearity, equation (38) can be rewritten using equation (35) as

\[
e(x) = \int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} g(x - v) \times \left[ \sum_{i=1}^{N} \int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} m_i(u) h_i(v - u) \, du \right] \, dv \tag{39}
\]

where the function inside the square brackets is just the scalar kernel \( k(x) \), which may be written as

\[
k(x) = \nabla^2 w_i(|x|) \tag{40}
\]

the Laplacian of some window \( w_i(r) \) in \( N \) dimensions.

3. IMPLEMENTATION OF THE ESTIMATORS

Conversion of the estimators to the discrete lattices which tend to occur in practice is straightforward. The resulting algorithms are listed below in the same order as presented above.

3.1. Iterative algorithms

The implementation of the iterative estimator described above consists of the basic structure indicated below where \( g(n)(x) \) and \( \mu(n)(x) \) are respectively the data and local centroid at position \( x \) in some feature space on the \( n \)th iteration of the algorithm. \( L(n)(x) \) is the position of the mass originally at position \( x \).

\[
n := 0 \\
\text{Initialise } g(n)(x), \mu(n)(x) \\
\text{Converged := FALSE} \\
\text{WHILE NOT (Converged)} DO \\
\quad n := n + 1 \\
\quad \text{compute } \mu(n)(x) \text{ at all } x \\
\quad \text{update } g(n)(x), L(n)(x) \\
\quad \text{Converged := (} g(n)(x) = g(n-1)(x) \text{ at all } x) \\
\text{END} \\
\text{Classify the data} \\
\text{Note that the final classification of the data is performed using the final value of } L(n)(x). \text{Hence a point originally at position } x \text{ is assigned to position } x_c \text{ where:}
\]

\[
L^{(m)}(x) = x_c 	ag{41}
\]

\( m \) being the final value of \( n \).

If \( x \) is an \( N \)-dimensional vector, then \( \mu(x) \) and \( L(x) \) are themselves \( N \)-dimensional and the computation of each is a simple extension of the scalar case (cf Section 2.5). If the number of data vectors \( p \) is much less than the number of quantised positions in feature space, \( M^N (M \text{ being the number of intervals on each axis}) \) then the algorithm can be adjusted to take advantage of this sparseness by computing \( \mu(n)(x) \) only if \( g(n)(x) \neq 0 \).

For a given window radius \( r \), it is a simple matter to compute the relative efficiencies of the different approaches, recognising that the bulk of the computational effort is spent computing the local centroids of each point. For the case where no account is taken of any sparseness of the data, the complexity per iteration is given by:

\[
C_0 = O(M^N r^N). \tag{42}
\]

If there are only \( p \) non-zero points then

\[
C_1 = O(p^2 N) \tag{43}
\]

as, if the window is centred at some point \( x \), all the remaining \((p - 1) \) points must be visited to determine whether they lie within a distance \( r \) of \( x \). Since in general the number \( p \) of distinct points decreases with the iteration number, the overall computation time is greatly reduced.

The version of the algorithm employed in the 2-D test described in Section 4 is given below. It should be noted that for this test, the estimator kernel used was simply

\[
h(x) = -x \quad \|x\| \leq r \tag{44}
\]

In other words, the window \( w(r) \) was a simple rectangular one. In this test, only the internal consistency measure \( \rho(a) \) (equation 30) was used, while in the version employed in the segmentations described in references (4, 13), the external check (equation 31) was employed.

\[
r := r_{\text{min}} \\
good\_\text{classification} := \text{FALSE} \\
\text{WHILE NOT (good\_classification)} DO \\
\quad n := 0 \\
\quad \text{initialize } g(n)(x), L(n)(x) \\
\quad \text{Converged := FALSE} \\
\quad \text{WHILE NOT (converged)} DO \\
\quad\quad n := n + 1 \\
\quad\quad \text{FOR } i := 1 \text{ TO } p \text{ Compute } \mu(x_i) \\
\quad\quad \text{Update } g(n)(x), L(n)(x) \\
\quad\quad \text{Converged := (} g(n)(x) = g(n-1)(x) \text{ for all } x) \\
\quad \text{END} \\
\quad \text{Compute } \rho \text{ for each class found} \\
\quad \text{good\_classification := (} \rho > \text{Threshold}) \text{ for each class} \\
\quad \text{IF NOT (good\_classification)} THEN \\
\quad\quad r := 2r + 1 \\
\quad \text{END} \\
\text{END}
\]

3.2. Filtering algorithms

The matched filter employed in the test had the simple form

\[
m(x) = x/\|x\| \quad 0 < \|x\| < r \tag{45}
\]

This filter was convolved with the centroid estimate \( \mu(x) \) and the output normalised giving

\[
g_1(x) = - \sum_{\|x\| \leq r} \mu(x - v).v/\sum_{\|x\| \leq r} \|\mu(x - y)\| \tag{46}
\]

The signal \( g_1(x) \) was thresholded at a level of 0.9 and
the positions of the peaks of the thresholded signal used as the cluster centres. The window radius \( r \) was again increased by a factor of 2 until two successive radii were found satisfying the external consistency check of equation (31).

The linear filtering approach was based on the Laplacian pyramid described in reference (12). Thus, defining level 0 as the base of the pyramid and level \( N \) as the highest pyramid level, the signal \( g_p(i, j) \) at level \( p \) is given by:

\[
g_p(i, j) = g'_\rho(i, j) - E(g'_{p+1}(i, j)) \quad p > 0
\]
\[
g_N(i, j) = g_N(i, j)
\]

where \( g'_\rho(i, j) \) is the \( \rho \)th level of the Gaussian pyramid:

\[
g'_\rho(i, j) = R(g'_{\rho-1}(i, j)) \quad p > 0
\]
\[
g'_0(i, j) = h(i, j)
\]

\( h(i, j) \) being the original data. The symbols \( E \) and \( R \) represent expansion (interpolation) and reduction operators respectively:

\[
R(s(i, j)) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m, n) s(2i + m, 2j + n)
\]
\[
E(s(i, j)) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m, n) s((i+m/2, (j+n)/2)
\]

where \( w(m, n) \) is the generating kernel and where, in the case of the second equation, only those terms for which \((i+m)/2 \) and \((j+n)/2 \) are integers contribute to the summation. For this experiment, the generating kernel was taken to be symmetric and separable:

\[
w(m, n) = w(m)w(n)
\]
\[
w(m) = w(-m)
\]
\[
w(0) = 0.4, w(1) = 0.25, w(2) = 0.05.
\]

An event signal \( e_p(i, j) \) is defined at level \( p \) as follows:

\[
e_p(i, j) = 1 \text{ if } g_p(i, j) > g_p(i+i', j+j')
\]
\[-1 \leq i' \leq 1, -1 \leq j' \leq 1.
\]

Once again, consistency of the event signals across a pair of adjacent resolutions was used as the validation of the cluster centres. While both of the filtering algorithms can be based on a multiresolution pyramid, it is clear that the scalar approach is significantly faster than the vector filtering method, requiring \( O(r^N M^N) \) computations on an \( N - D \) problem defined on an \( M \)-point lattice, where \( r \) is the filter radius, as opposed to \( O(r^N M^N) \) for the vector filtering approach.

4. TESTING THE ESTIMATORS

4.1. A comparison of the three estimators

In order to compare the three approaches, a noisy 2-d test signal was generated using the three-cluster Gaussian signal

\[
g(x) = \sum_{i=1}^{3} \exp\left[-(x - \mu_i)^T(x - \mu_i)/\sigma^2\right]
\]

where \( \mu_1 = (40, 40)^T, \mu_2 = (80, 40)^T, \mu_3 = (80, 80)^T \) and \( \sigma = 20 \). White Gaussian noise \( \nu(x) \) with a standard deviation \( \sigma(x) = 0.3 g(x) \) was added to this signal to give the noisy 128 x 128 image illustrated in Fig. 1. Figures 2 and 3 illustrate the nonlinear and linear filtering results respectively, for a range of resolutions. Results for the three methods are compared in Table 1, where it can be seen that all three estimates are close to the actual centroids, with the scale-space method significantly worse on average than the other two, as shown by the standard deviation figures.

Note that the iterative algorithm, which converged in 9 iterations, requires a larger window radius than the filtering algorithms for successful clustering. At smaller window sizes, noise causes spurious clusters to be identified. On the other hand, if the window radius exceeds 40, then pairs of cluster centres are mutually "visible", again leading to significant errors. There is therefore always a limited range of scales over which any of the algorithms will perform adequately and as the level of noise increases this range will decrease. Similarly, as the cluster radius

<table>
<thead>
<tr>
<th>Method</th>
<th>Window radius</th>
<th>( \mu_1 )</th>
<th>( \mu_2 )</th>
<th>( \mu_3 )</th>
<th>s.d.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iterative</td>
<td>16</td>
<td>42.40</td>
<td>78.42</td>
<td>80.80</td>
<td>2</td>
</tr>
<tr>
<td>Nonlinear filter</td>
<td>8</td>
<td>41.40</td>
<td>78.42</td>
<td>80.79</td>
<td>1.7</td>
</tr>
<tr>
<td>Scale-space</td>
<td>8</td>
<td>34.82</td>
<td>38.41</td>
<td>78.75</td>
<td>12.3</td>
</tr>
</tbody>
</table>
(\sigma$, equation 53) increases, there must come a point where the clusters are no longer identifiable.

4.2. General performance

There are two aspects to examination of the performance of the local centroid algorithms. The first is to uncover any fundamental limitations on the algorithms, such as were hinted at above. The second is simply to illustrate that they are useful in a range of applications.

Perhaps the most important general limitation on these methods is also the most obvious: clusters which are too close together cannot be separated. Indeed, experience with these algorithms on a wide range of applications suggests a lower bound on the ratio $\rho$ of cluster radius $r_i$ (measured for example as standard deviation) and distance between cluster centres, $d_{ij}$, of the form

$$\rho_{\text{min}} = \min_{i,j} \left( \frac{d_{ij}}{r_i} \right) = 2.5.$$  \hspace{1cm} (54)

In other words, clustering can only be accomplished if the ratio $\rho$ exceeds 2.5. Moreover, this bound applies in the absence of noise in the data: the iterative algorithm in particular will fail to identify clusters if this criterion is not satisfied, even if the data are noiseless. As noise is added to the data, the smoothing necessary to counteract its effects inevitably tends to spread the clusters, increasing their radii and causing a concomitant increase in the minimum distance between centres. However, whereas the filtering procedures show, for small filter widths, a gain in signal-noise ratio which is proportional to the filter volume, the iterative algorithm seems to be more robust, especially in the presence of noise with a multiplicative effect, such as that used in the above experiment. This is illustrated in Figs 4 and 5, which show the results obtained for a 1-D two-cluster problem, in which the data were given by

$$g(x) = \sum_{i=1,2} \exp(-0.0025(x-\mu_i)^2)(1 + v(x))$$ \hspace{1cm} (55)

where $\mu_1 = 100$, $\mu_2 = 150$ and the noise process $v(x)$ was a sequence of independent, uniformly distributed random variables with standard deviations of 0.2 and 0.6 respectively. Note that the clustering performance of the iterative algorithm, as a function of window size, is virtually unaffected by noise of comparable magnitude to the signal. On the other hand, in additive noise, for a given filter radius there

Fig. 2. (a) Nonlinear filter output from data of Fig. 1, window radius 8.

Fig. 3. Scale-space filter outputs from Fig. 1 data, levels 1-5.

Fig. 2. (b) Result of nonlinear filtering on Fig. 1 data with radius 16.
is a threshold such that the clustering error varies slowly as a function of noise variance below the threshold and very rapidly over the threshold. This behaviour, which is qualitatively similar to that of a threshold detector has so far defied a full analysis, due to the nonlinearity of the iterative procedure. A qualitative explanation can be found by noting that the rapid increase in error is accompanied by a change in the number of classes. In other words, the noise has a significant probability of altering the cluster shape, to produce 2 or more clusters where the data only contain one.

This will happen if the noisy data look more like \( m \) clusters than 1 cluster, or in other words if the noisy data \( g(x) \)

\[
g(x) = f(x) + \nu(x)
\]

(56)

has a higher correlation with \( f_m(x) \), the representative signal for \( m \) clusters, than with \( f(x) \), namely

\[
\int_{-\infty}^{\infty} g(x)f_m(x) \, dx > \int_{-\infty}^{\infty} g(x)f(x) \, dx
\]

(57)

which amounts to the condition

\[
\int_{-\infty}^{\infty} \nu(x)(f_m(x) - f(x)) \, dx > \tau
\]

(58)

where \( \tau \) is some threshold value. Denoting by \( e_m \) the event that the threshold is exceeded, the probability with which \( e_m \) occurs for some \( m > 1 \) is just

\[
P_e = P \left( \bigcup_m e_m \right)
\]

(59)

which will exhibit a threshold behaviour as a function of the noise variance.

Table 2. Normalised error as a function of window size and noise variance

<table>
<thead>
<tr>
<th>Noise variance</th>
<th>0.0625</th>
<th>0.125</th>
<th>0.25</th>
<th>0.5</th>
<th>1</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Window radius</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.002</td>
<td>0.07</td>
<td>0.27</td>
<td>0.41</td>
<td>0.55</td>
<td>0.67</td>
<td>0.75</td>
</tr>
<tr>
<td>8</td>
<td>0.0001</td>
<td>0.0006</td>
<td>0.02</td>
<td>0.15</td>
<td>0.32</td>
<td>0.44</td>
<td>0.58</td>
</tr>
<tr>
<td>10</td>
<td>0.0001</td>
<td>0.0003</td>
<td>0.0005</td>
<td>0.02</td>
<td>0.13</td>
<td>0.26</td>
<td>0.39</td>
</tr>
<tr>
<td>12</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0004</td>
<td>0.0006</td>
<td>0.001</td>
<td>0.05</td>
<td>0.17</td>
</tr>
</tbody>
</table>
The plausibility of this explanation is illustrated in Table 2, which shows the error resulting from using the iterative algorithm with a rectangular window on a single 1-D cluster of radius 10, with varying amounts of additive uniform noise. The threshold effect is clearly visible in these results. Defining the threshold noise variance as that value for which the normalised error $P_r = 0.1$, an approximate relation between window radius and threshold noise variance $\sigma_r^2$ emerges, of the form

$$\sigma_r = 0.2r - 0.8 \quad r \geq 6$$  \hspace{1cm} (60)

which differs from the conventional threshold detector both in the existence of a threshold radius ($r = 6$) below which the algorithm always produces errors and in the approximately linear relation between radius and $\sigma_r$ above the threshold. This is presumably due to the iterative nature of the algorithm, which results in an effective radius which is significantly larger than the radius of the filter.

Thus while the iterative algorithm requires a minimum radius for successful operation, its sensitivity to noise is significantly lower than that of the more conventional algorithms based on peak detection.

A further indication of the performance of the algorithms is their success or failure in applications. The iterative form of the algorithm has been used in a wide variety of applications, which are illustrated in Figs 6--8. Figure 6 shows the use of a 2-D form in a line-thinning procedure, applied to the output of an edge detection process. After thinning, the image was thresholded, resulting in much better retention of weak or slowly varying image features than is the case if the edge image is thresholded directly. Figure 7 shows the effect of applying the 1-D algorithm to the gray level histogram of the image, at a single resolution, leading to a total of 11 clusters. Finally, Fig. 8 illustrates the use of the multiresolution algorithm in a 13 dimensional texture feature space, which formed a central component of an image texture segmentation algorithm. Full descriptions of these applications are contained in the references.

Finally, in order to test the limitations of the algorithm, it was run on the non-convex cluster pair illustrated in Fig. 9. The resulting cluster centres are shown in bold print, the cluster masses being indicated by the size of the blocks. This example also illustrates a point which the reader may have missed—these convolutional algorithms work just as well on clusters consisting of point masses, despite their having been derived on the assumption of continuity of $f(x)$ (Section 2).

5. CONCLUSIONS

This paper has described an attempt to put the mysterious art of clustering on a foundation of esti-
New approach to clustering

Fig. 9. Clustering of non-convex data set using iterative algorithm.

In scale between a global centroid estimator of cluster centres and local peak (mode) detector. A heuristic derivation of the estimator is supplemented by a derivation based on Wiener filtering. It is shown that the new estimator can be seen either as a mode detector operating on smoothed data or as a localised, windowed form of centroid estimator. In other words, by varying the spatial scale of the new estimator, it is possible to pass continuously from a completely global centroid estimate to a completely local mode estimate. The importance of spatial scale is emphasised by introducing a multiresolution approach to the problem. It is shown that this leads to new tests of cluster validity, which can be used to enhance the robustness of the method to noise and to variations of spatial scale. Iterative and non-iterative implementations of the new estimator are presented and shown to be more accurate than a scale-space filtering approach in locating the cluster centres of some noise 2-D data. Results obtained with the iterative algorithm on practical applications including line thinning and segmentation are used to illustrate the potential of these methods in practice.

Acknowledgement—This work was partly supported by the UK SERC and IBM (UK) Ltd.

REFERENCES


**APPENDIX**

1. **DERIVATION OF THE OPTIMUM ESTIMATOR**

The noisy data are given by

\[ g(x) = g_0(x) + \nu(x) \quad (A.1) \]

where \( \nu(x) \) is a sample from an independent white normal noise process and

\[ g_0(x) = \sum_i a_i f(x - x_i) \quad (A.2) \]

with \( a_i > 0 \) chosen independently under a density \( p(a) \) and \( x_i \) selected by a Poisson process of intensity \( \lambda \) (see for example reference 7).

The estimation problem is to find the linear estimator \( \hat{g}(x) \) which minimises the joint probability

\[ \text{Prob} \{ g_0(x) \hat{g}(x) > 0 \} \quad \forall x \quad (A.3) \]

that the derivative \( g'(x) \) of the noiseless data and the estimate have the same sign. This is equivalent to demanding that their zero-crossings are as close as possible.

By central limit arguments, \( g_0(x) \) is the sum of an indefinite number of independently chosen random variables and will therefore have a normal density. Since \( g_0(x) \) is a linear function of the sum of \( g_0(x) \) and an independent normal process, the joint density of \( g_0(x) \) and \( \hat{g}(x) \) will be normal. But it then follows from the arcsine law\(^7\) that

\[ \text{Prob} \{ g_0(x) \hat{g}(x) > 0 \} = \frac{2}{\pi} \arcsin \left[ \frac{Eg_0(x)\hat{g}(x)\sqrt{Eg_0(x)Eg_0(x)}}{Eg_0(x)Eg_0(x)} \right] \quad (A.4) \]

where \( E \) denotes expectation. In other words, the estimator \( \hat{g}(x) \), whose correlation with \( g_0(x) \) is greatest is the optimum estimator. This is, however, the Wiener estimator \( \hat{g}(x) \), defined by the orthogonality principle\(^7\)

\[ E\hat{g}(y)g(y - x) = \int_{-\infty}^{\infty} h(y)Eg_0(g(z - x + y) dy \quad (A.5) \]

where \( h(y) \) is the estimation kernel. Equivalently, \( h(.) \) can be defined in terms of its Fourier transform \( H(\omega) \)

\[ S_{\nu}(\omega) = A^2|\hat{f}(\omega)|^2 \quad (A.6) \]

\[ H(\omega) = \frac{S_{\nu}(\omega)}{S_{\hat{g}}(\omega)} \quad (A.7) \]

Now by the differentiation theorem of the Fourier transform,\(^14\) the kernel \( h(x) \) can be written as

\[ h(x) = w'(x) \quad (A.10) \]

where \( w(x) \) has Fourier transform \( W(\omega) \)

\[ W(\omega) = \frac{\lambda A^4|\hat{f}(\omega)|^2}{1 + \lambda A^4|\hat{f}(\omega)|^2} \quad (A.11) \]

It is easy to see that \( W(\omega) \) is a positive real even function of \( \omega \) which has a maximum at \( \omega = 0 \) because \( \hat{f}(x) \neq 0 \). In other words \( w(x) \) represents a low-pass filter and, by the symmetry theorem of the Fourier transform, is real and even. It also has a maximum at \( x = 0 \), and is continuous, since \( W(\omega) \) is positive.\(^14\) It follows that if \( w'(x) \) is truncated to form \( w'(x) \) defined by

\[ w'(x) = w'(x) \quad |x| < X \]

\[ = 0 \quad \text{else} \quad (A.12) \]

where \( X \) is chosen so that \( w(x) \equiv 0 \), \( 0 < x < X \), then an estimation kernel of the form of equations (9) and (13) results.

2. **RELATION BETWEEN THE ESTIMATION KERNELS**

The centroid kernel can be expressed as

\[ h(x) = -xw_1(x) \quad (A.13) \]

which has an integral

\[ w(x) = -\int_{-\infty}^{\infty} yw_1(y) dy \quad (A.14) \]

where \( w_1(x) = 0 \) if \( |x| > X \). Now, since \( w_1(y) \) is continuous, so is the integral and since it is bounded, so is \( w(x) \).
Clearly \( w(x) \) is even, for

\[
\begin{align*}
w(-x) &= - \int_{-x}^{-v} w(y) dy + \int_{v}^{x} w(y) dy \\
&= w(x)
\end{align*}
\] (A.15)

and since \( w(x) = 0 \) if \( |x| > X \), then \( w(x) \) must also be of finite support

\[
w(x) = 0 \quad |x| > X
\] (A.16)

Finally, since \( w_t(y) > 0 \),

\[
w'(x) = -xw_t(x) \begin{cases} > 0 & \text{if } x < 0 \\ < 0 & \text{if } x > 0 \end{cases}
\] (A.17)

and so \( w(x) \) has a single peak at 0 and is positive. Thus \( w(x) \) satisfies 2.1.1–2.1.4.

The final element of the relation between the two estimators is the proof that the derivative of \( g(x) \) in the noiseless case is the limit of the windowed centroid estimate. The windowed centroid, with appropriate magnitude scaling, can be written in the form

\[
\begin{align*}
\eta_n(x) &= a^2 \beta \int_{-x}^{x} y w'(ay) g(x + y) dy \\
&= \beta g'(x) \int_{-x}^{x} y^2 w(y) dy + \alpha^2 \beta \frac{g'''(x)}{3!} \int_{-x}^{x} y^3 w(y) dy + \ldots
\end{align*}
\] (A.18)

where \( w(x) = 0, |x| > X \), as above. For \( a \) large enough, \( g(x + y) \) can be replaced by its Maclaurin expansion about \( x \), giving

\[
\eta_n(x) = a^2 \beta \int_{-x}^{x} y w'(ay)(g(x) + yg'(x) + \frac{y^2}{2!}g''(x) + \ldots) dy.
\] (A.19)

Now all terms in the series which contain even powers of \( y \) contribute 0 to the integral because \( yw(ay) \) is an odd function, giving

\[
\eta_n(x) = \beta g'(x) \int_{-x}^{x} y^2 w(y) dy + \alpha^2 \beta \frac{g'''(x)}{3!} \int_{-x}^{x} y^3 w(y) dy + \ldots
\] (A.20)

finite support

\[
w(x) = 0 \quad |x| > X
\] (A.16)

Finally, since \( w_t(y) > 0 \),

\[
w'(x) = -xw_t(x) \begin{cases} > 0 & \text{if } x < 0 \\ < 0 & \text{if } x > 0 \end{cases}
\] (A.17)

and so \( w(x) \) has a single peak at 0 and is positive. Thus \( w(x) \) satisfies 2.1.1–2.1.4.

The final element of the relation between the two estimators is the proof that the derivative of \( g(x) \) in the noiseless case is the limit of the windowed centroid estimate. The windowed centroid, with appropriate magnitude scaling, can be written in the form

\[
\eta_n(x) = a^2 \beta \int_{-x}^{x} y w'(ay) g(x + y) dy
\] (A.18)

where \( w(x) = 0, |x| > X \), as above. For \( a \) large enough, \( g(x + y) \) can be replaced by its Maclaurin expansion about \( x \), giving

\[
\lim_{x \to \pm \infty} \eta_n(x) = g'(x) \int_{-x}^{x} y^2 w(y) dy
\] (A.21)

and setting \( \beta^{-1} = \int_{-x}^{x} y w(y) dy \) gives the required result.

As a corollary, note that if \( \delta > 0 \), there exists a scale parameter \( a \) such that

\[
\delta > 2a^{-2} \beta \left[ \frac{g'''(x)}{3!} \int_{-x}^{x} y^3 w(y) dy + \ldots \right] \quad \forall x
\] (A.22)

but then

\[
|\eta_n(x) - \eta_t(x)| = (a^{-2} + \alpha^{-2}) \beta \left[ \frac{g'''(x)}{3!} \int_{-x}^{x} y^3 w(y) dy + \ldots \right] < \delta
\] (A.23)

Thus for any \( \delta > 0 \), there is some \( a \) for which A.23 holds for all \( a \).