



# SpaRef: a clustering algorithm for multispectral images

Thanh N. Tran, Ron Wehrens, Lutgarde M.C. Buydens\*

*Laboratory of Analytical Chemistry, University of Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands*

Accepted 4 June 2003

## Abstract

Multispectral images such as multispectral chemical images or multispectral satellite images provide detailed data with information in both the spatial and spectral domains. Many segmentation methods for multispectral images are based on a per-pixel classification, which uses only spectral information and ignores spatial information. A clustering algorithm based on both spectral and spatial information would produce better results.

In this work, spatial refinement clustering (SpaRef), a new clustering algorithm for multispectral images is presented. Spatial information is integrated with partitional and agglomeration clustering processes. The number of clusters is automatically identified. SpaRef is compared with a set of well-known clustering methods on compact airborne spectrographic imager (CASI) over an area in the Klompenwaard, The Netherlands. The clusters obtained show improved results. Applying SpaRef to multispectral chemical images would be a straight-forward step.

© 2003 Elsevier B.V. All rights reserved.

*Keywords:* Clustering algorithm; Multispectral image segmentation; Spatial information

## 1. Introduction

Clustering is the organisation of a data set into homogenous and/or well separated groups with respect to a distance or, equivalently, a similarity measure. Its objective is to assign to the same cluster data that are more close (similar) to each other than they are in different clusters [1]. In multispectral satellite images, organising the data pixels into classes, also called image segmentation, can reveal the underlying structure of the images, i.e. spectrally homogeneous characteristics. This information can be used in a number of ways, e.g. to obtain optimum information for the selection of training regions for subsequent supervised land-use

segmentation [2]. In vegetation areas, the gradient may change very slowly from one vegetation type to another. This makes it very difficult to identify a border between clusters, leading to clusters scattered in the spatial domain, which makes interpretation very difficult. This is also true for multispectral chemical images. What is needed is a clustering method that takes both spectral and spatial information into account.

Clustering methods fall into two types: partitional and hierarchical approaches [1]. Variants of  $K$ -clustering, such as  $K$ -means, ISODATA [3], and Fuzzy  $c$ -means [1], are the partitional clustering methods that are most widely used for satellite images.  $K$ -clustering is computationally attractive, which makes it applicable for large data sets, but it is very sensitive to small clusters and outliers, i.e. noise or mixed pixels (pixels containing information from two or more classes) [4]. Agglomerative hierarchical clustering (AHC) works well with small data sets

\* Corresponding author. Tel.: +31-24-3653192;

fax: +31-24-3652653.

*E-mail addresses:* [tnthanh@sci.kun.nl](mailto:tnthanh@sci.kun.nl) (T.N. Tran),

[lbuydens@sci.kun.nl](mailto:lbuydens@sci.kun.nl) (L.M.C. Buydens).

and can handle outliers very well but its computation is very expensive and therefore it is not feasible for a large data set. Moreover, it also has a ‘chaining’ problem for a complex data set [5]. In several papers, these clustering methods are compared [2,6] but the fundamental problems remain. In other research, agglomerative hierarchical clustering is performed on a number of homogenous classes with an assumption of uniform neighbourhoods in the dataset in order to avoid the limitations of agglomerative hierarchical clustering, which is not true in general cases [7].

In this study,  $K$ -clustering and agglomerative hierarchical clustering are analysed. Their advantages as well as limitations are illustrated. A new clustering algorithm, spatial refinement clustering (SpaRef), is designed to take advantage of the characteristics of both clustering methods and eliminate their potential limitations. SpaRef can work with a complex and large dataset, including small objects and outliers. Briefly, SpaRef method works as follows. First, a high number of small, homogeneous clusters are identified by  $K$ -means. These so-called cells are clustered using agglomerative hierarchical clustering and the optimal number of clusters is identified based on the ratio of the within- and between-cluster variation. Our main contribution, the refinement process, is introduced at the last stage. It reallocates mis-assigned points using the information of points in the spatial domain.

First, we will discuss relevant characteristics of  $K$ -clustering and hierarchical clustering methods in more detail. Then, we will discuss several ways to pick the optimal number of clusters and to validate the results of an image segmentation. We proceed by describing on SpaRef method in more detail, and apply it to a real-world multispectral image.

SpaRef is compared with  $K$ -means, ISODATA and a hierarchical clustering and shows better results.

## 2. Notation

We will consider an image consisting of  $N$  pixels, where each pixels is characterised by  $D_{\text{im}}$  variables (reflectance values).

We will use the following notation

- $K$  is the number of clusters and  $k$  is the index of the cluster.

- $M$  is the number of cells, clusters of a high homogeneity,  $M \ll N$ .
- $\text{Minsize}$  is a minimum size of a normal cluster (so the clusters should contain at least  $\text{Minsize}$  pixels).
- $B_c$  is number of boundary points of cluster  $c$  in the spatial domain. A boundary point of cluster  $c$  is defined as the point which has at least one adjacent point belonging to another cluster  $d$  ( $d \neq c$ ).
- $C_k$  is a set of point indices that belong to cluster  $k$ .

$$c_k = \frac{1}{n_k} \sum_{i \in C_k} x_i \quad (1)$$

is the centre of the  $k$ th cluster, in spectral space.

$$c = \frac{1}{N} \sum_{i=1}^N x_i = \frac{1}{N} \sum_{k=1}^K n_k c_k \quad (2)$$

is the mean centre of the entire data set, in spectral space.

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^d (x_{il} - x_{jl})^2} = \|x_i - x_j\| \quad (3)$$

is the Euclidean distance of two points,  $x_i$  and  $x_j$ .

$$W_k = \frac{1}{n_k} \sum_{i \in C_k} d(x_i, c_k) \quad (4)$$

is within-cluster inertia of class  $k$

$$B_{kj} = d(c_k, c_j) \quad (5)$$

is between-cluster inertia of classes  $k$  and  $j$ .

### 2.1. $K$ -clustering

$K$ -means and ISODATA [8] are among the most popular, well-known ‘hard’ partitional clustering algorithms, in which each point is assigned to only one particular cluster.  $K$ -means produces a clustering by optimising the sum-of-squares criterion,  $E$

$$E = \sum_K \sum_{i \in C_k} d^2(x_i, c_k) \quad (6)$$

The algorithm addresses directly the problem of dividing a set of data into several homogeneous groups. For a given number of  $K$  clusters, the algorithm starts by choosing  $K$  cluster centres (randomly or by some heuristic process) [8]. The Euclidean

distances between all points and the cluster centres are calculated. Points will be assigned to the closest cluster centre. Cluster centres are recalculated and the process is repeated unless a convergence criterion is met. A major disadvantage of  $K$ -means clustering is that one must specify the number of clusters  $K$  in advance. Moreover, the algorithm is very sensitive to noise, mixed pixels and outliers in the data set [4], all situations that occur frequently with satellite images. Furthermore, the algorithm easily gets stuck in a local optimum on the sum-of-square error space. For these reasons, the  $K$ -means clustering results are not stable, i.e. they heavily depend on different choices of the initial cluster centres.

ISODATA [3] is a modification of  $K$ -means that starts with a high number of clusters and permits splitting of clusters when a cluster variance is above a pre-specified threshold or merges them when distances between clusters are small, below another threshold. Starting with a higher number of clusters, ISODATA is more stable, but the algorithm requires many input parameters that can be difficult to find.

Fuzzy  $c$ -means [1], on the other hand, is a 'soft' partitional  $K$ -clustering which attempts to assign each point  $x_i$  to several clusters, depending on the degree of the fuzzy membership,  $u_{ik} \in [0, 1]$ , in order to optimise the sum-of-squares criterion,  $E_f$

$$E_f = \sum_K \sum_{i \in C_k} u_{ik} d^2(x_i, c_k) \quad (6b)$$

The algorithm works similar to  $K$ -means. In most cases, if one has no interest in a fuzzy membership, then Fuzzy  $c$ -means result—the membership matrix  $U$ —will be converted to a hard membership matrix by thresholding the fuzzy membership value, which is similar to a hard clustering result.

## 2.2. Agglomerative hierarchical clustering

Agglomerative hierarchical clustering yields a hierarchical structure of clusters, representing how cluster pairs are joined. In principle, the algorithm starts with assigning each pixel to individual clusters. At each iterative step, the proximity matrix is calculated for all cluster pairs and the two 'closest' pair clusters are merged. The process will continue until there is only one cluster.

Depending on the definition of a distance between clusters, agglomerative hierarchical clustering are variants of single linkage [9], complete linkage [10], average linkage and Ward's [11] algorithms. In single linkage, the distance of two clusters is the distance between two nearest points. Similarly, the distance is the maximal distance between points in different clusters in complete linkage, and the average distance of points in average linkage clustering. The distance in Ward's method is defined as the squared Euclidean distance of the cluster mean vectors. Hence, Ward's method is related to  $K$ -means through the minimum-variance criterion. In this paper, agglomerative hierarchical clustering with Ward's distance measure is used.

A dendrogram is produced, representing nested clusters and the similarity levels at which clusters are joined. The dendrogram can be cut at several levels in order to obtain an arbitrary number of clusters. It circumvents the problem of the pre-defined number of  $K$  clusters in  $K$ -clustering algorithms. By starting with assigning each pixel to individual clusters the algorithm is not sensitive to outliers [5]: outliers will be kept in separate clusters, not influencing the other clusters.

Overall, agglomerative hierarchical clustering considers only clusters that were obtained in the previous step. This means that once a point has been merged to a cluster, it cannot be considered for joining another cluster in later iterations. This rule is not optimal for complex data sets where cluster homogeneity levels are low or not uniform [5].

The algorithm requires calculation, storage and sorting of the proximity matrix a maximum size of  $N^2$ . If  $N$  is large then this matrix becomes huge and sometime it is not feasible [5].

## 2.3. Number of clusters

Determining the number of clusters is a difficult problem in all clustering algorithms. Many criteria have been developed [12,13] often based on measures of spread within- and between-clusters. The within-cluster inertia,  $W$ , is defined as variation of individual points to their centre and the between-cluster inertia,  $B$ , is defined as the variation of cluster centres around the overall mean.

$$W = \frac{1}{N} \sum_K \sum_{i \in C_k} d(x_i, c_k) \quad (7)$$

$$B = \frac{1}{N} \sum_K n_k d(c_k, c) \quad (8)$$

Clustering algorithms minimising the sum-of-squares criterion (Eq. (6)) would thus minimise  $W$ . By keeping track of the within-cluster inertia (or other criteria based on it) for a varying number of clusters, one can often observe a sharp increase at a certain level. Just before this increase, the spread of the clusters is minimal and then the optimal number of clusters can be found.

Many criteria [12,13] in one way or another illustrate this situation [4], for example, by minimum Duun or Davies–Bouldin indices [13,14]. Duun and generalised Duun indices are also used in some cases [14] but they are very computation-expensive and not suitable for a dataset with large number of points [15]. The Davies–Bouldin index is a function  $\bar{R}$  of within-cluster scatter and between-cluster separation

$$R_k = \max_{j \neq k} \left\{ \frac{W_k + W_j}{B_{kj}} \right\} \quad (9)$$

$$\bar{R} = \frac{1}{K} \sum_{k=1}^K R_k \quad (10)$$

Here, we simply use the ratio of within-cluster to between-cluster inertia,  $I$ , to determine the optimal number of clusters where there is a sharp change at a certain level

$$I = \frac{W}{B} \quad (11)$$

Using this ratio allows us to see the change in homogeneity more clearly and it is not dependent on a particular clustering algorithm.

#### 2.4. Cluster validity

It is notoriously difficult to assess the results of clustering algorithms in remote sensing. Usually qualitative, subjective criteria are applied, such as the homogeneity in the spectral domain (compactness) of the segments, and the degree of fragmentation (dispersion) of the segments in spatial domain. The index function  $I$  and Davies–Bouldin index can also be used for cluster validation. Small values of these indices correspond with better results. For validating

a clustering result in terms of dispersion of points in the spatial domain, we introduce  $D_c$ , a dispersion index for cluster  $c$ , to be the ratio of the number of boundary points of cluster  $c$ ,  $B_c$ , to the total number of points of cluster  $c$ ,  $n_c$ . A boundary point of cluster  $c$  is defined as a point where at least one of its adjacent points belongs to another cluster  $d$  ( $d \neq c$ ).

$$D_c = \frac{B_c}{n_c} \quad (12)$$

$D$ , the average dispersion degree over the image, is equal to the ratio of the total number of boundary points to the total number of points of the image. A fuzzy image will have a higher dispersion degree than an image containing large continuous areas with sharp straight edges.

$$D = \frac{1}{N} \sum_{c=1}^K B_c \quad (13)$$

### 3. Description of SpaRef

SpaRef is designed to use a combination of  $K$ -clustering and agglomerative hierarchical clustering to take advantage of the characteristics of both clustering methods and eliminate their potential limitations by introducing a refinement process using spatial information.

In order to prevent the (expensive) application of AHC to a large data set, SpaRef is first pre-processed by  $K$ -means with a high number of classes,  $M$ . When  $M$  is high enough, clusters can be considered as highly homogenous classes. These form the input to the agglomerative hierarchical process. The number of classes  $M$  is much smaller than the total number of points  $N$ , typically in the order of 100.

Determining the number of clusters in a data set by using the index function  $I$  (Eq. (11)) is very time consuming with  $K$ -clustering, where the algorithm has to be run for each number of clusters  $K$ . On the other hand, it is much easier for agglomerative hierarchical clustering, where we can calculate the index function at each merge level, which is used in SpaRef. For each level in the dendrogram, the clustering index  $I$  is calculated and the ‘best’ choice of  $K$  number of clusters thus is identified where there is a sharp change at the level  $K$ .

In a data set containing also noise, mixed pixels or outliers, we often find a number of very small clusters with abnormal cluster sizes, the set  $S$ , which are well separated from normal clusters by the threshold,  $M_{\text{minsize}}$ . They are ‘stable’, isolated and highly homogenous [16]. They may contain noise, mixed pixels, outliers and very small objects. Noise pixels must be rejected, mixed pixels have to be considered to merge to the spatially ‘closest’ neighbour cluster and small objects may be identified using a priori information. How to discriminate between the different types of small classes is the subject of further study. Here, we will remove these classes from the data set and concentrate on the larger clusters.

Let  $O$  to be the set of other clusters,  $K \setminus S$ . These clusters are large, probably less well separated and quite disperse. Agglomerative hierarchical clustering may have problems separating these clusters, because of the lack of flexibility imposed by the hierarchical structure. To deal with this problem, we introduce a refinement process to all boundary points of the clusters in spatial domain. We assume that if there are mis-assigned points in clusters, they would first appear in the boundaries of clusters. Therefore, boundary points will be re-assigned to the ‘closest’ adjacent clusters, and cluster boundaries will be redrawn. The refinement process iterates until there is no more change in border point classification. This leads to a smoothing on the spatial domain, while still keeping in mind the information from the spectral domain.

The flowchart of SpaRef is given in Fig. 1.

SpaRef alleviates the inflexibility of agglomerative hierarchical clustering. By limiting the refinement only to boundary points, the clustering is expected to have a high continuity. At any iteration, let  $x_i$  be point in cluster  $S_c$  but not a border point. Even if there exists a cluster  $d$  such that  $d(x_i, c_d) < d(x_i, c_c)$  then  $x_i$  is not considered to be re-assigned to cluster  $d$ . It will only be joined to cluster  $d$  when it is at the boundary of cluster  $c$ . Therefore, SpaRef is fast, since only a limited number of reallocations have to be considered.

SpaRef depends on two main input parameters,  $M$  and  $M_{\text{minsize}}$ .  $M$ , the number of cells, is dependent on the image type. Images with a higher degree of complexity would require a higher setting for  $M$ . The main purposes of defining the number of cells  $M$  are to separate noise, mixed pixel class and small objects,

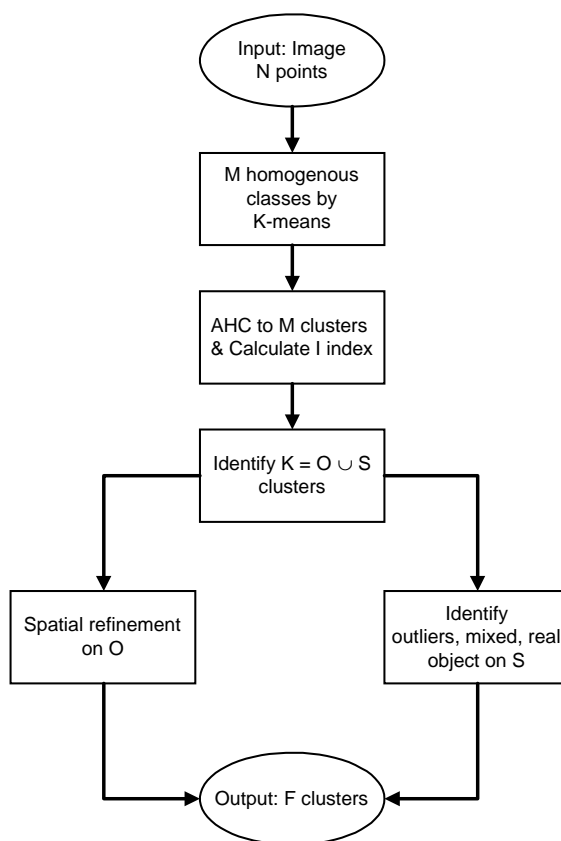


Fig. 1. Flow chart of SpaRef method.

and stabilise clustering result. Therefore, with a ‘high enough’ setting of  $M$ , the clustering method will not be significantly affected by the exact setting. In most cases, after the AHC merging stage, very small clusters will be well separated from normal and large clusters. The setting for  $M_{\text{minsize}}$  is thus easily defined, in practice.

The total complexity of SpaRef is equal to  $O(M \log M) + O(M^2)$ . For a large dataset, when the number of points  $N$  is big, the complexity of SpaRef is much less than  $O(N^2)$  as with AHC.

#### 4. Software

Software has been developed using C (GCC) in SunOS operating system. Pre- and post-processing of the image is done in Matlab. MultiSpec (©Purdue

Research Foundation), a multispectral image data analysis system [17], and ERDAS IMAGINE product [18] are used for image manipulation and clustering comparison.

## 5. Segmentation experiments

### 5.1. Data

As an example, we will use a multispectral satellite image recorded by a compact airborne spectrographic imager (CASI) scanner from the Natural Environment Research Council (NERC) that was taken at 1536 m over an area in the Klompenwaard, The Netherlands during August 2001. The CASI has provided 10 bands for this study from 437 to 890 nm, with bandwidths of 10 nm, except for band 9 with 8 nm. The area has size of  $211 \times 301$  pixels leading to 63511 pixels with 3 m resolution covering  $633 \times 903 \text{ m}^2$  (Fig. 2). The original multispectral data were mean centred and compressed via a principal components analysis in order to reduce computation time. The clustering methods were all performed on the first four principal components, which account for more than 99.8% of the spectral variance. Next, the application of four clustering methods to these data will be described. The methods are SpaRef, *K*-means, ISODATA and Ward's clustering.

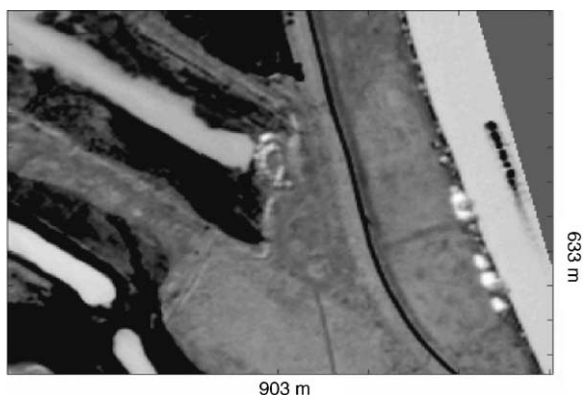


Fig. 2. Gray-scale image of the first principal component (PC1) of the Klompenwaard image, obtained with CASI scanner on 15 August 2001. The white areas are corresponding to parts of the Klompenwaard river and small lake containing water. Dike and road can be identified as the less gray region along the Klompenwaard river. Vegetation and forest areas can be seen elsewhere as lower dark and higher dark areas, respectively.

### 5.2. Application of SpaRef

*M* is set to 300. The *K*-means clustering was first applied to the images to obtain 300 classes (cells). The agglomerative hierarchical clustering of 300 classes was then continued and the index function was calculated. We present in Fig. 3 the plot of the index function over the number of classes. The figure shows the location of the 'best' choice of number of clusters to be 39, where there is a sharp change. This data set is expected to contain also noises, mixed pixels, and hence, Min-size is set to 100. Otherwise, Min-size is zero. Fourteen clusters with sizes smaller than 100 pixels, containing in total 765 points (1%), have been rejected (Fig. 4). Indeed, by comparison with ground-truth information, those points are shadow areas, small objects (buildings, structures of a boat, etc.). The remaining 25 normal classes with 62,746 points, 99% of points are subjected to the refinement process as described earlier.

### 5.3. Application of *K*-means

*K*-means is sensitive the choice of initial centre points, so that we performed *K*-means 100 times with random initialisation. The non-compactness *I*, dispersion and Davies–Bouldin indices are illustrated in Fig. 5. Clearly, for the 100 runs, the variability in all three indices is quite large.

### 5.4. Application of ISODATA

The data set has been also clustered by ISODATA algorithm [Fig. 6a] which is implemented in Multi-Spec software, a multispectral image data analysis system for interactively analysing Earth observational multispectral image [17]. With the prior information about the number of clusters and maximum cluster size, in order to find settings leading to 25 clusters, a trial-and-error strategy has been applied. A 'good' setting of convergence, a stop-criterion, is 99%. The algorithm is more accurate but takes more computation time if the stop-criterion is high. The minimum cluster size is 10, the distance threshold used in deciding whether two clusters should be merged is 990, and the threshold determining if a cluster should be split is 2000. The number of clusters would not be 25 otherwise. Lower split-threshold or lower distance

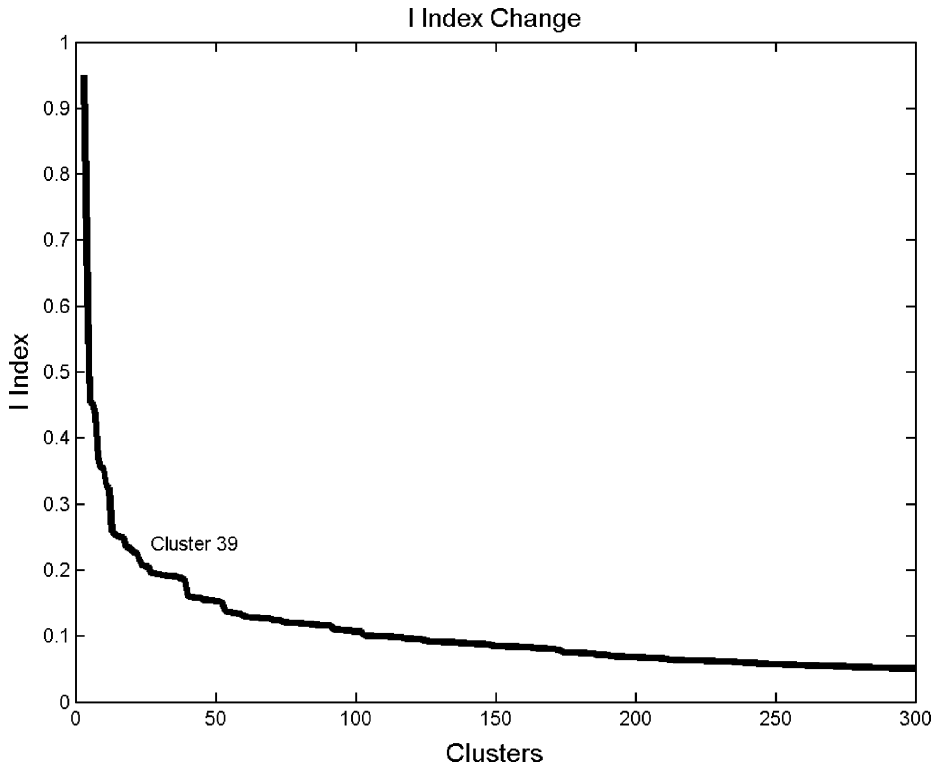


Fig. 3. I index changes while applying AHC to 300 homogenous clusters. The optimal number of clusters is identified to be 39 where there is a sharp change.

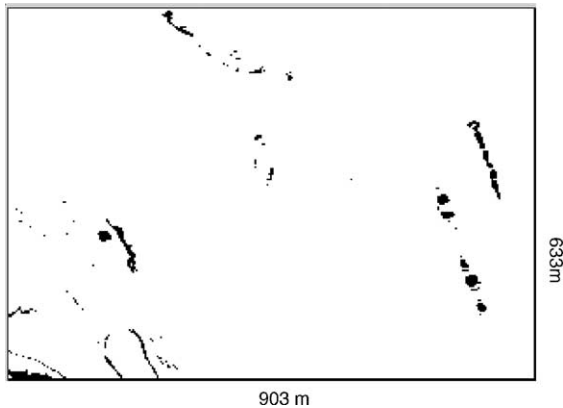


Fig. 4. Unclassified points in 14 very small classes (765 points, 1% of total points). Those points are shadow areas, small objects (buildings, structures of a boat, etc.).

threshold leads to more clusters. It is very difficult to find good settings for ISODATA algorithm without prior information about the data set. This is also the main limitation of ISODATA.

#### 5.5. Application of Ward's clustering

For convenience, the process of agglomerative merging of 300 cells instead of individual pixels is considering as the modification of Ward's method (*M*-Ward). This is actually our method without the refinement process. The modification of Ward's method is expected to have slightly lower values of non-compactness *I* and Davies–Bouldin indices and a slightly higher value of the dispersion *D* index, compared to the original Ward method. This difference is not significant when *M* is high enough.

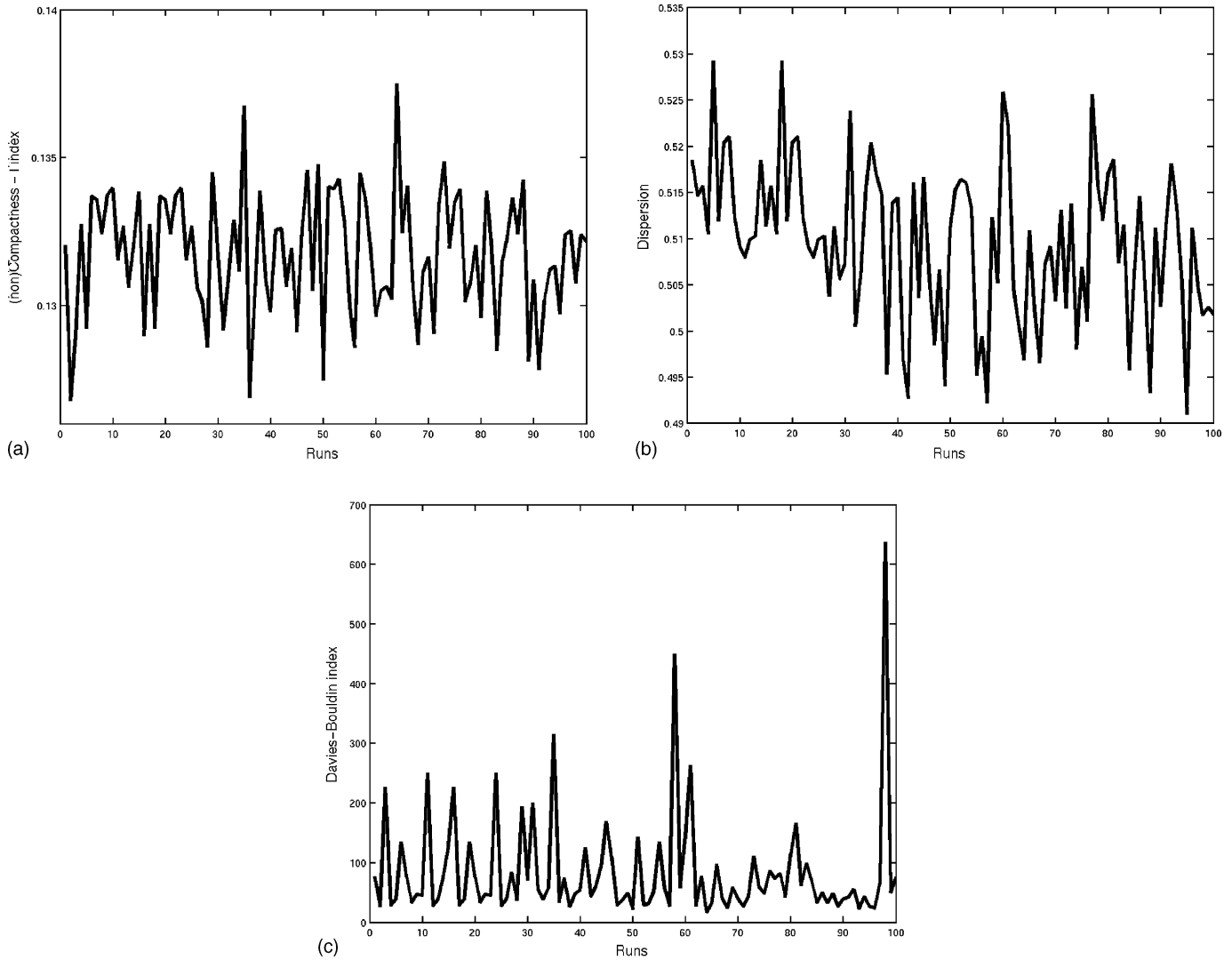


Fig. 5. Indices of *K*-means for 25 clusters for 100 runs: (a) non-compactness I index; (b) dispersion index; (c) Davies–Bouldin index.

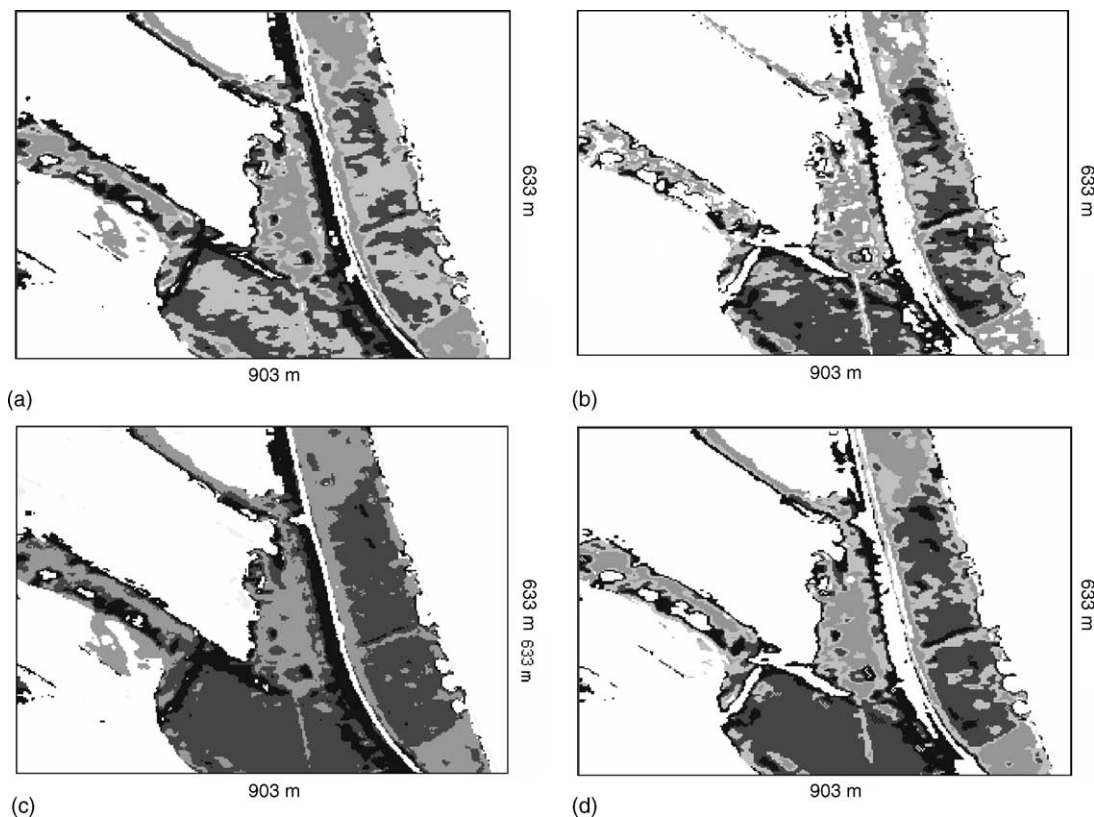


Fig. 6. Four clusters of clustering results: (a) by ISODATA; (b) by *K*-means; (c) by Ward; and (d) by SpaRef. The colour 'white' in (A–D) signifies 'others clusters'; there are only four shades of gray in (A).

## 5.6. Results

Four clusters for ISODATA, the first of 100 runs of *K*-means and SpaRef, and three clusters for *M*-Ward method, in total covering roughly the same area, are chosen from the clustering results in order to present the results in gray-scale image. These clusters are shown in Fig. 6(a–d) for ISODATA, the first run of *K*-means, *M*-Ward and SpaRef, respectively. In all cases, small clusters are excluded. By this setup, ISODATA and *K*-means have inherited the advantage of not considering small classes and noise, which would otherwise degrade performance.

We compare clustering results of SpaRef to *K*-means, ISODATA and *M*-Ward clustering using I, Davies–Bouldin and dispersion indices.

Table 1 shows non-compactness I, dispersion *D* and Davies–Bouldin indices of different methods. In

Table 1  
Validity indices for each clustering methods (*K*-means, ISODATA, Ward and SpaRef)

Methods	I index	Davies–Bouldin	Dispersion index
<i>K</i> -means (100 runs)			
Minimum	0.1268	16.3289	0.4910
Average	0.1317	83.4246	0.5095
Maximum	0.1375	636.4714	0.5292
ISODATA	0.1318	46.3295	0.4750
Ward	0.1606	48.6012	0.3811
SpaRef	0.1295	27.8711	0.4595

All indices would be as low as possible. SpaRef shows a very good performance on all three criteria simultaneously. Note that the minimal values for *K*-means in the different columns are caused by different clusterings.

non-compactness I index, ISODATA leads to comparable values with the average value obtained from 100 runs *K*-means. *M*-Ward clustering, with the highest I degree, is worse than any clustering obtained with *K*-means [19]. The response from SpaRef is comparable to the best case obtained from *K*-means. The table also shows that Davies–Bouldin index gives the same scenario as the I index.

In the dispersion index, *D*, *K*-Ward method gives the lowest value (the highest continuity degree). It is because of the ‘nearest neighbourhood rule’ affected on the spatial domain and it may thus be lower than the expected value of a ‘true’ response of the *D* index. *K*-means obtains bad responses in all cases. ISODATA gives better result than *K*-means. Lastly, SpaRef obtains a lower *D* index than ISODATA and *K*-means. It is higher than the response from *K*-Ward method but, as mentioned, it may be more close to the ‘true’ value of *D* index. Indeed, in Fig. 6c, the clustering result from *K*-Ward method, a large cluster on the middle-bottom area and on the right side along the river has a very low dispersion degree (high continuity degree). In the same area in Fig. 6d, the result from SpaRef, boundaries of this cluster with other clusters are curtailed and hence the dispersion degree of this cluster is higher. In Fig. 6a and b, the clustering results from ISODATA and *K*-means, respectively, the study area is dispersed and shared with other clusters. The dispersion degrees of these clustering results are thus very high. Overall, SpaRef does very well on all criteria simultaneously.

## 6. Conclusion

The paper presents a new clustering algorithm, SpaRef, for hyperspectral images. The proposed clustering method, using spatial information, has the advantages to be stable, and leads to clusters with a high degree of compactness and continuity. Moreover, SpaRef can work with a large dataset, by applying an agglomerative merging process on a moderate number of highly homogenous classes, instead of on a very high number of points. Potential shortcomings of the agglomerative hierarchical clustering are corrected by introducing a refinement process to points in the spatial domain. SpaRef method has given good

results on Klompenwaard CASI image where it has been compared with *K*-means, ISODATA and Ward’s method. It would be a straight-forward step to successfully apply the algorithm to multispectral chemical images.

The noise, mixed pixel and very small objects are not taken into account by SpaRef. Future work on categorisation of very small classes is necessary in order to cluster a complete image.

## Acknowledgements

We thank Gertjan Geerling for sharing the data and stimulating discussions.

## References

- [1] A.K. Jain, M.N. Murty, P.J. Flynn, *ACM Comput. Surv.* 31 (3) (1999) 264–323.
- [2] T. Duda, M. Canty, *Int. J. Remote Sens.* 23 (11) (2002) 2193–2212.
- [3] G.H. Ball, D.J. Hall, *ISODATA—A Novel Method of Data Analysis and Pattern Classification*, Springfield, Stanford, 1965.
- [4] M. Halkidi, Y. Batistakis, M. Vazirgiannis, *SIGMOD Rec.* 31 (2) (2002) 40–45.
- [5] R.G. Brereton, *Multivariate Pattern Recognition in Chemometrics, Illustrated by Case Studies*, Elsevier, Amsterdam, 1992.
- [6] A. El-Hamdouchi, P. Willett, *Comp. J.* 32 (1989) 3.
- [7] M. Amadasun, R.A. King, *Pattern Recognit.* 21 (3) (1988) 261–268.
- [8] M.R. Anderberg, *Cluster Analysis for Applications*, Academic Press, New York, 1973.
- [9] P.H.A. Sneath, R.R. Sokal, *Numerical Taxonomy*, Freeman, London, UK, 1973.
- [10] B. King, *J. Am. Stat. Assoc.* 69 (1967) 86–101.
- [11] J.H.J. Ward, *J. Am. Stat. Assoc.* 58 (1963) 236–244.
- [12] J.C. Duun, *J. Cybern.* 4 (1974) 95–104.
- [13] D.L. Davies, D.W. Bouldin, *IEEE Trans. Pattern Anal. Mach. Int.* 1 (2) (1979) 224–227.
- [14] R. Kothari, D. Pitts, *Pattern Recognit. Lett.* 20 (1999) 405–416.
- [15] J.C. Bezdek, N.R. Pal, *IEEE Trans. Syst. Man. Cyber. Part B* 28 (3) (1998) 301–315.
- [16] J.A. Richards, X. Jia, *Remote Sensing Digital Image Analysis*, Springer, Berlin, 1999.
- [17] <http://www.ece.purdue.edu/~biehl/MultiSpec/Index.html>.
- [18] <http://www.erdas.com>.
- [19] C. Goutte, P. Toft, E. Rostrup, F. Nielsen, L. Hansen, *NeuroImage* 9 (1999) 298–310.