Including Spatial Information in Clustering of Multi-Channel Images

een wetenschappelijke proeve op het gebied van de
Natuurwetenschappen, Wiskunde en Informatica

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de Rector Magnificus prof. dr. C.W.P.M. Blom,
volgens besluit van het College van Decanen
in het openbaar te verdedigen op maandag 21 november 2005
des namiddags om 3.30 uur precies

door
Thanh Ngoc Tran
Geboren op 25 juli 1973
Te Hanoi, Vietnam

Promotor:	Prof. dr. Lutgarde M.C. Buydens
Copromotor:	Dr. Ron Wehrens
Manuscriptcommissie:	Prof. dr. Piet van Espen University of Antwerpen, Belgium
	Prof. dr. Freek van der Meer International Institute for Geo- Information Science and Earth Observation (ITC)
	Dr. Dirk H. Hoekman Wageningen University

Print Print Partners Ipskamp
ISBN 90-9019787-7
Cover photo: C-band Polarimetric SAR Image of Flevoland in the Netherlands
(Source Wageningen University)

CONTENTS

MOTIVATION, OBJECTIVE AND OVERVIEW OF THE THESIS

This thesis is the result of work in clustering of multi-variate/-spectral images at the Department of Analytical Chemistry, Institute of Molecules and Materials (IMM), Radboud University Nijmegen, The Netherlands. In this chapter, the motivation and objectives of this thesis are presented. Finally, an overview of the content of the thesis will be given.

1.1 Motivation

Nowadays, high resolution images are often measured in many current imaging systems. Clustering has become an important tool for revealing underlying structure in images for various applications. For example, remote sensing images have made it possible to map remote areas and to update existing information efficiently and cheaply at both global and regional scales. Advances in spatial resolution allow us to work on even very small scales. In applications such as daily monitoring of agricultural objects by creating agricultural block-maps or maps for urban and wet-, flood-land areas, most of the interpretation is still made by human experts on aerial photos (Rydberg, 2001). This is an expensive procedure, and in many cases impossible for the huge numbers of images that have been collected over several years over large study areas. Hence, an automated classification method would reduce costs significantly, and makes many previously impractical applications feasible.

Supervised classification is preferable when training samples are available. However, collecting training samples again consumes very much time and effort. Sometimes, it is even impossible because of the size or accessibility of the research area. Clustering (i.e. unsupervised classification), on the other hand, works without the need of prior knowledge in the form of training samples. Human experts are still useful to verify clustering results and to make a decision to select a specific clustering method that is the most appropriate for the dataset at hand. This can normally be done using a smaller sample dataset and it can be extended to the larger set or to another dataset of the same type.

Not only in remote sensing applications, but also in many other fields, clustering techniques play an important role. Clustering of Magnetic Resonance Images (MRI) and X-ray images has been applied for quality inspection of food, vegetables and postharvest products (Abbott, 1999, Hall et al., 1998, and Noordam, 2005), in which without a priori information, clustering is used to detect small defects or abnormalities in the inspection object image. In medical applications, with the recent development of Magnetic Resonance Spectroscopic Imaging (MRS), clustering of the combination of MRI and MRS data brings more reliable and non-invasive brain tumor diagnosis (Simonetti, 2004).

1.1.1 Clustering

Clustering normally works with no prior knowledge about the classes that are present. There are many ways to define clustering:

> *"clustering in which each of member of clusters is **in some way similar** and different from the members of other clusters." (Kaufman, 1990)*

"clustering is used to classify objects, characterized by the values of a set of variables, into groups." (Vandeginste et. al. 1998).

*"clustering is to help to understand **relationships** of objects by similarity" (Tran, 2004)*

A fundamental issue in clustering is in the definition of the "similarity" of objects to form a "natural" ("homogeneous") group. Due to the adaptivity of the "similarity" concept, it is too much to expect a single method to be optimal for all cases; for example, in remote sensing, land cover types within the urban environment have a very complex nature and diverse composition. Hence, "homogeneity" is also diverse. Moreover, clusters can have different shapes, sizes, populations, or distributions. A huge number of clustering methods therefore has been developed during the last decades and it is necessary to know in which cases, which clustering method can do best (Chapter 2).

Partitional clustering methods, such as K-means, fuzzy C-means (Bezdek, 1981), ISODATA (Ball and Hall, 1965) and mixture modelling (McLachlan and Peel, 2000) by Expectation Maximization (Dempster et al. 1977) are the most often-used methods for moderate and large datasets, due to their time-efficient computation. Especially mixture model clustering, modeling a statistical distribution by a finite mixture distribution of other distributions, becomes more and more popular nowadays in remote sensing applications (Ichoku and Karnieli, 1996, Brown et al., 2000) and many other applications; see, e.g, (Yeung et al., 2001, Alexandridis et al., 2004) for clustering gene expression in genomics. However the initialization is critical on determining the right input parameters (Fraley and Raftery, 2002).

Some other methods based on a hierarchical clustering scheme and mixture modelling, e.g. model-based clustering (Fraley and Raftery, 2002), find a better way to identify the number of clusters and the corresponding input parameters. However, applying them to large datasets is difficult. It often happens that clusters are overlapping; information of objects in the overlapping area may be very similar and it is hard or even impossible to separate these objects.

1.1.2 Clustering multi-spectral images

Image data is different from normal spectrum-only data because of the availability of the spatial information, the spatial relations between pixels in the image. This is important information which can improve the performance of clustering methods. However, most image clustering methods are pixel-based approaches, taking pixel by pixel without paying attention to the spatial information. In this thesis specific research questions were addressed:

Q1. How can we use spatial information to derive better clustering algorithms?

Q2. Can we make the algorithms efficient for very large multi-spectral images?

Q3. Can we apply the algorithms for images of a very high spectral dimension?

Q4. Can we automatically identify the number of clusters in the image?

Detailed discussion of the problems and guidelines to clustering on multivariate images are given in Chapter 2.

1.2 Objective

The objective of this thesis is to study a possibility of extension of clustering techniques, especially the mixture modelling, to moderate and large multivariate/multi-spectral images

taking advantage of spatial information. The main interest is to improve the robustness of clustering methods (input parameters and the number of classes), and the total accuracy by reducing the influence of the problems of overlapping clusters and noise on (but not limited to) remotely sensed images.

1.3 Overview of the thesis

This thesis comprises research papers that were written during participation in the doctoral program at the Department of Analytical Chemistry, Radboud University Nijmegen.

Chapter 2 presents a detailed introduction to the major types of clustering techniques and their problems. Particular attention will be devoted to the extension to take into account both spectral and spatial information of the image data. General guidelines for the optimal use of these algorithms are given.

Chapter 3 focuses on the automatic determination of the number of clusters (*Question four*) in a high dimensional data set (*Question three*). Especially, the proposed method, KNNCLUST, is based on nonparametric density-based clustering methods. This method has major advantages over all traditional density-based methods to deal with clusters of widely different densities. Spatial information is not used by this method: this will be studied intensively starting from the next chapter. Due to a reasonably high computational complexity, KNNCLUST is useful for small datasets with the problem of different cluster densities.

The thesis pays particular attention to solutions to *Question one*. The spatial information can be used at different places in the clustering process; either at the beginning of the clustering process to identify good initial parameters, or during clustering by introducing a weight function to the ordinary similarity function, or at the final stage to filter the clustering result to improve performance of clustering algorithms. Especially, the influence of overlapping clusters, noise/artefacts, and mixed pixels are reduced by the modification of the similarity function by a weight function, so that pixels in the overlapping area are closer if they form a spatial region, otherwise they are far apart. It will be illustrated in Chapter 2, 5, and 6. Noise can be treated in the same way.

In **Chapter 4**, the simple combination of the often-used k-means clustering and Ward's hierarchical clustering is presented. The refinement step, introduced at the end of the clustering algorithm, uses spatial information. It leads to an improvement of the performance of clustering on a remote sensing Compact Airborne Spectrographic Imager (CASI) image from an area in the Klompenwaard, the Netherlands.

Mixture model clustering becomes more and more popular and a central issue is determining the number of components (clusters) and their initial parameters. For a large and complex image, it is often very hard to apply the mixture model clustering to the entire image (Fraley and Raftery, 2002, Murat Dundar and Landgrebe, 2002). These situations will be investigated in **Chapters 5** and **6**, where the spatial information is used to deal with these problems (*Question one*).

Briefly, **Chapter 5** uses the combination of statistical testing and hierarchical clustering to produce the initial parameters for clusters. In **Chapter 6**, two novel strategies to mixture model clustering on multivariate image are proposed. One of the strategies is intended for the normal situation of mixture modelling, where the density of a cluster is modelled by a single normal distribution, the second is designed for a more complex situation, where the density of individual clusters is a mixture of several normal sub-clusters.

The main part of both strategies is the estimation of the initial parameters based on the combination of the simple region growing segmentation and model-based hierarchical clustering (Fraley, 1998). Since the number of regions is much smaller than the number of pixels, the algorithm can work very fast (*Questions one and two*). In the case where one cluster is modelled by several Gaussians, an additional merge is performed to join clusters that are overlapping; these can be regarded as sub-clusters. The final classification step extends the classification to the entire image. Again, spatial information can optionally be used to improve clustering by using Markov Random Field (*Question one*). The clustering procedure is fast enough to be used for moderate-size and large multivariate images (*Question two*).

In both **Chapter 5,** and **6**, the best model is identified by the Pseudolikelihood Information Criterion (PLIC) (Stanford and Raftery, 2002) and Bayesian Information Criterion (BIC) (Schwarz, 1978), respectively (*Question four*). In **chapter 7**, we summarize our conclusions from preceding chapters and discuss directions for future research.

References

Abbott, J.A., (1999). Quality measurement of fruits and vegetables, Postharvest Biology and Technology. 15(3) 207-225.

Alexandridis, R. Lin, S. and Irwin, M (2004), Class discovery and classification of tumor samples using mixture modeling of gene expression data-a unified approach, Bioinformatics, 20(16) 2545-2552

Ball, G.H., and Hall, D.J. (1965) ISODATA, A novel method of data analysis and pattern classification, Techn. Rep., Stanford Research Institute, Menlo Park, CA.

Bezdek, J.C. (1981) Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York.

Brown, M. Lewis, H.G. Gunn, S.R. (2000).Linear spectral mixture models and support vector machines for remote sensing. IEEE Trans. on Geoscience and Remote Sensing, 38(5) 2346 – 2360.

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977) Maximum likelihood from incomplete data via the EM algorithm , J. R. Statist. Soc. B, (39) 1-38.

Fraley, C. (1998) Algorithms for Model-Based Gaussian Hierarchical Clustering, SIAM J. Sci. Comput., (20) 270-281

Fraley, C. and Raftery, A. E. (2002) Model-based clustering, discriminant analysis, and density estimation, J. the Amer. Statist. Asso., (97) 611-631.

Hall, L., Evans, S. and Nott, K. (1998) Measurement of textural changes of food by MRI relaxometry. Magnetric Resonance Imaging, 14 (5/6) 485-492

Ichoku, C., and Karnieli, A., (1996), A review of mixture modeling techniques for sub-pixel land cover estimation, Remote Sensing Reviews, 13.

Kaufman, L., and Rousseeuw, P. J. (1990). Finding groups in data: an introduction to cluster analysis. New York: Wiley.

McLachlan, G. and Peel, D. (2000). Finite Mixture Models, Willey series in probability and statistic, Canada.

Murat Dundar, M. and Landgrebe, D. (2002). A model-based mixture-supervised classification approach in hyperspectral data analysis. IEEE Trans. Geosci. Remote Sensing . 40(12) 2692-2699.

Noordam, J.C. (2005) Chemometrics in multispectral imaging for quality inspection of postharvest products, Phd. thesis, Radboud University of Nijmegen.

Tran, T.N., Wehrens, R., and Buydens, L.M.C. (2005). Clustering multi-spectral images: a tutorial, to appear in Chemom. Intell. Lab. Syst.

Rydberg, A. (2001). Multispectral image analysis for extraction of remotely sensed features in agricultural fields. Phd. thesis. Swedish University of Agricultural Sciences.

Schwarz, G. (1978). Estimating the dimension of a model. Annuals of Statistics (6) 461-464.

Simonetti, A. (2004) Investigation of brain tumor classification and its reliability using Chemometrics on MR spectroscopy and MR imaging data. Phd. thesis, Radboud University of Nijmegen.

Stanford, D.C., and Raftery, A.E. (2002) Approximate Bayes Factors for Image Segmentation: The Pseudolikelihood Information Criterion (PLIC). IEEE Trans. on Pattern Anal. Mach. Intell. (24) 1517-1520.

Vandeginste, B.G. M., Massart, D.L., Buydens, L.M.C., Jong, S.De., Lewi, P.J., and Smeyers-verbeke, J. (1998). Handbook of Chemometrics and Qualimetrics, Part B. Elsevier, 57

CLUSTERING MULTI-SPECTRAL IMAGES: A TUTORIAL

Abstract

A huge number of clustering methods have been applied to many different kinds of data set including multivariate images, such as magnetic resonance images and remote sensing images. However, not many methods include spatial information of the image data. In this tutorial, the major types of clustering techniques are summarized. Particular attention will be devoted to the extension of clustering techniques to take into account both spectral and spatial information of the multivariate image data. General guidelines for the optimal use of these algorithms are given. The application of pre- and post-processing methods is also discussed.

Keywords: Pattern recognition; Unsupervised classification.

T.N. Tran, R. Wehrens and L.M.C. Buydens, *Chemometrics and Intelligent Laboratory Systems* ,vol. 77/1-2, pp. 3-17, 2005.

1. Introduction

Automatic grouping of pixels having a "similar characteristic" in a multivariate image is an important problem in a variety of research areas such as biology, chemistry, medicine, and computer vision. In spite of several decades of research, the task is still challenging due to the dramatic improvement of imaging technology in recent years. Examples are magnetic resonance images (MRI), which has become a standard tool in medicine, and remote sensing of the earth surface from satellite or airborne scanners. In both examples, a huge number of multivariate images, often with a very high spectral and spatial resolution, are generated routinely. If there is no priori information about the classes, the grouping of pixels has to be done in an unsupervised way. This is called clustering [1][2][3]. In general, clustering groups objects characterized by the values of a set of variables into separate groups (clusters), based on their "similarities". This may help to understand relationships that may exist among them. Examples of the application of clustering techniques on non-image data type in chemometrics are exploring of environmental data structure representing physical and chemical parameters [4], computational analysis of microarray gene expression profiles [5], or electron probe X-ray microanalysis in [6]. In these cases, the clustering method is integrated with visual display allowing direct interpretation of internal structure of the data. Another application is identifying chemical compounds for combinatorial chemistry [7], where clustering was studied on a data set of alcohols and the interpretation of the results was consistent with chemistry. Clustering can also be combined with other methods such as genetic algorithms for molecular descriptor selection in [8]. And last but not least, clustering can also be applied for process monitoring [9][10][11]. In this case, cluster centers are updated automatically by the method according changes due to, e.g., process drifts by seasonal fluctuations [9]. Clustering helps to interpret the model and study short-term changes and long-term changes due to drifting [10].

Clustering techniques can also be applied to multivariate images. In general, a multivariate image is defined as a stack of images, where each image represents a different variable. Many physical characteristics can be used in multivariate images such as temperature, mass, wavelength, polarization etc. As an example, MRI T1 and T2 weight images, corresponding to different relaxation times, are often use in clinical decision making. More general, a variable can be also a latent variable, e.g. principal components (PCs). These (latent) variables form the so-called feature information of pixels in the multivariate image. A major difference with non-image data is that spatial information, in the form of X and Y coordinates, is available besides pixel information on the feature space. In general, we expect that classes form spatially continuous regions. This is sometimes called a "spatial relation" of neighbor pixels, "local characteristics", or "local dependency" [12],[13]. Spatial information is usually ignored. In most cases, taking it into account will improve the clustering result significantly. Examples of the application of clustering of multivariate images in chemometrics are the localization of clusters of brain tumours in MRI images [14][15], or identifying clusters of pixels having similar ground cover types in remote sensing images [16].

In this tutorial, the main types of problems for clustering of multivariate images are discussed in detail in section 2. In the following sections, the major types of clustering techniques are overviewed and possible extensions taking into account spatial information [14][15][16] will be evaluated. Preprocessing of multivariate images and post-processing of clustering results will be treated in the last section.

2. Problems for clustering multivariate images

We consider a multivariate image containing N pixels (objects) in d-dimensional multivariate space (also called a *feature* space). In other words, a pixel (an object) is described by d variables corresponding to the d-dimensional feature space. The main problems encountered when clustering multivariate images are listed below:

Image size: The improvement in image sensor sensibilities has increased the resolution in the spatial domain of multivariate images drastically. As a result, the size of images has increased too. The size of typical data set can easily get up to millions of pixels. For many clustering algorithms, especially the ones that use a distance matrix such as hierarchical methods, this is prohibitive in terms of memory and processing time.

Feature dimension: The improvement in image sensor sensibilities gains not only a large number of pixels but also a large number of variables. In many cases, the inverse of the covariance of a cluster has to be computed during clustering. This is very expensive. For very small clusters, it may not be possible to calculate, because of singularity.

Noise: Many image scanners produce noise/outliers in images due to limited sensor sensitivity, statistical variation, or signal interference (c.f. the speckle in the SAR Flevoland image data in Figure. 3a.). Not only can noise make the result very difficult to interpret, but also it can lead to a completely wrong solution.

Mixed pixels: Despite the increase in resolution of scanners, pixels often contain the spectral response of several components. These pixels are not easily classified in one cluster. Some clustering methods, such as fuzzy C-means, allow a mixed pixel to be classified to more than one cluster. Another approach is to use spatial information.

In addition, several general problems are also relevant to clustering images.

Overlapping clusters: More often than not, clusters are overlapping in the feature domain; even though two objects may belong to different clusters they may have features that are very similar. Then, if a clustering algorithm uses only feature information, it will not lead to a good result.

Number of clusters: In many cases there is no clear priori reason to favor a particular number of clusters. The clustering method then has to find the best number of clusters from the data. This is often very difficult to find.

Unequal cluster density: The density of a cluster at a particular point in the feature space is the number of pixels contained in a unit of the data space. Clustering methods based on density often have problems with clusters of very different densities, e.g. river and lake clusters in [17].

Unequal cluster size: if cluster populations are very different then it could influence clustering results. Sometimes, a small cluster can be very important but it is often not found because the larger clusters determine the clustering result. For example in an image of a St. Paulia flower, it is difficult to recognize a pistil on the image [18]. This problem can be different from the unequal density problem when the densities remain the same and the feature space is very different.

In summary, the image size and the feature dimension problems often make a method unsuitable due to computation time and computer memory. On the other hand, other problems affect the accuracy of clustering method rather than its feasibility. They will be discussed in more detail in section 5 and 6. In many cases, clustering taking into account spatial information can reduce the influence of these problems to clustering accuracy [12][13][19].

3. Example images

In this tutorial, three experimental setups are used for demonstration purposes.

Experiment 1 (SYN)*:* A synthetic image of size 40 x 40 consisting of two overlapping Gaussian clusters in one dimension is generated. Pixels of two Gaussian clusters are distributed in the image as shown in Figure 1a. One is in the centre of the image and the other is around it. The density functions of the two distributions are plotted in Figure 1b. It illustrates the overlap of two clusters in the feature domain.

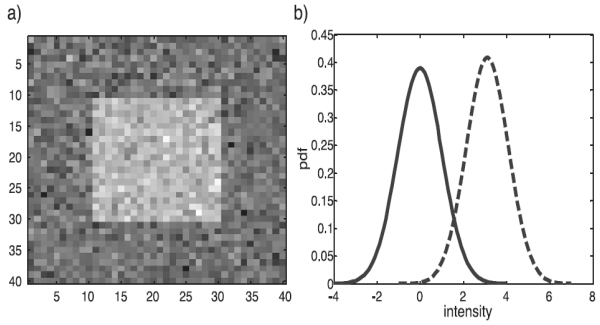


Figure 1. SYN image. (a) Gray image size 40 x 40. (b) Gaussian distribution functions of the two clusters.

Experiment 2 (MEAT): A multivariate image of minced meat was recorded with the ImSpector V7 imaging spectrograph (Spectral Imaging Oulu, Findland) as described in [20]. The image size is 318x318 with 257 variables (bands) from 396 nm to 736 nm (1.3 nm for each band). The incoming light is split and captured by CCD Sony camera to obtain a color image, which will be used as the reference image for clustering result. The CCD color image and the plot of representative spectral of clusters are shown in Figure 2a and b, respectively. The full spectral image of large number of variables is pre-processed by averaging technique to 11 planes (bands) image in order to reduce computation time.

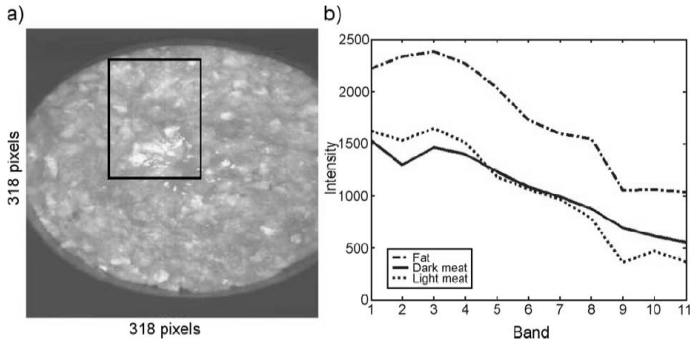


Figure 2. (a) Meat CCD color image of size 318x318, (b) representative spectra for clusters: a fat meat spectrum located at (119, 134), a dark meat spectrum at (32, 119) and a light meat spectrum at (78, 94).

The CCD image shows a petri disk filled with a piece of minced meat. It contains 4 classes: the petri disk, dark meat, light meat and fat. The difference between dark meat and light meat is caused by the amount of blood in the meat. The dark pixels represent the dark meat class and the white spots represent the fat class. The fat class is quite separated from other classes. The light meat class surrounds the fat class and gradually turns into the dark meat class. This causes the overlap problem between the dark meat and light meat classes [20]. The large number of variables and the overlap of clusters are problems for clustering this image.

Experiment 3 (SAR)**:** An area of 400 x 400 pixels of a remote sensing SAR image was taken over Flevoland, an agricultural area in The Netherlands, by the NASA/Jet Propulsion Laboratory (JPL) AirSAR on 3 July 1991. The image used here is in C- and L-band full polarimetry and contains 18 intensities. Figure 3a shows a false-color image of first three intensities of the image data. Ideally, one would like to obtain a clustering that corresponds to the seven expected crop types [21], as shown in Figure 3b.

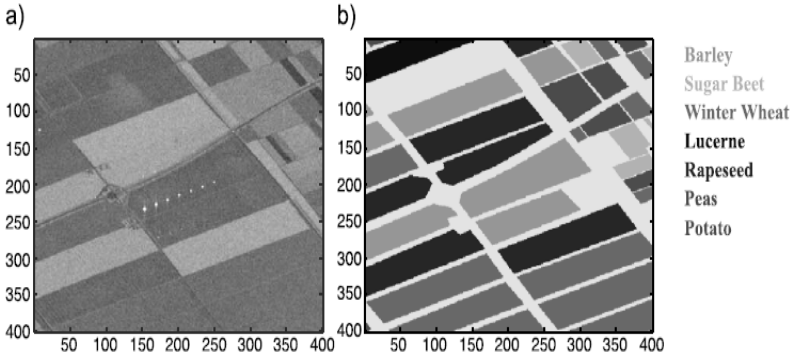


Figure 3. (a) False-color image of the first three intensities on C-band of 400 x 400 pixels. (b) Map of seven crop types (ground-truth). The Yellow color is a mask where the ground truth is uncertain: these pixels are predicted but does not take into account when calculating prediction accuracy of clustering result.

Heavily overlapping clusters are shown in Figure 4 between Barley (Green) and Winter Wheat (Magenta) clusters. Noise is also present in the data set due to statistical variation of the signal (speckle). Noise and cluster overlap are the two main problems for this image.

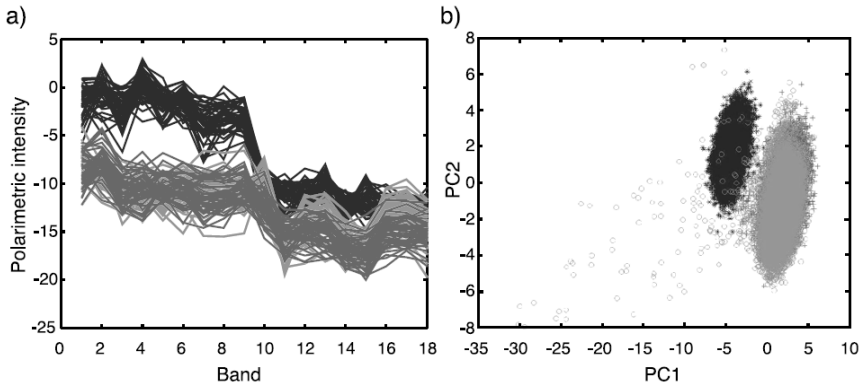


Figure 4. (a) Spectra of 50 objects for each of the three classes, Barley (Green), Winter Wheat (Magenta), and Rapeseed (Brown), (b) Score plot of the two first PCs of all pixels in the three classes.

4. Similarity Measures

A measure of similarity is essential to clustering. It can be a distance in deterministic clustering or a likelihood in probabilistic clustering. Both are called similarity function in this tutorial and will be indicated by $\wp$.

4.1. Similarity measures with no spatial information

The similarity function with no spatial information uses only information in the feature space. It can be calculated between two pixels, two clusters, or between a pixel and a cluster. In the deterministic case, the most popular measure of dissimilarity between pixels x_i and x_j is the Euclidean distance, $\wp_{eucl}\left(x_i, x_j\right)$, which is a special case of the Minkowski distance with p = 2. This is given by:

$$\wp_{\text{minkovski}}(x_i, x_j) = \left(\sum_{l=1}^{d} \left| x_{il} - x_{jl} \right|^p \right)^{\frac{1}{p}} \quad (1)$$

where $x_i = \{x_{i1}, .., x_{id}\}$. The Minkowski distance with p = 1 is called the Manhattan distance. These distances can be also applied for measuring dissimilarity between pixel x_i and cluster ω_j where the mean of cluster ω_j, μ_{ω_j}, is used instead of the pixel x_j. However, the covariance C_{ω_j} of the cluster is then not taken into account. The Mahalanobis distance, on the other hand, does use the covariance:

$$\wp_{\text{Mahalanobis}}(x_i, \omega_j) = (x_i - \mu_{\omega_j})^T C_{\omega_j}^{-1} (x_i - \mu_{\omega_j}) \quad (2)$$

The Bhattacharyya distance is a distance between two clusters, ω_1 and ω_2, both having a normal distribution:

$$\wp_{\text{Bhattacharyya}}(\omega_1, \omega_2) = \frac{1}{8} (\mu_{\omega_1} - \mu_{\omega_2})^T \left(\frac{C_{\omega_1} + C_{\omega_2}}{2} \right)^{-1} (\mu_{\omega_1} - \mu_{\omega_2}) + \frac{1}{2} \ln \left(\frac{\left| C_{\omega_1} + C_{\omega_2} \right|}{2\sqrt{\left| C_{\omega_1} \right| \left| C_{\omega_2} \right|}} \right) \quad (3)$$

where μ and C again indicate means and covariances, respectively. The first part of the Bhattacharyya distance is dominated by the difference in means, and the second part by the difference in covariance.

In the probabilistic case, where clusters are explicitly modeled as a distribution, such as a t-distribution or a normal distribution, the likelihood is used as similarity function [22], [23]. More details are discussed in section 5.

4.2. Including spatial information in the similarity measure

In (multivariate) images, the spatial information of a pixel x_i consists of cluster information of the neighbor pixels. Many neighbor-schemes, ∂_i, can be used. An often-used one is a square window centered at the pixel x_i. In principle, it is possible to define a similarity function that not only takes into account information in the feature domain, but also clustering information of neighboring pixels. This can be done by a weight function $w(x_i, \partial_i, \omega_j)$ to the cluster ω_j. Such a similarity function for comparing a pixel and a cluster is generally expressed by two general forms as follows:

Addition form: $\tilde{\wp}(x_i, \omega_j) = \wp(x_i, \omega_j) + w(x_i, \partial_i, \omega_j)$ (4)

Multiplication form: $\tilde{\wp}(x_i, \omega_j) = w(x_i, \partial_i, \omega_j) . \wp(x_i, \omega_j)$ (5)

The spatial weight function $w(x_i, \partial_i, \omega_j)$ is defined differently depending on the particular clustering method, which will be discussed in more detail in section 5. Similar expressions could be set up to compare two pixels or two clusters, but this has not appeared in the literature.

5. Clustering techniques

5.1 General ideas

One can often see clustering in a "hard" form, which assigns each pixel x_i to one and only one cluster. A "soft" or a "fuzzy" technique, on the other hand, assigns to each pixel x_i a fractional degree of membership $u_{ij} \in [0,1]$ for all clusters. The higher the degree of the membership u_{ij}, the more probable it is that the pixel x_i belongs to cluster j. A deterministic similarity is often used in hard clustering and a probabilistic distance (or a fuzzy variant) is used in soft clustering. A soft clustering contains more information than a hard clustering and it can be converted to a hard clustering.

Clustering techniques in general can be categorized into three main types: partitional clustering, hierarchical clustering, and density-based clustering, as illustrated in Figure 5. Each can be further subdivided, the best-known clustering algorithms in chemometrics are K-means, Fuzzy C-means [9][19], hierarchical agglomerative [4][5], model-based [14] (or mixture modeling), and density-based [24] clustering methods.

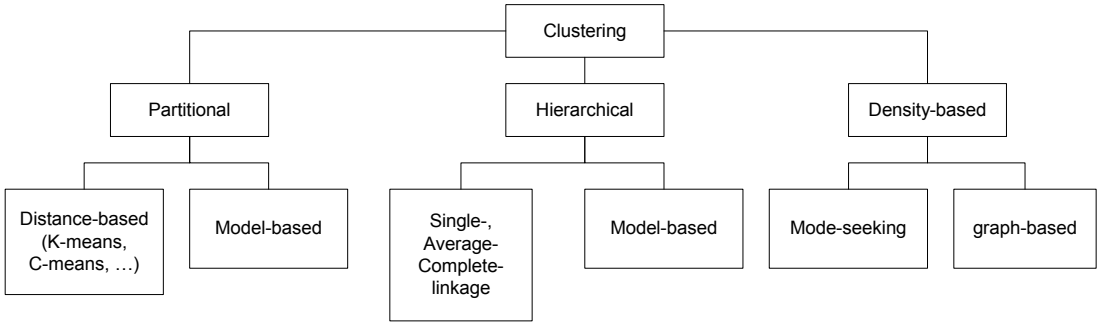


Figure 5. A taxonomy of clustering methods.

5.2. Partitional clustering

5.2.1. Ordinary partitional clustering without spatial information

Given a number of clusters, g, a partitional clustering technique seeks an organization of pixels which optimizes a target function E. This can be a minimum or maximum, depending on the clustering method. E.g, in Kmeans a compactness function is minimized and in model-based clustering, the log-likelihood is maximized. E can be written as:

$$E = \sum_{j=1}^{g} \sum_{i \in C_i} u_{ij} \wp\left(x_i, \omega_j\right) \tag{6}$$

where u_{ij} is a degree of membership of the pixel x_i. Again u_{ij} is either 0 or 1 in "hard" partitional clustering methods. In fuzzy clustering, u_{ij} is replaced with u^q_{ij} with $q > 1$. Often, $q = 2$ is used.

An optimal solution for this clustering problem requires an exhaustive combinatorial search, but it is not possible to perform in practice. It is often estimated by an iterative process:

Algorithm:

(1) Start: The algorithm starts with an initial guess of the set $u_{ij} \in [0,1]$, often random.

(2) Iteration: A number of iterations are performed to improve the compactness function (Eq. 6) by updating the degree of membership u_{ij} according to new centroids of the clusters. In "hard" partitional clustering, it is interpreted as assigning each pixel to the cluster with the smallest $\wp(x_i,\omega_j)$. The degree of membership is updated as:

u_{ij} = 1iff $\wp(x_i,\omega_j)$ = minimum of $\wp(x_i,\omega_j) \mid \forall k$)

u_{ij} = 0 otherwise.

(3) End: The algorithm ends if a stop-criterion holds, otherwise the algorithm is repeated at step 2. The stop-criterion could be a number of iterations, a threshold of the compactness function or convergence of the solution. The algorithm basically provides a better solution with more iterations and more processing time.

A big advantage of partional clustering is the computation time. The complexity is only Nlog(N), where N is the number of pixels. This makes it possible to apply the algorithm to even very large data sets. However, partional clustering has several drawbacks:

- The number of clusters needs to be defined before hand, the "number of cluster problem".

- Most partional clustering methods are heavily dependent on the initial guess. It may lead to very different results upon repeated application. A locally optimal solution is often obtained instead of the global optimum of the compactness function

- The "unequal cluster size problem" might influence the clustering result, because the centre of a smaller cluster often tends to drift to an adjacent larger cluster.

- 'Noise', present in a data, also interferes with the result of the partitional clustering by influencing the calculation of new cluster centers. It is less influential in a soft/fuzzy clustering because pixels far from the center of a cluster, such as noise/outliers, are assigned a lower degree of membership.

Partitional clustering methods can be divided into deterministic and model-based approaches.

5.2.1.1 Deterministic Partitional Clustering

A deterministic partitional clustering is a partitional clustering where the similarity function, $\wp(x_i,\omega_j)$, is a distance. Different deterministic partitional clustering algorithms have different definitions of the distance $\wp(x_i,\omega_j)$ and ways of updating of the membership degree u_{ij}. The most popular hard deterministic partitional clustering is K-means, where the distance $\wp(x_i,\omega_j)$ is the Euclidean distance $\wp_{eucl}(x_i,\omega_j)$.

Some variation of the K-means algorithm involves selecting a different distance function $\wp(x_i,\omega_j)$, for instance the Mahalanobis distance [25], but the algorithm then tends to produce unusually large or unusually small clusters.

Another variation of K-means is ISODATA clustering [26]. It is designed to solve the "number of clusters" problem. ISODATA starts with a high number of clusters. The method is different from the ordinary partitional method, which permits splitting a '*big*' cluster, merging two close clusters, and deleting a very small cluster. This way, the number of clusters is identified by the method. However, thresholds for cluster variance and cluster size need to be defined, which are difficult to control in practice.

Nowadays, much attention has been paid to soft or fuzzy deterministic partitional clustering. Fuzzy C-means (FCM) or fuzzy K-means (FKM) is a famous example of this type [9][15][27][28][29]. During the iterations, the fuzzy membership u_{ij} is updated as a function of distance to clusters:

$$u_{ij} = \frac{1}{\sum_{c=1}^{g} \left(\frac{\wp_{c-means}(x_i, \omega_j)}{\wp_{c-means}(x_i, \omega_c)} \right)^{\frac{1}{q-1}}} \tag{7}$$

where $q>1$ is the fuzziness index. Normally, q is 2. The similarity function is given by:

$$\wp_{c-means}(x_i, \omega_j) = (x_i - \mu_{\omega_j})^T A (x_i - \mu_{\omega_j}) \tag{8}$$

where A is a d x d symmetric, positive definite matrix, and d is the feature dimension of the data set. The $\wp_{c\text{-}means}(x_i,\omega_j)$ distance is $\wp_{mahalanobis}(x_i,\omega_j)$ when A is the inverse of covariance matrix, or it can be $\wp_{eucl}(x_i,\omega_j)$ when A is the identity matrix. The later case is the usual form of $\wp_{c\text{-}means}(x_i,\omega_j)$.

The similarity function is defined differently in the FMLE (Fuzzy Modification of the Maximum Likelihood Estimation) algorithm [28] and p-norm FCM [29], where an exponential distance and the Minkowski distance are employed, respectively.

A nice feature of fuzzy deterministic partitional clustering is that a pixel in an area of overlapping clusters is not assigned with a very high membership. This way, it does not influence the cluster parameters very much. In other words, such a pixel always has a larger uncertainty. This also holds for outliers/noise.

As the example, clustering results of SYN image by K-means and FCM algorithms to two clusters, corresponding to White and Black area, are plotted in Figure 6. Many pixels in the overlapped area are misclassified.

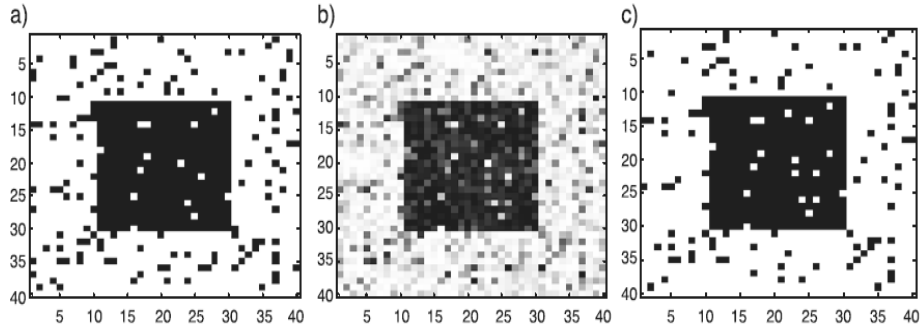


Figure 6. Clustering result of SYN image to two clusters (White and Black); a) by K-means, b) by fuzzy C-means, c) hard clustering result based on fuzzy C-means. The gray pixels indicate fuzzy memberships in the fuzzy C-means clustering result.

The results of clustering MEAT by K-means and the hard result based on FCM are given in the Figure 7. Due to the overlapping clusters, both have quite similar problems. Many dark meat areas are replaced by light meat and the fat spots are extending over the light meat regions. However, the problem is smaller with FCM.

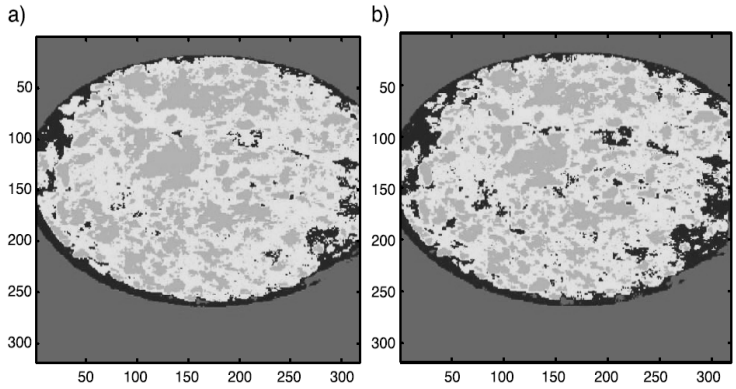


Figure 7. Clustering of MEAT image by (a) K-means, (b) Fuzzy C_means.

5.2.1.2 Model-Based Clustering (MBC)

Model-based clustering, sometimes also called mixture modeling, is a "soft" partitional clustering based on a statistical approach [23][30]. Every cluster c is described by a multivariate distribution f with parameters θ_c. For example, for the Gaussian distribution, the most often used, θ_c contains mean μ_c and covariance C_c. The total data set is described by a linear combination of individual cluster and the coefficients correspond to mixture proportions π_c.

The probability density function of the pixel x_i under a g-component (cluster) mixture is given by:

$$f(x_i;\Psi)=\sum_{c=1}^{g}\pi_c f(x_i;\theta_c) \tag{9}$$

Now, the probabilistic likelihood function is given by the following expression:

$$L(\Psi)=\prod_{i=1}^{n} f(x_i;\Psi) \tag{10}$$

where ψ contains all cluster parameters and mixture proportions. The aim of the model-based clustering is to obtain a configuration ψ in which it maximizes the log-likelihood $\log L(\psi)$. This is equivalent to the "optimizing the log-likelihood":

$$\log L(\Psi)=\sum_{c=1}^{g}\sum_{i}^{n} u_{ic}\log(\pi_c f(x_i;\theta_c)) \tag{11}$$

where u_{ic} corresponds to the conditional probability of object x_i belonging to cluster c. The maximization of the log-likelihood probability function is analogous to the optimization of the compactness function (eq. 6). This is usually performed by the EM (Expectation Maximum) algorithm [31]. According to the general procedure for partitional clustering, the step 2 is split into two sub-steps in EM algorithm, called the M-step (Maximization step) maximizing π_c and θ_c, and the E-step (conditional Expectation step), estimating u_{ic}. The E- and M- steps are iterated until convergence, or until the number of iterations exceeds a certain threshold.

5.2.2. Partitional clustering with spatial information

Including spatial information, i.e. class information of neighboring pixels, may enable a clustering method to distinguish two clusters that are close together in feature space, but

far apart in the image. Moreover, it will smoothen the result. Although in many cases a somewhat noisy classified image may be very well interpretable by an expert, there are also cases where the noise seriously decreases the quality of the clustering. Furthermore, automatic assessment of the areas of the different clusters (by counting pixels) will be less reliable in the presence of noise or outliers. In all cases, by taking into account the spatial information, the overlapping problem in clustering is reduced.

5.2.2.1 Spatial information in deterministic partitional clustering

The spatial information of multivariate image can be taken into account by using the appropriate distances $\tilde{\wp}(x_i, \omega_j)$as in Eqs. 1 and 2.

The compactness function then becomes:

$$\tilde{E} = \sum_{j=1}^{g} \sum_{i \in C_i} (u_{ij})^q \tilde{\wp}(x_i, \omega_j) \tag{12}$$

In general, many spatial weight functions are possible. This concept has been applied in [19][32] for fuzzy C-means. For example of the additive inclusion of spatial information, in robust fuzzy C-means (RFCM) [32], the distance function $\tilde{\wp}(x_i, \omega_j)$ is defined as:

$$\tilde{\wp}(x_i, \omega_j) = \left[\wp(x_i, \omega_j) + \frac{\beta}{2} \sum_{l \in \partial i} \sum_{m \in C \setminus \omega_j} (u_{lm})^q \right] \tag{13}$$

where u_{lm} is the conditional probability of pixel xl in the neighbor-scheme ∂_i belonging to cluster m, which is not ω_j. The parameter β is a positive spatial dependency parameter. Larger values of β encourage neighbors to be in the same cluster. RFCM is identical to standard FCM when $\beta=0$.

As an example of the multiplicative inclusion of spatial information (eq. 5), a spatial weight function is defined in this paper as:

$$w(x_i, \partial_i, w_j) = \sum_{l \in \partial i} e^{\left(-\beta . u_{lw_j}\right)} \Big/ \sum_{l \in \partial i} \sum_{c=1}^{g} e^{(-\beta . u_{lc})} \tag{14}$$

The parameter β, again, is a positive spatial dependency parameter. Larger values of β encourage neighbors to be in the same cluster. The resulting modification of the standard fuzzy C-means clustering is called Spatial Conditional FCM (SCFCM) clustering. Figure 8 illustrates the effectiveness of the integrating of the spatial information into clustering of SYN and MEAT data by SCFCM. The SYN result (Figure 8a) is consistent with the design of the image data. In Figure 8b, the result on MEAT data, the dark meat regions are larger and the fat regions coincide with regions of light spots in the original image (Figure 2).

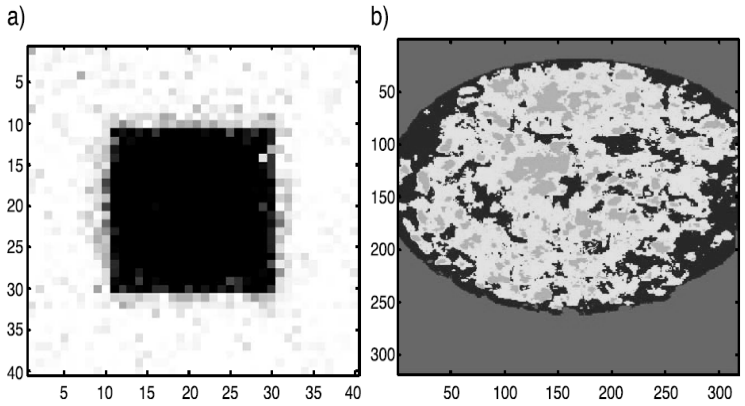


Figure 8: Clustering result using Spatial Conditional FCM (SCFCM) with spatial information. (a) SYN data; (b) the MEAT data.

GGC-FCM (Geometrically Guided Conditional FCM) is another example in [19]. It follows the general multiplication form $\tilde{\wp}(x_i, \omega_j) = k_i \wp(x_i, \omega_j)$ described in [33], where k_i corresponds to the "condition" for the pixel x_i, which is equivalent to the spatial weight $w(x_i, \partial_i, \omega_j)$. This condition value is determined by the majority class of neighboring pixels in ∂_i. More discussion of the condition value is in [19].

5.2.2.2 Spatial information in model-based clustering

Similar to deterministic partitional methods, the spatial continuity weight function $w(x_i, \partial_i, \omega_j)$ can also be included in model-based approaches. Probably the most often used weight function is based on Markov Random Field (MRF) theory [13][34]. Given the neighbor-scheme ∂_i, the simplest weight function can be defined for a model-based clustering as follows:

$$w(x_i, \partial_i, w_j) = e^{\left(\beta \cdot \sum_{l \in \partial i} u_{lw_j}\right)} \Big/ \sum_{c=1}^{g} e^{\left(\beta \cdot \sum_{l \in \partial i} u_{lc}\right)} \tag{15}$$

where β is the spatial continuity parameter. More positive values encourage neighbors to be of the same cluster. Hence, the new similarity function $\tilde{\wp}(x_i, \omega_j)$ can be formed, usually as in Eq. 5. Thus, the product of the weight w and the likelihood is maximized. The weight w approaches 1 if all neighboring pixels are in the same class as x_i, otherwise it is smaller. Research in the field was very active after the work in [35]. The same author proposes the famous ICM (Iterated Conditional Modes) algorithm [13]. ICM estimates the maximum of the marginal probabilities, which is equivalent to the optimizing the log-likelihood $L(\Psi)$. This is actually the conventional EM algorithm using the estimated conditional probability taking into account the spatial information. More detail on the modification of the EM algorithm taking into account the changes of posterior probability is in [23].

The neighbor-scheme ∂_i and the smoothness parameter β are often manually chosen. Automatic adjustment is also possible [13][34]. It is not very difficult to find good setting values for a small image or a small area. However, for a large image, presenting many different types of objects or structures, there may be no single parameter value for which good results are obtained. In such a case, many different local parameter values may be needed, and a multi-scale and a multi-resolution approach are needed [36][37].

For demonstration purposes, clustering results using ordinary MBC (using no spatial information) and ICM are reported in Figure 9. Clustering results are compared with the reference information in Figure 3b (not the Yellow area). The ICM algorithm, taking into account the spatial information of the image, shows better results. Not only is the agreement with the ground-truth higher but also the image looks much smoother. The parameters used in ICM are β=0.2 and ∂_i to be a square window of 11 x 11 pixels centered at x_i.

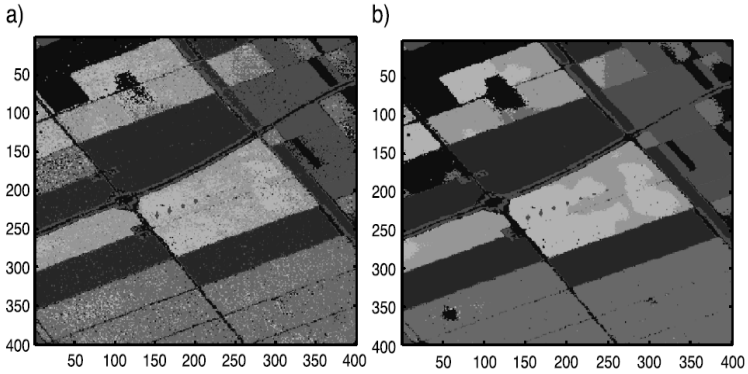


Figure 9. Clustering result of SAR image a) The best MBC after 50 random initializations with 71% accuracy, b) The best ICM after 50 random initializations with β=0.2 with accuracy of 81%, on the area having reference information (not the Yellow area in Fig 3b).

5.3. Agglomerative hierarchical clustering

Agglomerative hierarchical clustering (AHC) mostly refers to a hard deterministic hierarchical clustering. This yields a hierarchical structure of clusters, which represents how cluster pairs are joined. Conceptually, it is a simple idea that follows naturally from the concepts of distance and similarity [4][9][15][28][29]. In principle, the algorithm is as follows:

Algorithm:

(1) Start: Assign each pixel to an individual cluster, yielding N clusters.

(2) Iteration step: The similarities between all cluster pairs *i* and *j*, $\wp(\omega_i, \omega_j)$, are calculated and the two 'closest' clusters are merged.

(3) End: The algorithm ends if there is only one cluster.

Several variants of AHC exist: single linkage, complete linkage, average linkage, centroid linkage, and Ward's clustering, depending on the definition of the distance between clusters. In single linkage, the distance between two clusters is the distance between their two nearest points. Similarly, the distances are the maximal distance between points, the average distance of points, and the distance of mass centers in complete linkage, average linkage, and centroid linkage, respectively. Again, the distances can be the Euclidean, Manhattan, or more generally Minkowski distances. In Ward's clustering, the distance between two clusters, i and j, is the weighted version of the squared Euclidean distance of the cluster mean vectors;

$$\left(n_i n_j \Big/ n_i + n_j \right) \wp_{eucl}(\mu_i, \mu_j) \tag{16}$$

where n_i, n_j and μ_i, μ_j are the numbers of points and means of cluster i and j, respectively.

The result of AHC is a dendrogram, representing nested clusters and the similarity levels where clusters are joined. The dendrogram can be cut at several levels in order to obtain any number of clusters. This property makes it easy to compare many different numbers of clusters. However, determining a 'good' number of clusters is difficult. Several criteria will be mentioned below. Visualization of a dendrogram is only useful in a small data set, although in the field of microarray data analysis large dendrograms are often shown, e.g. analysis of gene expression profiles in [5][38].

In contrast to partitional clustering, AHC methods are very stable. There are two reasons. First, clustering is always initialized in the same way. Secondly, the algorithm considers only clusters that were obtained in the previous step. This means that once a point has been merged to a cluster, it can not be considered for joining other clusters in later iterations. In some cases, it is an advantage but it also decreases the flexibility, a drawback of AHC. "Chaining", or "friends-of-friends", is a term for a typical problem of single linkage AHC, when series of smaller clusters are merged into an elongated chain.

AHC works on the distance matrix at every iteration. The size of the distance matrix, the square of the number of objects, can be very large. AHC therefore is very susceptible to the "image size problem". Because of this problem, AHC is rarely applied to an image data set.

If the data set contains noise, or outliers, these are kept in separate clusters and do not influence other clusters. In this case, the real number of clusters can only be defined after the clusters containing noise/outliers, which are normally very small in size, are eliminated [16].

These characteristics can be demonstrated by applying AHC to the SYN image. As expected, the single-linkage obtains one very small cluster containing only outliers, shown in Figure 10a. The results of complete linkage and average linkage are given in Figure 10b and c.

The AHC concept can be extended to a model-based variant where the *classification likelihood* is used [22][30]:

$$L_{CL} = \prod_{i=1}^{n} f\left(x_i ; \theta_{c_i}\right) \tag{17}$$

where f is a multivariate distribution with parameters θ_{ci} for cluster c_i to which x_i is assigned. Model-based agglomerative hierarchical clustering operates by successively merging pairs of clusters corresponding to the greatest increase in the classification likelihood L_{CL}. This method is equivalent to the Ward's AHC method when f is multivariate normal with uniform spherical covariance [30].

Just like integrating spatial and spectral information with partitional clustering, spatial information conceptually can also be used in AHC. Distances can thus be estimated by $\tilde{\wp}\left(x_i, \omega_j\right)$ at any iteration, taking into account the spatial weight function. There still is no report of this kind of work for AHC so far.

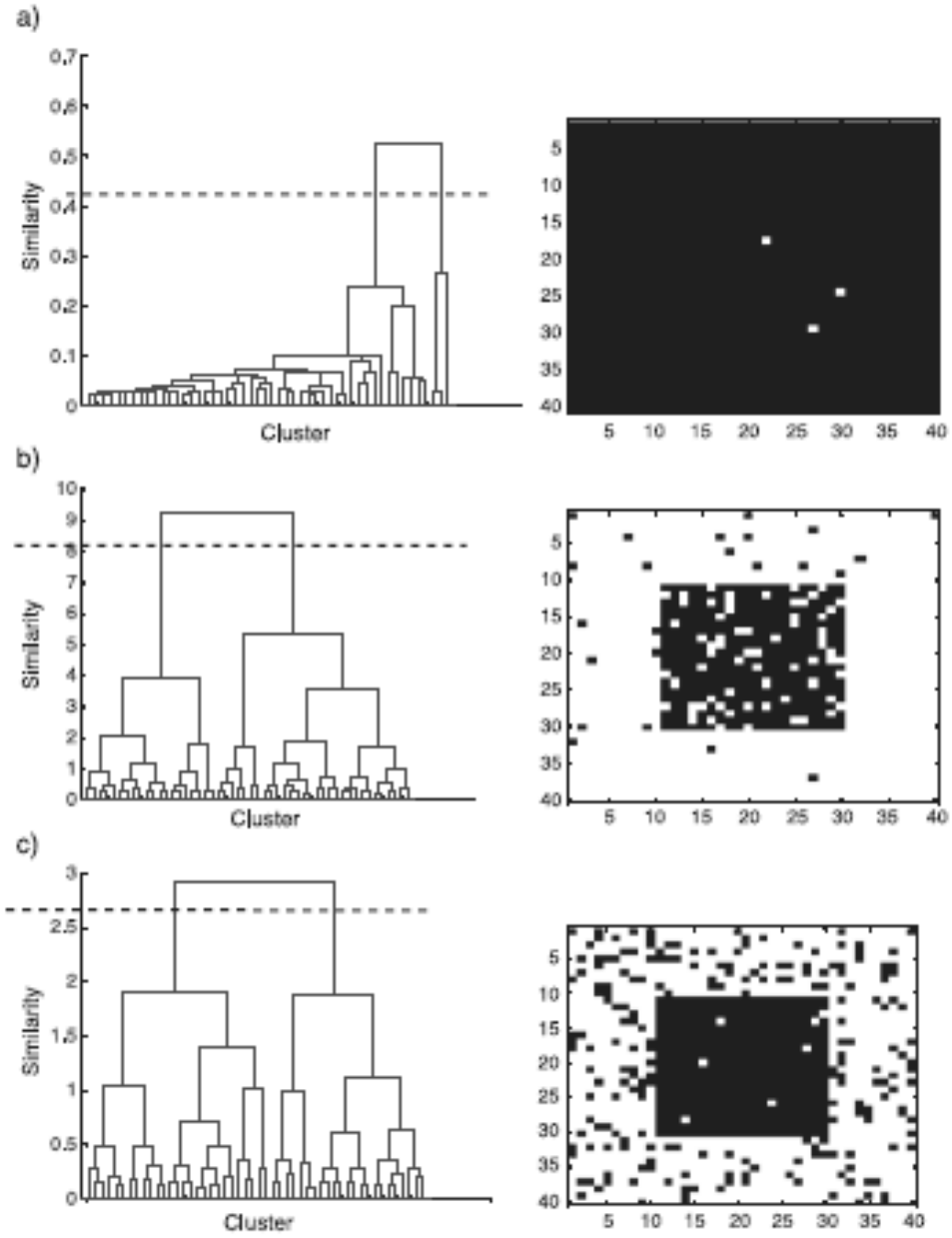


Figure 10. AHC clustering applied to the SYN image, a) single linkage, b) complete linkage, c) average linkage. The horizontal line indicates the cutting level to obtain two clusters.

5.4. Density-based method

Besides the hierarchical and the partitional approaches, density-based clustering methods, such as Denclust [39], CLUPOT [40] and DBSCAN [41], form a third clustering type. Density-based clustering estimates densities around individual objects. It is basically a hill-climbing procedure to a local density maximum [42]. Each local maximum then constitutes a cluster, and the cluster boundaries are given by the low density areas (valleys). They are determined by a density threshold. This, together with the size of the volume for which the local density is estimated, are the two main parameters of the method. Once these parameters are set, the number of clusters automatically follows. Density-based clustering was first presented as 'mean-shift' or 'mode seeking' methods,

based on an estimation of a gradient of local density functions, proposed in [43] and further improved in [42].

Basically, the density estimation for a particular point, x_i, is given by the number of points in a particular volume around that point, V_{xi}. Variable kernel methods use a kernel function $K(x_i)$ to give more weight to points close to x_i. Gaussian and triangular kernels are often used. A good review of non-parametric density estimation methods can be found in [1].

Density-based clustering was originally designed to detect clusters of arbitrary shape and to isolate noise, and in these aspects, it has advantages over other clustering methods. However, it was shown that current density-based clustering fails to identify both large and small clusters simultaneously due to very different densities [17]. Low density clusters tend to be assigned as noise/outliers. The method spends most of the computation time for computing the density estimation function for each object, which is very demanding. This feature prohibits the density-based method to be applied to multivariate image data. This is in contradiction to conclusions from [24]. Moreover, determining the right parameters for density-based clustering method can be challenging. There is no good way to identify them automatically and in practice it is a 'trial and error' strategy. Density-based clustering in general also has problems with overlapping clusters. The area of overlap often has a higher density than the neighborhood areas. This feature prohibits density-based clustering to separate two overlapping clusters, but tends to merge them together or to create a new cluster for the overlapped region. Those are the main reasons why density-based clustering is not widely used for multivariate images.

Graph-based clustering [44][45] is a special case of density-based clustering. In graph-based clustering, pixels, nodes in a graph, are connected based on a neighborhood function. A weak link is defined by a 'low' number of neighbor links. The clustering process is then a spanning process to identify a group of connected nodes when all weak links are broken (disconnected). The strength of the link is analogous to the density function.

DBSCAN and OPTICS [41] are well-known density-based clustering methods that have been applied recently in chemometrics [24]. Denclust [39] is generalization of DBSCAN using a gaussian kernel. A fixed volume is used in this case, and the density threshold is defined by two parameters, "ε" and "minpts", the radius of the volume centered at a particular point, and the minimum number of points in the volume, respectively.

The properties of density-based clustering are illustrated by a simple example of applying DBSCAN to the SYN image data in Figure 11.

Figure 11a shows the result and the density plot (value is the number of points in the volume without normalization to the total volume) when DBSCAN is applied with a high density threshold, $\varepsilon = 0.27$ and minpts = 100. Pixels in the second cluster are classified as noise. The best clustering result is obtained with a lower density threshold, $\varepsilon = 0.19$ and minpts = 70, which yields two clusters in Figure 11b. If the density threshold is even lower, $\varepsilon = 0.13$ and minpts = 50, three clusters are obtained, where the overlapping area shows a peak to form a separate cluster, Figure 11c.

There is no report showing the integration of spatial information with density-based clustering. In general, it is harder in this case because the density-based clustering does not use distances which can be extended with spatial information. To include the spatial information, the new estimated density function has to be calculated using a new feature space which is an extension of the original space. The parameters in this case will even more difficult to estimate.

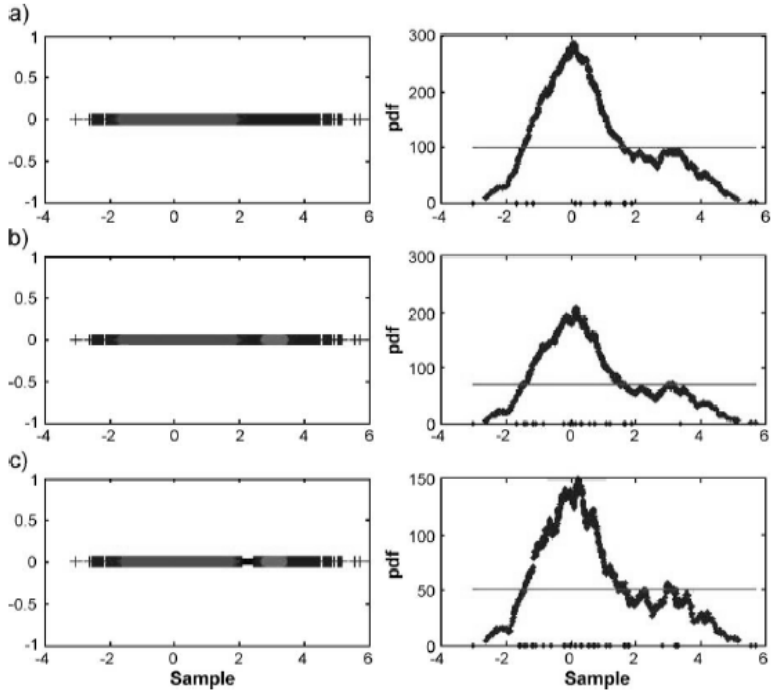


Figure 11. Application of DBSCAN to the SYN data set, on the left side showing results in the 1D feature space (Red: first cluster, Pink: second cluster, Black: third cluster, and Blue: noise. On the right side, the density plot is shown (the value is the number of points in the volume without normalization to the total volume): (a) one cluster obtained with $\varepsilon = 0.27$ and minpts = 100 (the red line), (b) 2 clusters obtained with $\varepsilon = 0.19$ and minpts = 70, and (c) 3 clusters obtained with $\varepsilon = 0.13$ and minpts = 50.

5.5. Choosing a good number of clusters

Not many clustering algorithms can provide a good number of clusters automatically. In many cases, the user needs to define the number of clusters, either directly, in the partitional clustering, or indirectly, in the hierarchical and density-based clustering. In general, a good number of clusters can be obtained by running an algorithm many times with a different number of clusters and comparing results with a criterion. The most popular criteria for deterministic partitional clustering are the Davies-Bouldin, Dunn, C-, and Goodman-Kruskal indices [46]. In Model-based clustering, on the other hand, the optimal number of clusters corresponds with the best fit of the data. AIC (Akaike's Information Criterion) and BIC (Bayesian Information Criterion) are the most popular criteria for mixture model clustering [23]. For example, in hierarchical model-based clustering [30], the BIC criterion is used to find an appropriate cutting level and the best number of clusters as the result. If the clusters can be described by normal distribution, these indices often perform very well. There is no useful criterion to determine the number of cluster to density-based clustering.

In many cases, a number of clusters of the multivariate image can also be determined by visualizing the clustering result using some priori knowledge about structure presenting in the image surface. This technique has been applied frequently, e.g., in detecting of brain tumors in MRI images [14].

The only criterion to date to take into account spatial information is the PLIC criterion [47], an extension of the BIC criterion. The conditional likelihood is estimated locally, i.e. from the immediate spatial neighborhood of the pixel. This criterion can be used for model-based clustering such as ICM clustering.

5.6. Application to image data

Image data is normally quite large and contains noise/outliers and overlapping clusters. A 'soft' partitional clustering is a good option if a number of clusters and a good initial set $u_{ij} \in [0,1]$ are available. Unfortunately, this is often not the case and the result may be very dependent on the random initial state. AHC, on the other hand, is more stable but hard to apply to image data directly because of the image size problem. Hence, AHC is normally used to estimate the correct number of clusters and their parameters using a small representative subset of the image data [30]. The problem then becomes how to obtain this subset. The simplest solution is to generate this set randomly from the whole data set. However, there is a real danger of missing a small cluster and more complicated methods may be needed [48]. Another option is to apply a partitional clustering to obtain a large number of cells, which are then joined together by AHC. This is much cheaper than starting AHC from singletons [16]. Both density-based and AHC clusterings suffer from time complexity and computer resource problems. In most of the cases, they are preferred for a small multivariate image, such as in MRI images [14][15]. Although AHC is still applied for larger images, e.g. in analysis of gene expression profiles in [5][38], or electron probe X-ray microanalysis in [6].

6. Pre- and Post-processing

In many cases, accuracy may improve by performing pre-processing of the raw image data or post-processing of the clustering result. However, the effectiveness depends on the clustering method and the particular image data. Preprocessing methods include smoothing techniques to decrease the amount of noise in the image data and dimension reduction techniques to decrease the computational demands. In post-processing, the most often used technique is a smoothing, performed by a majority-voting procedure. Again this mainly serves to decrease noise in the clustering result.

6.1. Noise/Outliers

One of the most often used pre-processing techniques is to remove unwanted noise/outliers from a raw multivariate image. It is an important task and necessary for clustering methods that are sensible to noise/outliers such as K-means. It is normally called a spatial filtering (low pass filtering) or a smoothing method. Many spatial filtering techniques have been proposed for gray images, based on local averaging of a mean intensity value on a local neighborhood at each image pixel. Linear filterings, such as Mean Value Smoothing and Median Filtering, are the most popular methods. Readers are referred to [49] for a complete review of image filtering methods. In the most simple case, these filtering techniques can be extended to multivariate image by performing filtering on each variable (parameter) individually. However, because they rely on only a raw data set without any knowledge of underlying structure, these techniques tend to displace structures and blur their boundaries. This side effect critically influences many clustering methods. Thus, this filtering technique is recommended only when the image does not contain many boundaries. In this case, a simple filtering method such as Median filtering can do the job. Otherwise, it is recommended to use clustering methods which can deal with noise, such as MRF model-based clustering. As an example, the SYN data is filtered by the Median filtering and clustered using the fuzzy C-means algorithm as in Figure 12a-b. The result is much better compared the result on the original raw image data (Figure 6b).

6.2. Dimension reduction

The dimensionality has to be kept as small as possible to improve the clustering performance due to the high feature dimension problem. In many cases, not all feature variables are important in clustering and one may select a subset of variables that together still capture most information of the image data. Moreover, calculating distances taking into account many uninformative variables may totally obscure cluster structure. Prior knowledge may help to decide which wavelength to use. In other cases, selection of features is an optimization problem, for which methods such as SA (Simulated Annealing), GA (Genetic Algorithm), or Tabu Search can be applied [50][51]. Projection methods form an alternative for feature selection. Linear transformations, such as PCA (principal component analysis), ICA (independent component analysis), and non linear mappings such as SOM (Self-Organizing Map), Nonlinear PCA, have been widely used [30][50]. The original feature space is then mapped to a latent space, in which the number of latent features is small and suitable for clustering algorithm. However, the structure of the cluster may be changed, sometimes in such a way that clusters disappears or start to overlap [52]. Note that both feature selection and projection methods do not take into account spatial information.

6.3. Filtering of clustering result

Filtering is not only used as a pre-processing step but can also be applied in the post-processing of a clustering result. The only difference is that the pre-processing takes into account the whole information on the raw image data but the post-processing takes only the clustering result into consideration. Point intensities are then replaced by clustering labels. Noise/outliers are also considered as members of clusters. Hence, depending on the clustering algorithm, if noise/outliers are still present in the image, they will be smoothed by this filtering. For mixed points in an area of cluster overlap, such as in the SYN data set, intensities are not changed much by applying a pre-processing filtering technique when spatial neighbor points are also in this overlapping area. This is often the case when the window size is small. This problem is less for the filtering applied as a post-processing method when the neighborhood intensities are replaced by the cluster labels. These properties are illustrated on SYN data set in Figure 12a-c. Figure 12a shows the pre-processing image and the followed clustering result by the fuzzy C-means is in Figure 12b. The result is compared with the situation where the fuzzy C-means is applied directly on the original SYN image and the clustering result is post-processed by filtering as in Figure 12c. As expected, the clustering result is better compared to the result using filtering as pre-processing.

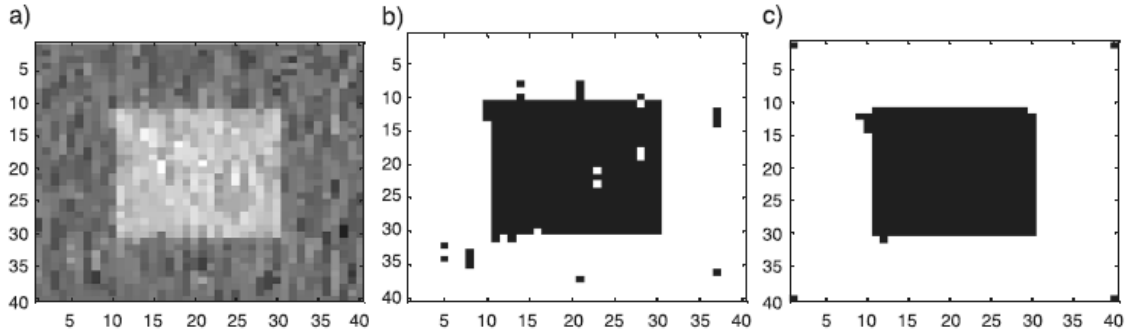


Figure 12. Pre- and Post processing on the SYN image, (a) image data after pre-processing by Median filtering with a 3x3 window , (b) the fuzzy C-means result of clustering the preprocessed data, (c) post-processing result of fuzzy C-means applied to the original SYN image using median filtering of clustering result with a 3x3 window.

The same scenario is performed on the SAR image as in Figure 13a-c. The Median filtering with a 5x5 window is used for both pre- and post processing. The fuzzy C-means with post-processing of the clustering is better than using only pre-processing technique. Accuracies are 77% and 64%, respectively. The fuzzy C-means without any pre- or post-processing achieves only 51%.

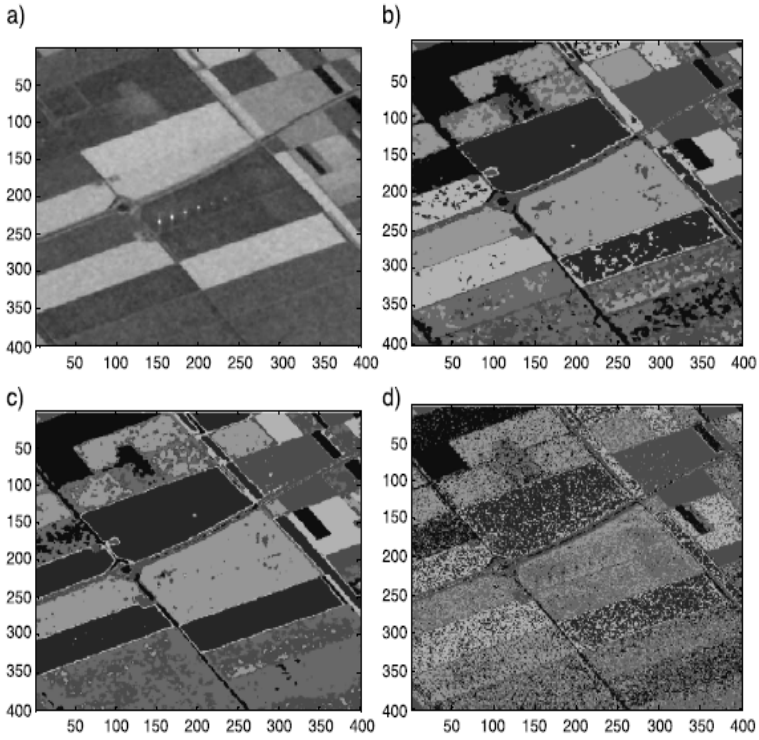


Figure 13. Pre- and Post processing on the SAR image, (a) false-color image of the SAR image data after pre-processing using Median filtering with a 5x5 window, (b) fuzzy C-means applied to the filtered image data yielding an accuracy of 64%, (c) the result after post-processing filtering of fuzzy C-means applied to the original image data, with an accuracy of 77%, (d) the fuzzy C-means result applied to the original image data with accuracy of only 51%.

7. Conclusion

This tutorial provides a broad survey of the most basic clustering techniques to multivariate image. The tutorial gives guidelines to determine the most relevant clustering for a particular multivariate image data set, depending on the list of "image data problems". In many cases, partitional clustering techniques taking into account spatial information form the best option for a large image, provided the number of clusters is known or can easily be estimated. The situation is more difficult if this information is unknown. Then, the process of trial and error using statistical criteria and visualization is an option. Careful pre- and post-processing can reduce the effect of noise/outliers and overlapping clusters. However, incorrect use of these techniques can disturb or blur structures in the image. Instead, using clustering techniques taking into account spatial information can deal better with these situations.

Some problems are still remaining for clustering multivariate images. A good clustering for a particular image using spatial information needs to have a good setting of parameters. Automatic settings do not always give a good result. In many cases, the setting can be obtained by a "trial and error" strategy and personal experience. This work is more difficult for a larger image, when more than one set of parameters may be required.

Furthermore, clustering multivariate images always has to deal with the huge data problem because the development of image scanner technology at the moment is often faster than computer technology. Validating the clustering result is another problem, due to the lack of reference information.

8. Acknowledgements

We thank Jacco C. Noordam, Department of Production & Control Systems, Agrotechnological Research Institute (ATO), and Dirk H. Hoekman, Department of Environmental Sciences, Wageningen University, for sharing the data sets.

References

[1] A. Webb, Statistical Pattern Recognition, Wiley, Malvern, UK, 2002.
[2] K. Fukunaga, Introduction to Statistical Pattern Recognition, 2nd edition, Academic Press, London, 1990.
[3] A. K. Jain and M. N. Murty, ACM Computing Surveys, 31 (1999) 264 323.
[4] A. Smolinski, B. Walczak and J. W. Einax, Chemom. Intell. Lab. Syst., 64 (2002) 45 54.
[5] J. Liang and S. Kachalo, Chemom. Intell. Lab. Syst, 62 (2002) 199 216.
[6] I. Bondarenko, H. Van Malderen, B. Treiger, P. Van Espen and R. Van Grieken, Chemom. Intell. Lab. Syst, 22 (1994) 87 95.
[7] A. Linusson, S. Wold and B. Nordén, Chemom Intell. Lab. Syst., 44 (1998) 213 227.
[8] F. Ros, M. Pintore and J. R. Chrétien, Chemom. Intell. Lab. Syst., 63 (2002) 15 26.
[9] P. Teppola, S.-P. Mujunen and P. Minkkinen, Chemom. Intell. Lab. Syst., 45 (1999) 23 38.
[10] P. Teppola, S.-P. Mujunen and P. Minkkinen, Chemom. Intell. Lab. Syst., 41(1998) 95 103.
[11] U. Thissen, H. Swierenga, A.P. de Weijer, R. Wehrens, W.J. Melssen, L.M.C. Buydens, Multivariate Statistical Process Control Using Mixture Modelling, submitted for publication (2004).
[12] Stan Z. Li, Markov Random Field Modeling in Image Analysis, Springer-Verlag Tokyo, 2001.
[13] J. Besag, J. R. Statist. Soc. B, 48 (1986), 259-302
[14] R. Wehrens, A. W. Simonetti and L. M. C. Buydens, J. Chemom., 16 (2002) 274 282.
[15] D.L. Pham and J. L. Prince, IEEE Trans. on Medical Imaging, 18 (1999) 737 752.
[16] T. N. Tran, R. Wehrens and L. M. C. Buydens, Anal. Chim. Acta, 490 (2003) 303 312.
[17] T.N. Tran, R. Wehrens and L.M.C. Buydens, 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (URBAN 2003), Proceedings of the Conference, May 2003, Berlin, Germany, 2003, pp. 147 151.
[18] R. Wehrens, L.M.C. Buydens, C. Fraley and A.E. Raftery, Model-based clustering for image segmentations and large datasets via sampling, Techn. Report no. 424, Dept. of Statistics, University of Washington, 2003.
[19] J. C. Noordam and W. H. A. M. van den Broek, J. Chemom., 16 (2002) 1-11.
[20] J.C. Noordam and W.H.A.M. van den Broek, L.M.C. Buydens, Chemom. Intell. Lab. Syst., 64 (2002) 65 78.
[21] D.H. Hoekman and M.A.M. Vissers, IEEE Trans. on Geoscience and Remote Sensing, 41 (2003) 2881 2889.
[22] C. Fraley, SIAM J. Sci. Comput., 20 (1998) 270 281.
[23] G. McLachlan and D. Peel, Finite Mixture Models, Willey series in probability and statistic, John Wiley & Sons, Canada, 2000
[24] M. Daszykowski, B. Walczak and D. L. Massart, Chemom. Intell. Lab. Syst., 56 (2001) 83 92.
[25] J. Mao and A.K. Jain, IEEE Trans. on Neural Networks, 7 (1996) pp. 16-29.
[26] G.H. Ball and D.J. Hall, ISODATA, A novel method of data analysis and pattern classification, Techn. Rep., Stanford Research Institute, Menlo Park, CA, 1965.

[27] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.
[28] I. Gath and A. B. Geva, IEEE Trans. on Pattern Anal. Mach. Intell., 11 (1989) 773 781.
[29] L. Bobrowski and J.C. Bejdek, IEEE Trans. on Systems Man and Cybernetics", 21 (1991) 545 554.
[30] C. Fraley and A. E. Raftery, J. Am. Stat. Assoc., 97 (2002) 611 631.
[31] A.P. Dempster, N.M. Laird and D.B. Rubin, J. R. Statist. Soc. B, 39 (1977) 1 38.
[32] D.L. Pham, Computer Vision and Image Understanding, 84 (2001) 285 297.
[33] W. Pedrycz, Pattern Recognition Letters, 17 (1996) 625 631.
[34] D. Geman and S. Geman, IEEE Trans. Pattern Mach. Intell., 6 (1984) 721 741.
[35] J. Besag, The Statistician, 24 (1975) 179 195.
[36] W. Qian and D.M. Titterington, Philosophical Trans. R. Soc. of London A, 337 (1991) 407 428.
[37] I.V. Cadez and P. Smyth, Modeling of inhomogeneous Markov Random Fields with applications to cloud screening, Technical Report No. 98-2. Irvine: Department of Information Science, University of California Irvine, 1998.
[38] M. Eisen, P. Spellman, P. Brown and D. Botstein, PNAS, 95 (1998) 14863 14868.
[39] A. Hinneburg and D. A. Keim, Knowledge Discovery and Data Mining (1998). Proceedings of the Conference, 1998, pp. 58 65.
[40] D. Coomans and D. L. Massart, Anal. Chim. Acta, 133 (1981) 225 239.
[41] M Ester., H.-P. Kriegel, J. Sander and X. Xu, Knowledge Discovery and Data Mining, Processdings of the Conference, 1996, pp. 226 231.
[43] K. Fukunaga and L.D. Hostetler, IEEE Trans. Info. Theory, 21 (1975) 32 40.
[42] Yizong Cheng, IEEE Trans. Pattern Anal. Mach. Intell., 17 (1995) 790 799.
[44] G. Karypis, E.-H. Han and V. Kumar, IEEE Computer, 32 (1999) 68 75.
[45] S. Guha, R. Rastogi, K. Shim, Information Systems 25 (2000) 345 366.
[46] S. Günter and H. Bunke, Patt. Recog. Lett., 24(2003) 1107 1113.
[47] D.C. Stanford, A. E. Raftery, IEEE Trans. on Pattern Anal. and Mach. Intell., 24 (2002) 1517 1520.
[48] C. Fraley, A. E. Raftery and R. Wehrens, Incremental Model-Based Clustering for Large Datasets with Small Clusters, Techn. Rep. no. 439, Dept. of Statistics, University of Washington, Dec. 2003.
[49] P. Geladi, H. Grahn, Multivariate image analysis, Wiley, New York, 1996.
[50] A.K. Jain and D. Zongker, IEEE Trans. Pattern Anal. Mach. Intell., 19 (1997) 153 158.
[51] J.A. Hageman, R. Wehrens, H.A. van Sprang and L.M.C. Buydens, Anal. Chim. Acta, 490 (2003) 211-222.
[52] W. C. Chang, Applied Statistics 32 (1983) 267 275.

KNN-KERNEL DENSITY-BASED CLUSTERING FOR HIGH DIMENSIONAL MULTIVARIATE DATA

Abstract

Density-based clustering algorithms for multivariate data often have difficulties with high dimensional data and clusters of very different densities. A new density-based clustering algorithm, called KNNCLUST, is presented in this paper that is able to tackle these situations. It is based on the combination of nonparametric k-nearest-neighbour (knn) and kernel (knn-kernel) density estimation. The knn-kernel density estimation technique makes it possible to model clusters of different densities in high-dimensional datasets. Moreover, the number of clusters is identified automatically by the algorithm. KNNCLUST is tested using simulated data and applied to a multispectral Compact Airborne Spectrographic Imager (CASI) image of a floodplain in the Netherlands to illustrate the characteristics of the method.

Keywords: Multivariate data; classification; clustering;

T.N. Tran, R. Wehrens and L.M.C. Buydens, *Proc. 2nd GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas, URBAN_2003*, May 2003, Berlin, Germany.

Revised for *Journal of Computational Statistics and Data Analysis*

1. Introduction

Clustering of multispectral data [1] groups objects, characterized by the values of a set of variables into separate groups (clusters) with respect to a distance or, equivalently, a similarity measure. Its objective is to assign to the same cluster objects that are more close (similar) to each other than to objects from different clusters, which may help to understand relationships that may exist among objects. Examples are exploring of environmental data representing physical and chemical parameters [2], computational analysis of microarray gene expression profiles [3], electron probe X-ray microanalysis [4], or process monitoring [5], and many others. However, the successful application of clustering on multispectral datasets is not a straightforward task. It depends on the understanding of the dataset and a good choice of the clustering algorithm.

Several types of clustering methods can be distinguished, among which partitional and hierarchical approaches are the most common [1]. Density-based clustering methods, such as CLUPOT [6], DBSCAN [7], and Denclust [8], form a third clustering type. Density-based clustering uses a local cluster criterion, in which clusters are defined as regions in the data space where the objects are dense, and clusters are separated from one another by low-density regions. Non-parametric density-based clustering is based on an estimation of a local non-parametrics density function, proposed by Fukunaga and Hostetler [9] and been further improved in [10][11]. Density-based clustering has advantages over partitional and hierarchical clustering methods in discovering clusters of arbitrary shapes, sizes. It is often used in data mining for knowledge discovery.

However, it was shown that current density-based clustering might have difficulties with complex data sets containing clusters with different densities [11]. In this case, it often identifies the very low density classes as noise [1]. Moreover, the high dimensionality of many multivariate data sets is another problem for density-based clustering. In this case, the volume of the data grows dramatically with the dimension, while the number of objects remains the same. One of the solutions for the dimensionality problem is proposed in [12][13], using a k-nearest-neighbor density estimation technique. Instead of defining a threshold to local density function, low-density regions, “valleys”, separating two clusters can be detected by calculating the number of shared neighbors. If the number of shared neighbors of two adjacent objects is below a threshold (number of objects), then there is a gap, a valley, in between. Hence, the two object belong to two different clusters. In this way, the method does not have to take into account the volume of the high dimensional search space. However, this clustering method still requires the “density” threshold to be defined, which is very difficult for a real dataset [14].

In this paper, a new density-based clustering algorithm, the so-called KNNCLUST, is developed. The proposed method is based on a combination of nonparametric k-nearest-neighbour (knn) and kernel density estimation methods (knn-kernel). It will be shown later in the text that knn-kernel is not a good solution for estimating the ‘true’ density of a distribution due to an overestimate of density in the tails of the distribution. However, the knn-kernel has attractive properties to clustering, shown for the first time in this paper. KNNCLUST has been implemented in MATLAB 6.5 and the toolbox is available on the web [15].

We review nonparametric density estimation techniques in section 2. The knn-kernel class-condition rule on clustering and the description of the new knn-kernel density-based clustering, KNNCLUST, are given in section 3. In section 4, its properties are illustrated

using a multispectral remote sensing image and compared to the results from DBSCAN. Finally, the work is summarized in section 5.

2. Knn-Kernel Density Estimation

An unknown probability density function of a data set can be estimated by a nonparametric kernel density estimation method. Consider a N x d dimensional data set. The d-dimensional space can be partitioned into a number of equal bins (volumes), V, e.g. hyper-rectangles. The multivariate kernel density estimate obtained at the object x with kernel K is defined as [16]:

$$\hat{f}(x) = \frac{1}{NV}\sum_{1}^{N} K((x - x_i)./H) \tag{1}$$

The size of the bin is given by a scale vector $H=[h_1..h_d]$ in d-dimensional space, and the matrix operation './' is the element-by-element division of two equal-sized matrices or vectors. The data volume V is $\prod_{i=1}^{d} h_i$. A list of common kernels is given in Table 1. A Triangular or Gaussian kernel function is normally used (Fig. 1).

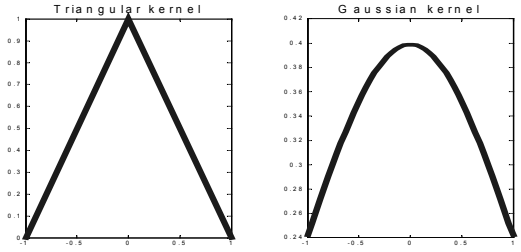


Figure 1. Triangular and Gaussian kernels

In Eq. 1, bin V is fixed in size. If bin V is defined just like in k-nearest-neighbor (knn) [16], where the volume around object x, V_x, is adjusted to include the k nearest neighbor objects, the method is called knn-kernel and given by:

$$\hat{f}(x) = \frac{1}{NV_x}\sum_{1}^{N} K((x - x_i)./H_x) \tag{2}$$

where H_x is a scale vector $[h^x_1..h^x_d]$ of the volume V_x in d-dimensional space.

Table 1. Commonly used kernels, where $z = (x - x_i)./H$.

Rectangular	½ if $x^Tx < 1$, 0 otherwise
Triangular	1- |x| if $x^Tx < 1$, 0 otherwise
Biweight	$\frac{15}{16}(1 - x^Tx)^2$ if $x^Tx < 1$, 0 otherwise
Gaussian	$\frac{1}{\sqrt{2\pi}}\exp(-x^Tx/2)$
Bartlett-Epanechnikov	$\frac{3}{4}(1 - x^Tx/5)\sqrt{5}$ if $x^Tx < \sqrt{5}$, 0 otherwise

The idea of knn-kernel is first introduced by Loftsgaarden and Quesenberry [17] and then generalized by Terrell and Scott[18], where the Euclidean distance or Mahalanobis distance to the k^{th} nearest neighbor is used. Here, we use the volume V_x, which is more

general. Knn-kernel can also be seen as a case of variable kernel density estimation methods [18][19].

Knn itself obviously is a simply case of knn_density estimation where the uniform kernel is used. Readers are referred to [16] for a complete overview of nonparametric kernel density estimation methods. The knn-kernel method has two advantages over other methods. Without the kernel, the first arises from density estimate is non-smooth; using a kernel makes the knn-kernel estimator smooth. The second advantage is the result of the application of knn and allows for an adaptive kernel width: a broader kernel in low density regions and a narrower kernel in high density regions. Comparing with fixed kernel width methods, abnormal small density peaks appear in low density regions (e.g. in Figure 3a), which will result in many small clusters found with ordinary density-based clustering. Hence, the knn-kernel method is useful for clustering, even though it is not better than the fixed kernel scheme for the purpose of estimating a density, due to an overestimate of density in the tails of the distribution [18].

These features are demonstrated in Figure 2, where the knn and the knn-kernel methods are applied to a synthetic data set, which includes 500 objects generated from one Gaussian distribution.

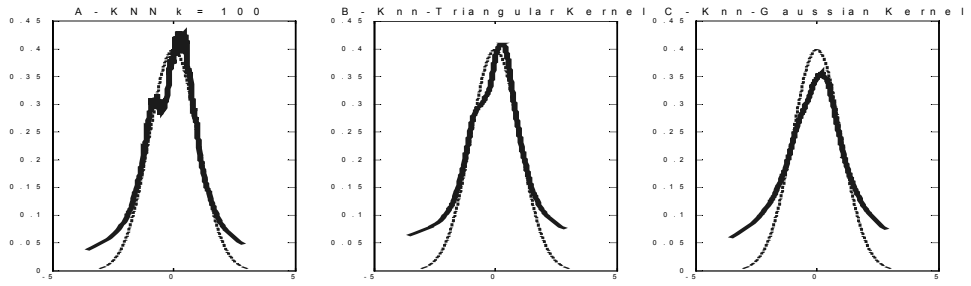


Figure 2. Knn and Knn-kernel estimation on the sample data set containing 500 samples generated from one Gaussian distribution (mean=0 and s=1), with k = 100. The dotted line is the theoretical pdf function for the data set.

The other simple example in Figure 3 shows the advantage of the knn-kernel on a data set containing two classes of different densities. Class one is a high density class containing 500 objects generated from one Gaussian distribution (mean=0 and s=1). Class two is a low density class containing 150 objects generated from one Gaussian distribution (mean=100 and s=10). The kernel-based estimation method (Eq. 1) provides a smooth estimate for the first class but a bad estimate for the second class, showing many sharp peaks, due to the aforementioned problem of the kernel-based method (Figure 3A). In contrast, the knn-kernel method (Eq. 2) with k = 100 provides a smooth density estimate for both classes (Figure 3B).

In general, nonparametric methods are sensitive to the choice of the smoothing parameter. If it is too small, the density estimate is too detailed, showing many sharp peaks (as in Figure 3A, the kernel method for cluster 2). If it is too large, the structure of the density function is lost. Hand [20] showed that the smoothing parameter can be estimated from the average distance of k nearest neighbors. The knn-kernel method, on the other hand, forms a flexible way to deal with a complex data set, where densities can be very different between clusters. Then, the smoothing parameter values are adapted locally for different clusters.

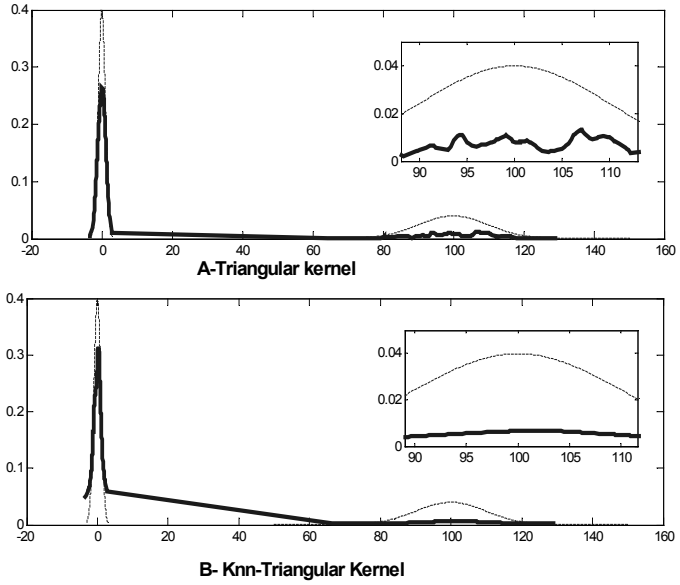


Figure 3. Density estimation functions for the data set of two classes of different densities.

3. Knn-Kernel Density-Based Clustering

3.1 Classification rule based on knn-kernel density estimates

The most common ways to assign objects to clusters, also called classification rules, are based on Bayes' decision rule:

$$p(x|\omega_i)p(\omega_i) > p(x|\omega_j)p(\omega_j), \forall j \neq i \tag{3}$$

where $p(x|\omega_i)$ is the class-conditional density function at x of each class ω_i and $p(\omega_i)$ is the prior probability function. The class-conditional density function can be estimated by the nonparametric knn-kernel method, mentioned earlier:

$$\hat{p}(x|\omega_i) = \frac{1}{n_i V_x} \sum_{x_j \in \omega_j} K\left((x - x_j)./H_x\right) \tag{4}$$

where n_i is the size of cluster ω_i, and $\Sigma\, n_i = N$. Bayes' knn-kernel class-condition can be rewritten as:

$$\frac{1}{n_i V_x}\left(\sum_{x_l \in \omega_i} K\left((x - x_l)./H_x\right)\right)p(\omega_i) > \frac{1}{n_j V_x}\left(\sum_{x_l \in \omega_j} K\left((x - x_l)./H_x\right)\right)p(\omega_j), \forall j \neq i \tag{5}$$

The prior probability functions $p(\omega_i)$ and $p(\omega_j)$ are normally estimated by n_i/N and n_j/N, respectively. Then, the knn-kernel Bayes' class-condition can be simplified:

$$\sum_{x_l \in \omega_i} K\left((x - x_l)./H_x\right) > \sum_{x_l \in \omega_j} K\left((x - x_l)./H_x\right), \forall j \neq i \tag{6}$$

Thus, the decision rule used here is the same as the one in the knn classifier [16] in the supervised classification method, but the density estimation is replaced by the knn-kernel. The advantage of this for clustering is illustrated in the following section.

3.2. The KNNCLUST algorithm

We propose in this section KNNCLUST as a "hard" clustering algorithm, which assigns each object x_i to one and only one cluster. Just like partitional clustering [1], which "seeks an organization of objects which optimizes a target function" [1], KNNCLUST forms clusters in order to maximize the total class-conditional density function for all objects defined by:

$$D = \sum_{i=1}^{N} \hat{p}(x_i|c) \tag{7}$$

where the point x_i is assigned to cluster c.

The framework of KNNCLUST is as follows:

Steps of the algorithm:

1. Start: *N* singleton clusters, the number neighbors *k*, and the knn table *T* of size (*N* x *k*) , the list of k nearest neighbors of all samples.

2. Iteration: re-calculate cluster memberships of all points using the class-condition (Eq. 6) in order to maximize the function D.

STOP: if no, or only a few cluster memberships change (stop-condition).

Otherwise LOOP and start new iteration (step 2).

Using the knn-kernel Bayes' class-condition (Eq. 6), in step 2, $\hat{p}(x_i|c)$ is replaced by $\hat{p}(x_i|d) = \max(\hat{p}(x_i|j)|\forall j \in C)$ for all points. The old membership c is replaced by new membership d of object x_i. At the end of iteration, there may be an empty cluster because all points were moved to other clusters. This cluster is removed from the system and the total number of clusters is decreased by one. The algorithm ends if the stop-condition is fulfilled. Note that $\hat{p}(x_i|c)$ never decreases at any stage. Therefore, eventual convergence is assured.

In KNNCLUST, only the triangular kernel is recommended for the kernel function K in knn-kernel Bayes' class-condition (Eq. 6) to reduce computation time. Using the Gaussian kernel gives similar results but is more time consuming. The rectangular kernel (equivalent to the well-known knn class-condition, often-used in supervised classification) is not used here. It leads to problems in the initial state where the knn estimated density values at any point are equal for all clusters.

A simple example in Figure 4 shows how KNNCLUST performs on a simple data set of eight object values in a 1D space with k = 2. Each row in Figure 4 plots objects in one particular step of the process when the membership is changed. For example, iteration one starts with object one, x1. Because k = 2, the width of the bin around x1 is given by

$H_{x_1} = x_3 - x_1$.

By applying the triangular kernel we obtain

$\hat{p}(x_1|*) = 1 - \left|(x_1 - x_2)/H_{x_1}\right|$,

$\hat{p}(x_1|\mathrm{o}) = 1 - \left|(x_1 - x_3)/H_{x_1}\right|$

It is obvious that (the class-condition Eq. 6), so x1 is assigned to the cluster of object x2, indicated with symbol *. The process is repeated to all other objects in turn; this concludes one iteration. The order in this case is random, e.g., object six is considered at step three of

iteration one. Only two iterations are needed for clustering the dataset into two clusters (o and Δ).

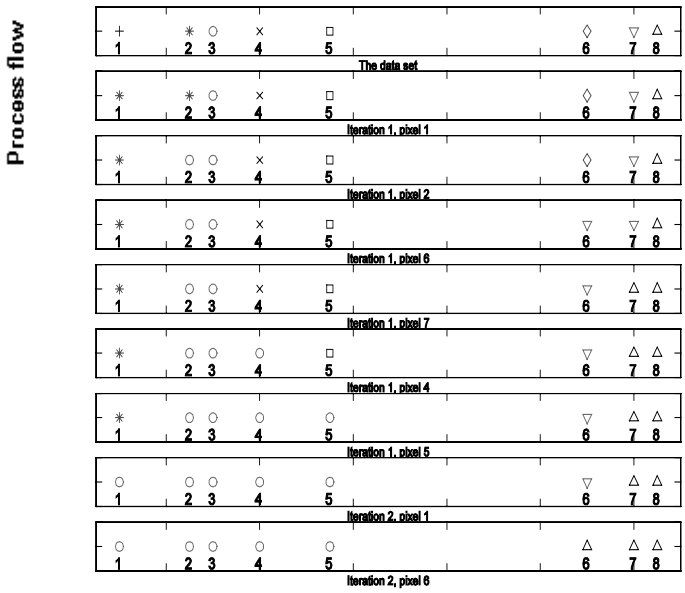


Figure 4. A simple example shows how KNNCLUST. The symbols: *, o, x, □, □, and Δ stand for cluster membership, in which pixels belonging to the same cluster have the same symbol.

In general, the object order, in which the objects are considered, may influence the result of the algorithm. One may order objects by their densities, in which higher density object is taken before the lower. However, density values are changed during iteration and the re-ordering at every step takes a lot of the computation time. In practice, objects may be processed in any convenient order. We have not seen any performance degradation.

3.2.1 Computational complexity

The computational complexity of KNNCLUST depends mainly on the calculation of knn table, the list of k nearest neighbors of all objects, which is very expensive. For example, if we acquire knn query for each object independently, the simplest way is to order the all distances from this object to other objects, which leads to a complexity of O(N log(N)). However, there are many ways to make it more efficient; e.g. integrate information on all queries (see [21] for a summary). The R-tree indexing technique is often utilized, e.g. in DBSCAN.

3.2.2 User-defined parameters

Apart from the choice of the kernel, the algorithm requires only one parameter, the number of neighborhood points, *k*. The smaller *k*, the more detail there is in the clustering and the more clusters can be obtained. In contrast, with a higher value for *k*, the clustering result is 'smoother' and a smaller number of clusters is obtained. In all cases, *k* should be smaller than the size of the smallest cluster, because this cluster will otherwise be missed. It may be difficult to find an optimal value of *k* for a dataset which has clusters of very different size. It is recommended to use several values of *k*, and to pick the one that captures the relevant features of data best. However, as will be shown below, in practice there will be a range of k values that give quite similar results.

3.3 Comparison of KNNCLUST to other clustering methods

KNNCLUST is not an agglomerative hierarchical clustering algorithm [1], where a pair of clusters is merged based on the similarity between pairs of clusters. KNNCLUST is more like partitional clustering [1], where the probability density function (pdf) is used instead of normally used distances, e.g., Euclidean or Mahalanobis distances. In this type of clustering, objects are allowed to be reassigned to other clusters. However, the number of clusters needs to be defined in partitional clustering methods, whereas it is automatically determined by KNNCLUST. Partitional clustering, such as Fuzzy C-means or mixture modeling by Expectation Maximization (EM) is sensitive to the initial choice of cluster centers and noise/outliers present in the data set. This is not the case for KNNCLUST. Moreover, different from mixture model clustering EM, KNNCLUST does not require clusters to have a certain statistical distribution; e.g. the Gaussian distribution is often used in EM. KNNCLUST also differs from ordinary density-based clustering by constructing the class-condition instead of using a density estimation function for detecting separation density valleys between clusters. As a consequence, KNNCLUST is less suited for finding very elongated clusters or clusters with strange shapes, something that is possible with ordinary density-based clustering. On the other hand, it can be used in cases when clusters have very different densities where other density-based methods cannot. Last but not least, KNNCLUST can work well with data in high dimensional feature spaces which is difficult for many clustering algorithms, such as the EM method.

4. Results

In this section, we demonstrate the effectiveness of KNNCLUST on two datasets, a simulated dataset and a remote sensing Compact Airborne Spectrographic Imager (CASI) image.

The 2D simulated dataset in Figure 5 contains four classes having sizes of 600, 400, 200 and 200 objects. To make the simulated dataset more realistic, class one is constructed from two overlapping Gaussians. The other three are generated from three single Gaussian distributions with very different cluster densities; the variances of clusters three and four are ten times smaller than cluster one and two, respectively. The Gaussians are illustrated by ellipses, shown in figure 5. In the plot, classes two and three (in the middle-right of Figure 5) are located in very small areas, and are difficult to distinguish.

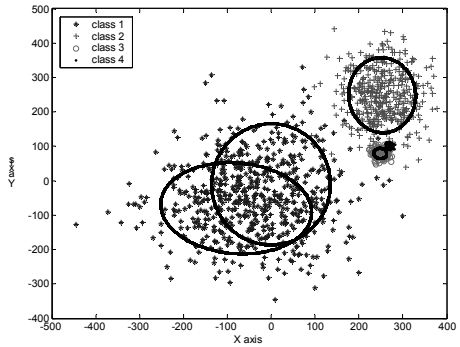


Figure 5. The simulated dataset. Class one is a mixture of two Gaussians and the other three are generated from three single Gaussian distributions with very different in cluster densities.

Using KNNCLUST, the four-cluster results can be obtained using k values in the range [180, .., 220] with total accuracy more than 95 % (by counting the misclassified objects). As an example, the result of KNNCLUST by k = 180 is given in Figure 6.

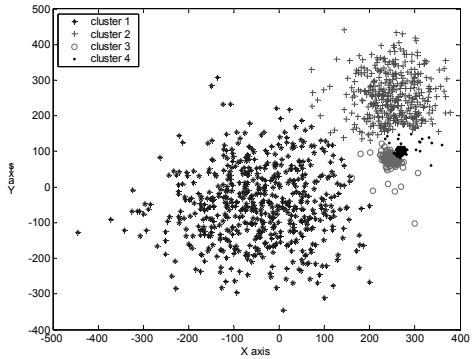
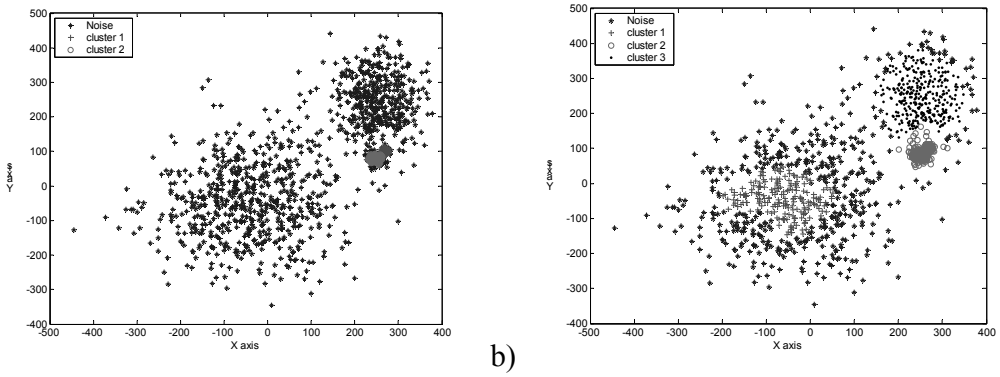


Figure 6. Clustering result by KNNCLUST with k = 180; the total accuracy is 95.9 %.

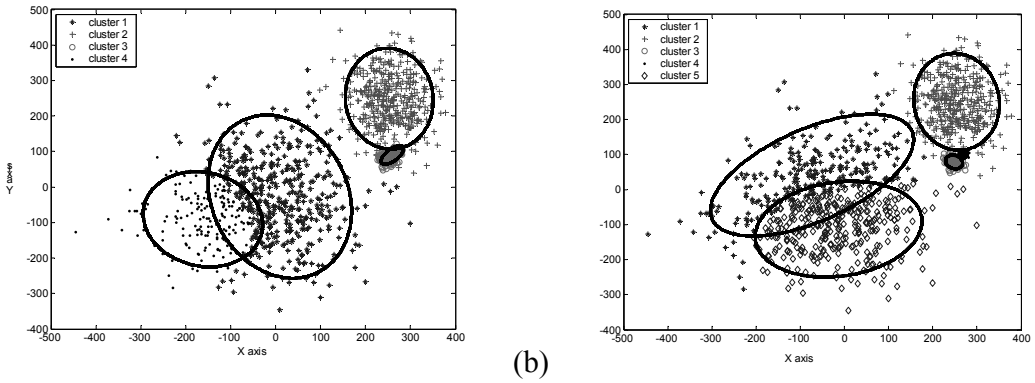
The often-used density-based clustering, DBSCAN [7], is applied to the dataset as well. The clustering result (in Figure 7) is very poor, as expected since clusters have very different densities. The best results of DBSCAN on two situations are discussed hereafter. In order to recognize classes three and four, a very high density threshold with min_points = 10 and $\varepsilon = 20$ is set, leading to objects of class one and two to be classified as noise (Figure 7a). In the opposite situation, using a low dense threshold with min_points =20 and $\varepsilon = 950$, classes three and four are merged (Figure 7b).



a) b)

Figure 7. DBSCAN (a) min_points = 10, $\varepsilon = 20$; (b) min_points =20, $\varepsilon = 950$

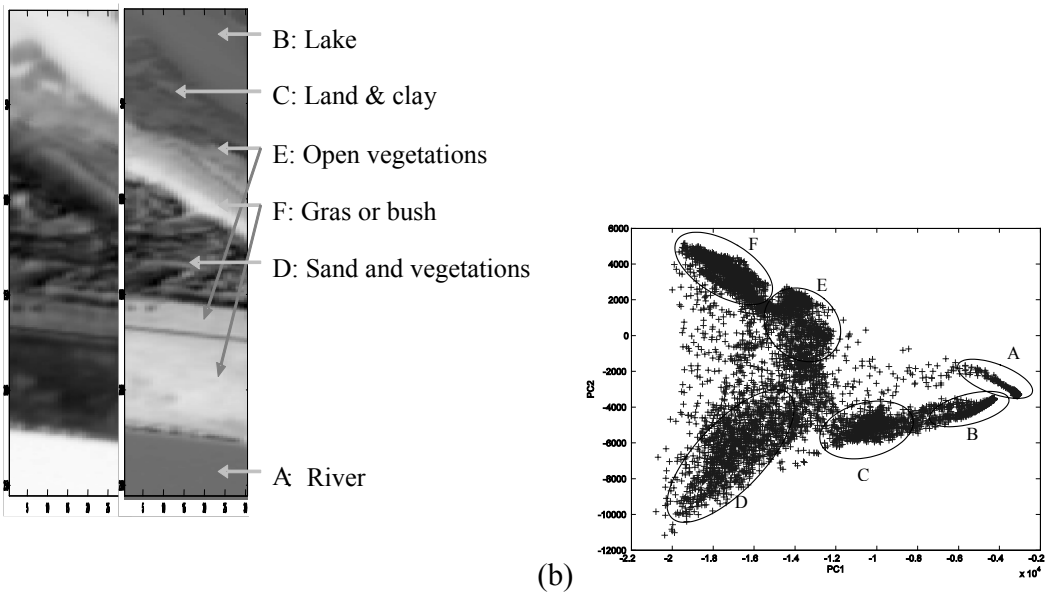
We also compared KNNCLUST with the state-of-the-art mixture model clustering by EM on this dataset. The EM algorithm is very sensitive to initialization [22][23]; a random initialization strategy is normally used. We performed EM to four clusters 100 times and the best clustering result in terms of the maximal likelihood criterion is shown in Figure 8a. Gaussian mixture model clustering assumes clusters to have normal distribution. Because of the mixture of two Gaussians in class one, EM needs two Gaussians to describe the class and the class three and four are merged together. EM works better when working with five clusters, the class three and four can be recognized. However, cluster one still divided to two parts (Figure 8b). Together with the difficulty of the initialization of and the identifying the number of clusters, KNNCLUST works better than EM for this dataset.



(a) (b)

Figure 8. The best of 100 runs of EM to (a) 4 clusters; (b) 5 clusters.

The second experiment is done on a multispectral remote sensing satellite image recorded by a CASI scanner from the Natural Environment Research Council (NERC). The image was taken at 1536 m over an area in the Klompenwaard, the Netherlands, during August 2001. The data set for this study contains 10 bands from 437 nm to 890 nm, with bandwidths of 10 nm, except for band 9 with 8 nm. The study area has size of 30 x 255 pixels with 3 m resolution, covering 68850 m^2. Principal Components Analysis (PCA) is used for reducing the complexity and visualization of the results. The original multispectral data were mean zero and unit variance and compressed via a PCA to the first four principal components, which account for more than 99.8 % of the spectral variance. KNNCLUST was applied on both the original 10-bands dataset and the four-component compressed dataset. The result shows no difference between the two cases. For convenience, the results shown in this paper are shown in PCA space.



(a) (b)

Figure 9. (a) The gray-scale images of the first two principal components (PC1 on the left, PC2 on the right), and the six main object classes that have been identified in the area. (b) The score plot of PC1 and PC2.

Figure 9a shows the gray-scale images of the first and second principal components, explaining 71 % and 27 % of the variance in the data, respectively. Six main object patterns have been estimated for the area from the work of Van den Berg [24].

The clusters are different in density, as can be seen clearly in Figure 9b; e.g., the clusters of the river (A) and the lake (B) are very dense, with a long narrow shape containing approximately 1000 points, compared to the large cluster corresponding to sand and vegetation (D) of 1440 points.

First, we apply an often-used density-based clustering, DBSCAN [7]. DBSCAN clustering is a spanning process, grouping points connected by high density cells and dividing points separated by low density cells. The threshold is a user-settable parameter, □. The second parameter that should be set is min-points, the minimum number of objects in the neighbourhood. The number of clusters is found automatically by DBSCAN. For this data set, many values for both user parameters have been used but none of them gave good results. Some examples are shown in Figure 10 (a-d). This is caused by the absence of a global threshold of the density for the whole data set. If the density parameter is adjusted to identify low density regions such as the sand and the vegetation cluster (D), then it is too high to distinguish between clusters B and C, as well as between clusters E and F in Figure 10c, and d. In other settings, cluster D could not be recognized due to its low density (Figure 10a and b).

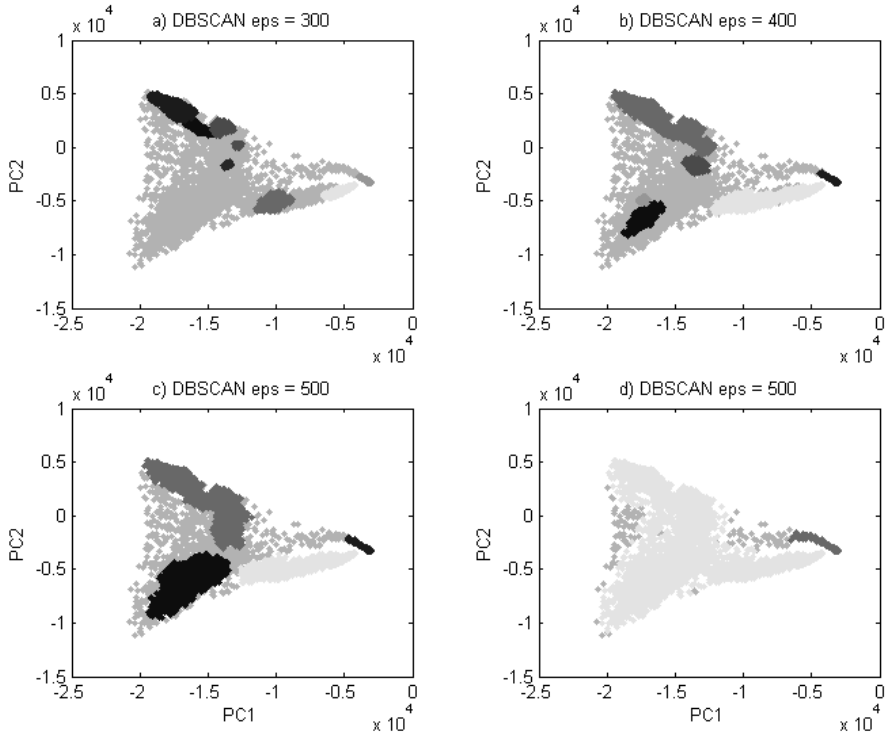


Figure 10. Score-plots of two first PCs by DBSCAN with parameter min-points is 25 (a) 8 clusters found by ε = 300, (b) 6 clusters found by ε = 400 and, c) 4 clusters found by ε = 500 and (d) 2 clusters found by ε = 900;

KNNCLUST was applied using the following values of k: [450, 500, 550, 600, 650]. In all cases, six clusters are found. The score-plots of two first PCs, showing the clusters obtained with k=550, is given in Figure 11c. The cluster sizes range from 950 to 1800 points. Seven and five clusters are found when values of k are 300 and 700, respectively. The method is also compared with K-means, and EM and the best results after 100 runs by randomly initialization are shown in Figure 11a, and b, respectively. In this case, the image result of the KNNCLUST (Figure 11c) and EM are comparable and look much smoother

than the one obtained by K-means, mainly because of the vegetation area (D). K-means incorrectly joins the lake (B) and the river (A), and divides the vegetation area D into two clusters.

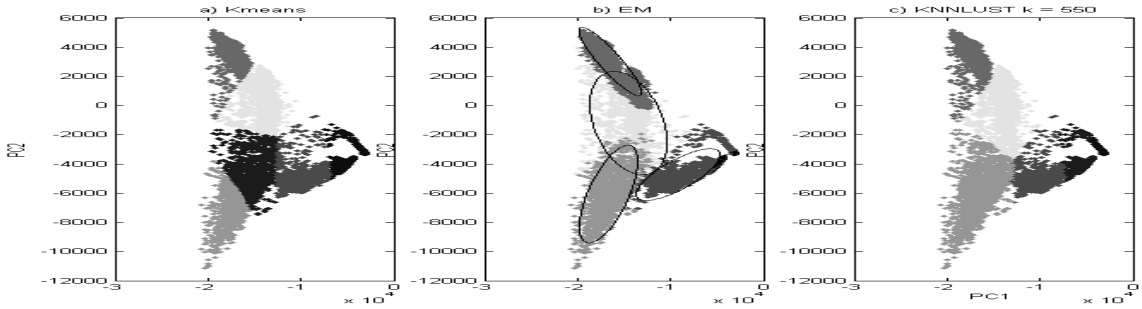


Figure 11. Score-plots of two first PCs and result images of six clusters obtained by (a) K-means (the best of 100 runs); (b) EM (the best of 100 runs) and (c) KNNCLUST with k=550.

The stability and the compactness of the clustering result also can be studied by using an index which measures the ratio of within-cluster variation and between-cluster variation [25]. A lower value indicates a higher compactness. This index is not designed for a data set with clusters of different shapes. Nevertheless, it might provide an idea about the stability and the compactness of the clustering results.

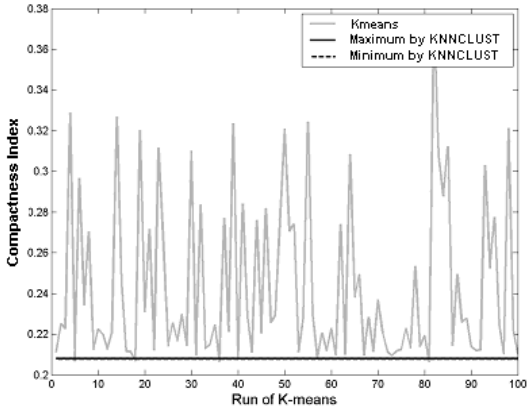


Figure 12. Compactness index of K-means in 100 runs compared to the index of the KNNCLUST result.

Figure 12 shows compactness index values of 100 replicated runs for K-means. It shows that K-means is not stable with a minimum value of the compactness index of 0.2063 and a maximum value of 0.3756. Also in the figure are the smallest and largest compactness values for KNNCLUST using all five values of k leading to six clusters. The smallest value for the index for KNNCLUST is 0.2077 when k = 650, and the largest value is 0.2081 when k = 450.

They are comparable to the best case obtained by K-means. The small variance of the compactness index indicates that KNNCLUST is not very sensitive to the values of k in the selected range.

5. Summary

Many clustering algorithms for multivariate data, such as, EM or most density-based methods, suffer from the problem of clusters in a high dimensional feature space with

different densities. This is not the case for our new proposed algorithm, KNNCLUST, making use of a knn-kernel density estimator using the triangular kernel. For a given kernel function, KNNCLUST has only one parameter, *k*, the number of neighbors. In most cases, it is not difficult to find a range of k for which clustering results are stable. The number of clusters is automatically determined by the algorithm upon convergence. The computational complexity for the algorithm is quite high, mainly caused by the calculation of the knn distance matrix. However, indexing techniques [21] could be used to improve the situation for a larger data set. KNNCLUST is less suited for finding very elongated clusters or clusters with strange shapes, something that is possible with ordinary density-based clustering. However, KNNCLUST can detect more "nature" clusters that are required to follow any type of statistical distributions like in mixture model clustering. In conclusion, it is a very good tool to cluster moderately-sized multivariate data set where the clusters are very different in densities.

6. Acknowledgements

We thank Gertjan Geerling, Department of Environmental Studies, for sharing the data and stimulating discussions.

References

[1] T.N. Tran, R. Wehrens and L.M.C. Buydens, "Clustering multispectral images: a tutorial", Chemom. Intell. Lab. Syst., in press.

[2] A. Smolinski, B. Walczak and J. W. Einax, "Hierarchical clustering extended with visual complements of environmental data set", Chemom. Intell. Lab. Syst., vol. 64, pp. 45-54, 2002.

[3] J. Liang and S. Kachalo, "Computational analysis of microarray gene expression profiles: clustering, classification, and beyond " , Chemom. Intell. Lab. Syst, vol. 62 , pp. 199-216, 2002.

[4] I. Bondarenko, H. Van Malderen, B. Treiger, P. Van Espen and R. Van Grieken, "Hierarchical cluster analysis with stopping rules built on Akaike's information criterion for aerosol particle classification based on electron probe X-ray microanalysis", Chemom. Intell. Lab. Syst, vol. 22, pp. 87-95, 1994.

[5] P. Teppola, S.-P. Mujunen and P. Minkkinen, "Adaptive Fuzzy C-Means clustering in process monitoring ", Chemom. Intell. Lab. Syst., vol. 45, pp. 23-38, 1999.

[6] D. Coomans and D. L. Massart, "Potential methods in pattern recognition Part2. CLUPOT – an unsupervised pattern recognition technique", Analytica Chimica Acta, vol. 133, pp. 225-239, 1981.

[7] M. Ester., H.-P. Kriegel, J. Sander and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". *in Proc. Knowledge Discovery and Data Mining*, 1996, pp. 226-231.

[8] A. Hinneburg and D. A. Keim, "An Efficient Approach to Clustering in Large Multimedia Databases with Noise," *in Proc. Knowledge Discovery and Data Mining*, 1998, pp. 58-65.

[9] K. Fukunaga, L.D. Hostetler. "The estimation of the gradient of a density function, with applications in pattern recognition", IEEE Trans. Inform. Theory, vol. 21, pp. 32–40, 1975.

[10] Y. Cheng , "Mean shift, mode seeking, and clustering", IEEE Trans. Pattern Anal. Machine Intell., vol. 17, no. 8, pp. 790-799, Aug. 1995.

[11] D. Comaniciu and P. Meer, "Distribution Free Decomposition of Multivariate Data", *Pattern Analysis & Applications*, vol. 2, pp. 22–30, 1999.

[12] L. Ertoz, M. Steinbach and V. Kumar, "A new shared nearest neighbor clustering algorithm and its applications", Proc. Workshop on Clustering High Dimensional Data and its Applications, Arlington, VA, USA, 2002, pp. 105-115.

[13] R. A. Jarvis and E. A. Patrick, “Clustering Using a Similarity Measure Based on Shared Nearest Neighbors,” IEEE Transactions on Computers, Vol. C-22, No. 11, November, 1973. [14] Z. Su, Q. Yang, H. Zhang, X. Xu and Y.-H. Hu, "Correlation-based Web-Document Clustering for Adaptive Web-Interface", *Knowledge and Information Systems*, vol. 4, pp. 151-167, 2002.

[15] T.N. Tran, “KNNCLUST in Matlab”, http://www.cac.science.ru.nl/people/tnthanh/.

[16] A. Webb, Statistical Pattern Recognition. Wiley, Malvern, UK, 2002.

[17] D. O. Loftsgaarden and C. P. Quesenberry, “A nonparametric estimate of a multivariate density function”, *Ann. Math. Statist.*, vol. 36, pp. 1049-1051, 1965.

[18] G. R. Terrell; D. W. Scott, “Variable kernel density estimation”, *The Annals of Statistic,* vol. 20, pp. 1236-1265, 1992.

[19] B.W. Silverman, Density estimation for statistics and data analysis. Chapman & Hall, 1986, pp. 21-74.

[20] D. J. Hand, *Discrimination and Classification*. Wiley, New York, 1981 pp. 28-29.

[21] B.V. Dasarathy, *Nearest Neighbour Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, Los Alamitos, CA, pp. 1-30, 1991.

[22] W. Seidel, K. Mosler, and M. Alker, “A cautionary note on likelihood ratio tests in mixture models”, Ann. Inst. Statist. Math., vol. 52, No. 3, pp. 481-487, 2000.

[23] G. McLachlan and D. Peel, “Finite Mixture Models”, Willey series in probability and statistic, Canada, 2000.

[24] G. J. v/d Berg, *Classificatie CASI-beeld: Een vegetatiekaart van de Klompenwaard*. Advies en Onderzoek Remote Sensing en Fotogrammetrie (GAR), 2001.

[25] R. G. Brereton, Multivariate pattern recognition in chemometrics, illustrated by case studies. Elsevier, 1992, pp. 179-204.

SPAREF: A CLUSTERING ALGORITHM FOR MULTI-SPECTRAL IMAGES

Abstract

Multi-spectral images such as multi-spectral chemical images or multi-spectral satellite images provide detailed data with information in both the spatial and spectral domains. Many segmentation methods for multi-spectral 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, SpaRef, a new clustering algorithm for multi-spectral 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 CASI image over an area in the Klompenwaard, the Netherlands. The clusters obtained show improved results. Applying Sparef to multi-spectral chemical images would be a straight-forward step.

Keywords: clustering algorithm; multi-spectral image segmentation; spatial information.

T.N. Tran, R. Wehrens, L. M.C. Buydens, *Anal. Chim. Acta Journal, 490, 303-312, 2003.*

1. Introduction

Clustering is the organization 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 multi-spectral satellite images, organizing 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 multi-spectral 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 works well with small data sets 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, SpaRef (Spatial Refinement clustering), 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 misassigned points using the information of points in the spatial domain.

Fist, 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 multi-spectral 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 pixel is characterised by D_{im} variables (reflectance values).

We will use the following notations:

- 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 << N
- Minsize is a minimum size of a normal cluster (so the clusters should contain at least 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 ≠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 class 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 :

$$E_f = \sum_K \sum_{i \in C_k} u_{ik} d^2(x_i, c_k) \tag{7}$$

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 (AHC) 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 can not 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 in 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.

$$W = \frac{1}{N}\sum_{K}\sum_{i \in C_k} d(x_i, c_k) \tag{8}$$

$$B = \frac{1}{N}\sum_{K} n_k d(c_k, c) \tag{9}$$

Clustering algorithms minimizing the sum-of-squares criterion (eq. 6) would thus minimize 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 generalized 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 $\overline{R}$ of within-cluster scatter and between-cluster separation:

$$R_k = \underbrace{\max}_{j \neq k}\left\{\frac{W_k + W_j}{B_{kj}}\right\} \tag{10}$$

$$\overline{R} = \frac{1}{K}\sum_{k=1}^{K} R_k \tag{11}$$

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:

$$I = \frac{W}{B} \tag{12}$$

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 ($d \neq c$).

$$D_c = \frac{B_c}{n_c} \tag{13}$$

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.

$$D = \frac{1}{N}\sum_{c=1}^{K} B_c \qquad (14)$$

3. Description of SPAREF

SpaRef is designed to use a combination of K-clustering and agglomerative hierarchical clustering (AHC) 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. 12) 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, 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\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 figure 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 reassigned 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.

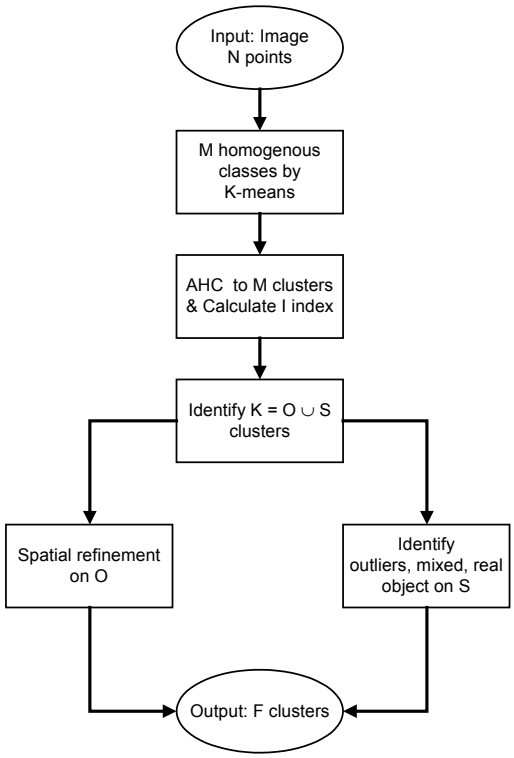


Figure 1. Flow chart of SpaRef method.

SpaRef depends on two main input parameters, M and 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, and stabilize 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 Minsize is thus easily defined, in practice.

The total complexity of SpaRef is equal to $O(MlogM) + O(M^2)$. 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 multi-spectral 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 multi-spectral satellite image recorded by a Compact Airborne Spectrographic Imager (CASI) scanner from the Natural Environment Research Concil (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 nm to 890 nm, with bandwidths of 10 nm, except for band 9 with 8 nm. The area has size of 211 x 301 pixels leading to 63511 pixels with 3 m resolution covering 633 x 903 m^2

(Figure 2). The original multi-spectral data were mean centered 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.

Figure 2. Grayscale-image of the first principal component (PC1) of the image. The white band on the right corresponds with the river; other with areas show small lakes. Dikes and roads are visible (e.g. parallel to the river). Vegetation and woodland are visible as darker shades of gray.

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 figure 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.

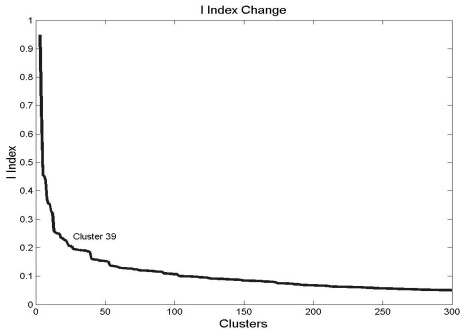
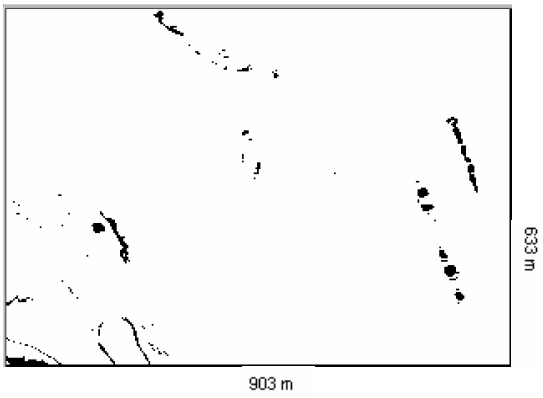


Figure 3. I index changes while applying AHC to 300 homogenous clusters. The optimal number of clusters is identified to be 39 where there is a sharp change.

This data set is expected to contain also noises, mix-pixels, and hence, Minsize is set to 100. Otherwise, Minsize is zero. 14 clusters with sizes smaller than 100 pixels, containing in total 765 points, (1%), have been rejected (Figure 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 62746 points, 99 % of points are subjected to the refinement process as described earlier.

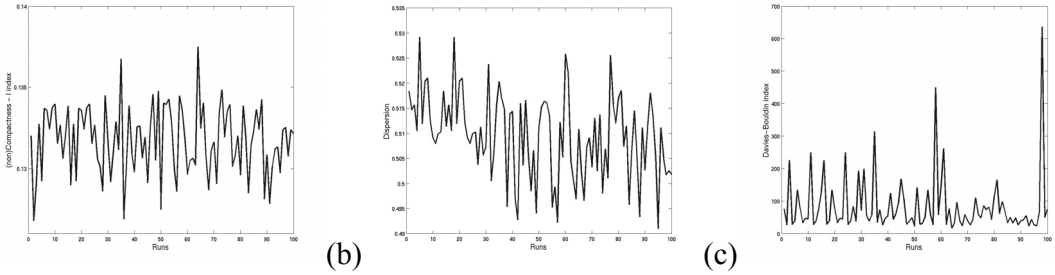


(

Figure 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.).

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 DB indices are illustrated in figure 5. Clearly, for the 100 runs, the variability in all three indices is quite large.



(a) (b) (c)

Figure 5. Indices of K-means for 25 clusters for 100 runs: (a). (non)Compactness I index, (b). Dispersion index (c). Davies-Bouldin index.

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

Methods	I index	Davies–Bouldin	Dispersion index
K-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

5.4 Application of ISODATA

The data set has been also clustered by ISODATA algorithm [Figure 6a] which is implemented in MultiSpec software, a multi-spectral image data analysis system for interactively analyzing Earth observational multi-spectral 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 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 DB 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.

5.6. Results

Four clusters for ISODATA, the first of 100 runs of K-means and SpaRef, and 3 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 Figure 6(a),(b),(c) and (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.

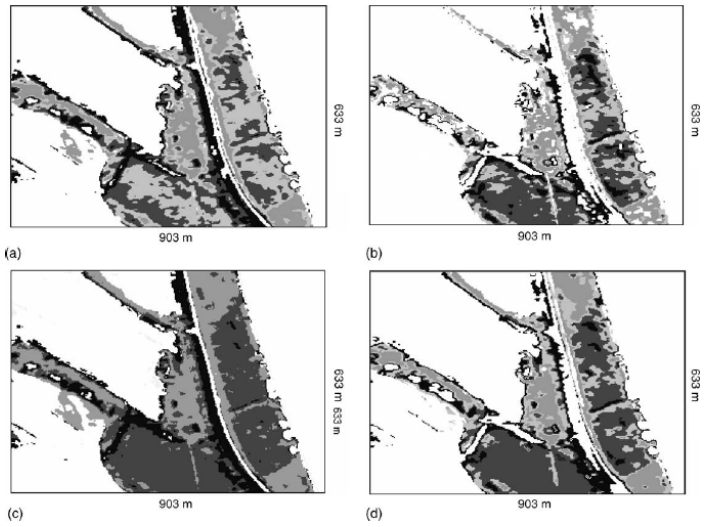


Figure 6. Four clusters of clustering results: (a) by ISODATA; (b) by Kmeans; (c) by Ward; and (d) by SpaRef. The colour 'white' in (a-d) signifies 'others clusters'; there are only four shades of gray in (a).

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 DB indices of different methods. In (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 DB index gives the same scenario as the I index.

In the dispersion index, D, M-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 M-Ward method but, as mentioned, it may be more close to the 'true' value of D index. Indeed, in figure 6c, the clustering result from M-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 figure 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 figure 6a and figure 6b, 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 multi-spectral 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.

7. Acknowledgements

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

References

[1] A.K. Jain, M.N. Murty, and P.J. Flynn, ACM Computing Surveys, 31:3 (September 1999) 264-323.

[2] T. Duda, M. Canty, Int.J.Remote Sensing, 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 Record, 31:2 (June 2002), 40-45

[5] R. G. Brereton, Multivariate pattern recognition in chemometrics, illustrated by case studies, Elsevier 1992.

[6] A. El-Hamdouchi, P. Willett, The Comp. J., 32:3 1989.

[7] M. Amadasun, R. A. King, Pattern Recognition, 21:3 (1988) 261-268.

[8] M. R. Anderberg, Cluster analysis for applications, Academic Press, New York, 1973.

[9] P.H.A. Sneath, and 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. of the American Statistical Association,58 (1963) 236-244.
[12] J.C. Duun, J. Cybern, 4 (1974) 95-104.
[13] D. L. Davies and D. W. Bouldin, IEEE Trans. on Pattern Anal. and Mach. Int,. 1:2 (April 1979) 224-227.
[14] R. Kothari, D. Pitts, Pattern Recognition Letters, 20 (1999) 405-416.
[15] J. C. Bezek, N. R. Pal, IEEE Trans. on Sys, Man, and Cyber. – part B, 28:3 (June 1998) 301-315.
[16] J.A. Richards, X. Jia, Remote Sensing Digital Image Analysis, Springer, 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.

INITIALIZATION OF MARKOV RANDOM FIELD CLUSTERING OF LARGE REMOTE SENSING IMAGES

Abstract

Markov Random Field clustering, utilizing both spectral and spatial inter-pixel dependency information, often improves classification accuracy for remote sensing images, such as multi-channel polarimetric Synthetic Aperture Radar (SAR) images. However, it is heavily sensitive to initial conditions such as the choice of the number of clusters and their parameters. In this paper, an initialization scheme for MRF clustering approaches is suggested for remote sensing images. The proposed method derives suitable initial cluster parameters from a set of homogeneous regions, and estimates the number of clusters using the Pseudolikelihood Information Criterion (PLIC). The method works best for an image consisting of many large homogeneous regions, such as agricultural crops areas. It is illustrated using a well-known polarimetric SAR image of Flevoland in the Netherlands. The experiment shows a superior performance compared to several other methods, such as fuzzy C-means and Iterated Conditional Modes (ICM) clustering.

Keywords: Image clustering; Spatial information; Parameter estimation; ICM;

T.N. Tran, R. Wehrens, D.H. Hoekman and L.M.C. Buydens, *IEEE Trans. on Geosci. Remote Sensing,* vol. 43, iss. 8, 1912-1919, 2005.

1. Introduction

Clustering is an important tool in multi-spectral/channel image analysis. Most clustering methods do not take into account spatial information of the image, the inter-pixel dependency in the image surface. Markov Random Field (MRF) clustering, first discussed by Besag [1][2] and later improved by Qian and Titterington [3], provides a way to integrate spatial information with a model-based clustering approach [4][5]. In many cases, this reduces a possible overlap problem of clusters and the effect of noise on the clustering result [6]. MRF clustering has also been applied to remote sensing [7][8][9][10]. In MRF clustering approaches, of which the iterated conditional modes (ICM) clustering is an example, the class probability of a pixel is locally dependent on its spatial neighbor clusters. In operation, just as the ordinary model-based clustering, the method assumes a mixture of all components (clusters). Starting from an initial guess model, an iterative method is used to fit the model to the dataset. The most common way is using maximum likelihood via an expectation-maximization (EM) algorithm. However, different from the ordinary model-based clustering, the only classes considered for a pixel are classes that are present among the neighboring pixels [3][4]. We refer the reader to McLachlan and Peel [4] for an extensive review of MRF mixture models.

Since the convergence of MRF clustering methods is local, the accuracy is much more dependent on the initial guess of cluster parameters than the classical model-based clustering algorithm. They typically work well in supervised mode, where the number of clusters and their associated parameters are known or can be estimated [10]. MRF clustering methods then tend to converge rapidly [1]. If the estimation of the initial parameters fails, classification results can be very poor [11], and a locally optimal solution is often obtained instead of a global solution. Thus, the initial parameters should be quite close to the true parameters. The initialization scheme is often simply random, or sometimes it is obtained from other clustering techniques, such as k-means [11] or fuzzy c-means.

As an alternative, an agglomerative hierarchical clustering (AHC) framework can also be used as an initialization scheme, either in the form of single- and complete-linkage methods, or in a model-based form [12]. The method provides a dendrogram, representing nested clusters. Initial parameters for model-based clustering, as well as MRF clustering in this case, can be easily extracted for different cluster models. However, ordinary AHC initialization starts with singleton clusters [6] which makes it impractical for large data sets.

In this paper, a new AHC initialization framework to MRF clustering, suitable for large remote sensing images, e.g. Synthetic Aperture Radar (SAR) images, is proposed. Instead of starting with singleton clusters, a limited number of homogeneous regions, obtained from a simple segmentation method (using a so-called “multi-level homogeneity test”) is used for building up the dendrogram. In general, many merging criteria, or distances between two clusters, can be used in AHC [11]. Here, a deterministic Bhattacharyya distance and a probabilistic likelihood are used.

In this study, an example of MRF clustering with the new initialization framework is evaluated on a polarimetric SAR image of an area in Flevoland in the Netherlands.

The paper is organized as follows. In Section II, we introduce the basic elements of model-based and MRF clustering. The proposed clustering strategy, using the hierarchical agglomeration initialization scheme, is described in Section III. Problems of determining the number of clusters and dealing with outliers are also discussed. Section IV shows the

application to the polarimetric SAR image of Flevoland. Our conclusions and discussions are given in Section V.

2. Basic Elements in Mixture Models and Markov Random Field Clustering

A. Mixture models

An image of size n in d dimensional feature space contains a set of pixels $X=\left(x_1^T,\ldots,x_n^T\right)^T$, where x_i is a vector of pixel values in the spectral domain. In model-based clustering [4][5], each cluster c is described by a multivariate distribution f with parameters θ_c. Most commonly, f is the multivariate Gaussian distribution, and θ_c contains mean μ_c and covariance Σ_c. The total data set is described by a linear combination of individual clusters and their corresponding mixture proportions π_c. The probability density function $f(x_i;\Psi)$ of the pixel x_i under a g-component (cluster) mixture is then given by:

$$f(x_i;\Psi)=\sum_{c=1}^{g}\pi_c f_c(x_i;\theta_c) \tag{1}$$

where g is the total number of clusters, and Ψ contains all cluster parameters and mixture proportions.

The probabilistic likelihood function $L(\Psi)$ is given by the following expression:

$$L(\Psi)=\prod_{i=1}^{N} f(x_i;\Psi) \tag{2}$$

Clustering can be seen as an incomplete-data problem, in which u_{ic} is defined as the conditional probability of object x_i belonging to cluster c. The complete-data X is now therefore declared to be [4]

$$X_c=\left(X^T,u^T\right)^T \tag{3}$$

where matrix u contains the values u_{ic}.

The complete-data log-likelihood function is then derived, $\log L_c(\Psi)$, (refer to [4] for a complete derivation):

$$\log L_c(\Psi)=\sum_{c=1}^{g}\sum_{i}^{N} u_{ic}\log(\pi_c f(x_i;\theta_c)) \tag{4}$$

The aim of model-based clustering is to obtain a configuration Ψ, so that it maximizes the log-likelihood function, log $L_c(\psi)$. It is usually performed by the EM (Expectation-Maximization) algorithm [13]. At each iteration, k, EM consists of two sub-steps, called the M-step (Maximization step), maximizing π_c and θ_c, and the E-step (conditional Expectation step), estimating u_{ic} by the normalized post probability, given by:

$$\pi_c^k=\frac{\sum_{i=1}^{N} u_{ic}^{k-1}}{N} \tag{5}$$

$$\mu_c^k = \frac{\sum_{i=1}^{N} u_{ic}^{k-1} x_i}{\sum_{i=1}^{N} u_{ic}^{k-1}} \text{ ; and } \quad \Sigma_{ic}^k = \sum_{c=1}^{g}\sum_{i=1}^{N} u_{ic}^k \left(x_i - \mu_c^k\right)\left(x_i - \mu_c^k\right)^T \tag{6}$$

$$u_{ic}^k = P^k(x_i \mid c) = \frac{\pi_c^k f_c\left(x_i; \theta_c^k\right)}{\sum_{d=1}^{g} \pi_d^k f_d\left(x_i; \theta_d^k\right)} \tag{7}$$

EM starts with an initial guess Ψ^o and iterates until convergence, or until the number of iterations exceeds a certain threshold. An advantage of model-based clustering methods is that the classification results are in a "soft" form, a conditional probability, instead of a "hard" form, e.g. as in K-means or ISODATA methods. The "soft" form of clustering result is more flexible to model remote sensing images, where there are mixtures of ground cover types within a resolution cell, noise due to limited sensor sensibility or, in case of radar, statistical variation because of speckle. Outliers or noise pixels normally show a low membership for all clusters. Moreover, the method is computationally efficient. However, just like many other clustering methods using only spectral information, the method is influenced by the problem of overlapping clusters [6]. This problem can be reduced by taking into account spatial information.

B. Markov Random Field and model-based clustering

Model-based clustering can be combined with the Markov Random Field (MRF) to take into account the spatial relation between pixels. In literature, the MRF model on model-based clustering first has been applied for the restoration of 'dirty' images [1] and referred to a smoothing technique which gives more weight to the fuzzy class memberships of spatial neighbor clusters. It is assumed that the class probability of a pixel is only dependent on class memberships of its (spatial) neighbor clusters, so that it reduces the possible influence of noise and overlapping clusters [6]. Practical examples in remote sensing applications show improvement of the separation of various ground cover classes [7][8][9][10].

More precisely, the *w-th* order neighborhood system for a particular pixel *i*, called ∂_i, is defined as a set of neighbor pixels belonging to a rectangular window of size *w*, centered at the pixel *i*. The conditional probability of point xi of belonging to cluster c under the neighboring system ∂_i is estimated by [1]:

$$P(c_i = c) = \frac{1}{Z}\exp\left[\beta \sum_{j \in \partial_i} u_{jc}\right] \tag{8}$$

where Z is a normalization constant and β is a spatial smoothness parameter. A higher (positive) β corresponds to higher spatial dependency of neighbor points. The EM algorithm is then adapted, leading to the log likelihood criterion:

$$\log L_{MRF}(\Psi) = \sum_{c=1}^{g}\sum_{i}^{N} u_{ic} \log\left(\pi_{ic} f\left(x_i; \theta_c\right)\right) \tag{9}$$

The mixture proportions π_c is now replaced by the transition probability π_{ic} [4]:

$$\pi_{ic} = \exp\left(\beta \sum_{j\in\partial_i} u_{jc}\right) / \sum_{h=1}^{g} \exp\left(\beta \sum_{j\in\partial_i} u_{jh}\right) \quad (10)$$

Not only does the algorithm tend to converge faster, since EM tends to converge to a local optimum, the clustering accuracy is heavily dependent on the initial guess Ψo, and the choice of number of clusters [11][1]. Therefore, obtaining a good Ψo is a key element of MRF clustering.

3. The proposed method

We propose a new initialization framework, based on agglomerative hierarchical clustering (AHC), which is suitable for remote sensing images. The ordinary AHC usually starts from *N* singleton clusters. Iteratively, the similarities between all cluster pairs, *i* and *j,* are calculated and two 'closest' clusters are merged. The algorithm ends when there is only one cluster. Variants differ mainly according to the criterion for optimality, the cluster similarities. Single-linkage, Complete-linkage, and Average-linkage are classical agglomerative methods with the merging criterion to be nearest, farthest, and average neighbor. In hierarchical model-based clustering [12], the probabilistic likelihood similarity is used, and a maximum-likelihood pair is merged at each stage according to a specific model.

The AHC algorithm yields a dendrogram, representing nested clusters and similarity levels where clusters are joined. In order to obtain initial parameters for a particular number of clusters model, the dendrogram is cut at the corresponding level. The equivalent parameters are extracted and they can be used for initialization of the MRF model-based clustering [12]. However, this method is suitable for only very small data set because the method demands very high computation time and computational resources proportional to the square of the number of pixels. In order to reduce the computation time, one solution is to apply the method on a small representative subset of pixels of image. Usually, random samples are taken [14]. Iterative procedures may also be used [15][16].

An alternative method is segmentation of the image into a number of homogeneous regions. The AHC clustering is then applied to these homogeneous regions rather than to the whole image. The minimum spanning tree and k-means, for example, are used to create such regions in [15] and in [17], respectively. In this study, a simple segmentation method, the so-called "multi-level homogeneity test", is used to obtain homogeneous regions.

The full proposed clustering procedure is summarized in the flowchart below:

The algorithm:

Step 1 (Representative regions): Obtain an over-segmented image by applying the multi-scale homogeneity test.

Step 2 (Parameter estimation): Apply agglomerative hierarchical clustering on the over-segmented image to obtain the initial parameters for M predefined models.

Step 3 (Actual clustering): Apply MRF clustering for each model, using the initial parameters obtained in the previous step. The final solution will be selected from the M models using the Pseudolikelihood Information Criterion (PLIC).

The image is first divided into a number of regions in a multispectral image. A region r is defined to be a group of pixels forming a continuous region in spatial domain, e.g. a rectangle or an ellipse. Given a region r and a set of sub-regions r_i, the region r is said to be totally homogeneous at significance level α, for instance 0.05, if for all pairs of sub-regions $<r_i, r_j>$ the test of complete homogeneity is not rejected at the significance level α. The test of complete homogeneity is defined below:

Given two groups of pixels, A and B, with numbers of pixels, mean vectors and covariance matrices, $\{n_A, \mu_A \hat{\Sigma}_A\}$ and $\{n_B, \mu_B, \hat{\Sigma}_B\}$, respectively. $\Sigma_{<AB>}$ is covariance matrix of the union of A and B. The test of complete homogeneity of two groups under the hypothesis Hc; $\mu_A = \mu_B$ and $\hat{\Sigma}_A = \hat{\Sigma}_B$ is the likelihood ratio test $\lambda = L_c / L$, where Lc and L are the maximized likelihoods under the hypothesis Hc and the unconstrained maximum likelihoods, respectively. The statistic

$$-2\log\lambda = (n_A + n_B)\log|\Sigma_{<AB>}| - (n_A \log|\Sigma_A| + n_B \log|\Sigma_B|) \tag{11}$$

has an asymptotic chi-squared distribution with $\frac{d(d+3)}{2}$ degrees of freedom [18], where d is the number of input bands of the image data.

Hence, two groups of pixels, A and B, are said to be "completely homogeneous" at significance level α if and only if $-2\log\lambda$ is not significantly larger than the critical value provided by the chi-squared distribution.

The homogeneity test for the region can also be recursively applied to all sub-regions r_i, as in Figure 1. It is then called the multi-level homogeneity test. In this study, the two-level homogeneity test is used, in which the test is repeated once for all sub-regions. It is obvious that less homogenous regions are obtained for higher level tests because these are stronger than the lower level test.

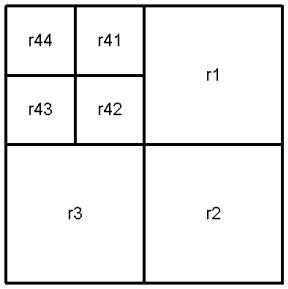


Figure 1. Multi-scale homogeneity test for "spatial region". Region R consists of r_1, r_2, .., r_4. Again, each sub-region r_i consists of r_{i1}, …, r_{i4}.

In principle, the size of regions will determine the total number of obtained regions. The choice is a trade-off and a trial-and-error strategy is normally applied. The test is less biased with a larger size of the region, but it should not be too large because very large regions would contain more than one component and the homogeneity test may fail. Moreover, small homogeneous regions may not be recognized. On the other hand, the size should not be too small, because this leads to higher bias on the test, even it is performed on the area of the same component. This would result in fewer homogenous regions and less reliable estimates of the parameters. In practice, we found that a good setting of the region size lies around 10 x 10 pixels for a two-level method. The choice, of course, depends also on image resolution. The task is easier for high resolution images.

The agglomerative hierarchical process is used in step 2 to merge homogeneous regions, yielding a dendrogram. From that, the statistical parameters can be extracted for each cluster model. The merging criterion in AHC can be a deterministic, e.g. the Euclidean distance, or a probabilistic likelihood similarity, as used in model-based hierarchical clustering [5][12]. In this study, the Bhattacharyya deterministic distance, which gives the distance between two Gaussian regions *r1* and *r2*, is used [19]:

$$B(r1,\ r2) = \frac{1}{8}(\mu_{r1} - \mu_{r2})^T \left(\frac{\Sigma_{r1} + \Sigma_{r2}}{2}\right)^{-1} (\mu_{r1} - \mu_{r2}) + \frac{1}{2}\ln\left(\frac{\left|\Sigma_{r1} + \Sigma_{r2}\right|}{2\sqrt{\left|\Sigma_{r1}\right|\left|\Sigma_{r2}\right|}}\right) \quad (12)$$

The Bhattacharyya distance consists of two terms, which are dominated by the differences in mean, and covariance, respectively. It is very close to the Bayes error of two clusters [19].

In many cases, outliers are also present in the regions obtained in step 1. By the hierarchical mechanism, they are trapped into isolated singleton clusters. The real number of clusters can be thus defined after these singleton clusters (outliers) are eliminated [17].

At this point, a list of solutions is extracted from the dendrogram for M interesting models. Then, statistical parameters for each cluster model can be calculated. Then, the last step, the actual MRF clustering is performed on the entire image for all selected models.

A. Determining the best model

One of the ways to determine the best model in model-based clustering is by using an approximate Bayes factor [20]. The Bayesian Information Criterion (BIC) is often used for the traditional model-based clustering [5]:

$$BIC = 2\log L(\Psi_k) - d_k \log(n) \quad (13)$$

where d_k is the number of parameters of the model.

The Pseudolikelihood Information Criterion (PLIC) is adapted from BIC for the MRF modeling [21]:

$$PLIC = 2\log L_{MRF}(\Psi_k) - d_k \log(n) \quad (14)$$

where the MRF log-likelihood function, $\log L_{MRF}(\Psi_k)$, is used instead of the ordinary log-likelihood function. PLIC is used in this study. The best model normally corresponds with the highest BIC or PLIC value. For a complex data set, where more Gaussians are needed to fit one class, the most significant increase in BIC or PLIC value is used.

Apart from the best model, suggested by measures like BIC or PLIC, the visualization of the list of *M* clusterings gives additional information for choosing a "good" number of clusters. This is a useful feature of this approach in practice, e.g. in remote sensing as in the example in Section IV.

B. Model Outliers

MRF clustering yields $z_{ic} = P(x_i \mid c)$, the posterior probability of point x_i on the component *c*. A "hard" clustering tends to interpret this by assigning the pixel x_i to a cluster *c* if $P(x_i \mid d)$ for all d. This interpretation is only valid if the pixel belongs to at least one cluster. This is not the case in this study, where the initialization process does not provide a complete list of the clusters, but a group of major clusters. It means that there will be a set of pixels, the so-called set *O*, that are not close to any of those major clusters. Put differently, these pixels are poorly fitted by the current model. In this case, the pixels in set

O can be identified as having "very" low probability densities for all identified clusters, and set *O* can be seen as containing the outliers to the model.

One of the ways to identify set *O* is to compare the Mahalanobis distances of a pixel to all clusters with the Hotelling T^2 distribution. Thus:

$O = \{x_i \mid Mah^2(x_i \mid c) < T_c^2\}$ for each cluster c, $Mah^2(x_i \mid c) = (x_i - \mu_c)^T \Sigma_{c-1} (x_i - \mu_c)$, and $T_c^2 = \frac{m(n_c - 1)(n_c + 1)}{n_c(n_c - m)} F_{\upsilon,m,n_c - m}$ where n_c is the number of pixels in cluster *c, m* is the number of dimensions, υ is the level of significance, $1 - \upsilon$ is the level of confidence, e.g. 95%, and *F* is the F-statistic. [18]

The set *O* may consist of pixels from a cluster, which is small in size or isolated in the spatial domain. One can further work on this set by using any 'spectral-only' clustering method. The suggestion here is to use incremental model-based clustering, described in [16]. The method builds a model taking into account the current model and iteratively adding new clusters to describe set *O*.

C. Computational analysis

The proposed method is fast. The statistical test in step 1 has a complexity of $O(n.w)$, where *w* is the size of the region. In step 2, it is noted that only part of the image is taken into account. Pixels in heterogeneous regions, rejected by the homogeneity test, are skipped and only homogenous regions are considered. The maximum number of operations is $O(s^2)$, where s is the total number of homogeneous regions. Lastly, the complexity of MRF clustering is equivalent to $O(n \log n)$. Hence the total complexity of the system is $O(n.w + s^2 + n.\log(n))$, which is acceptable for a normal size of remote sensing image.

4. Application to SAR Data

We investigate our method by applying it to a well-known SAR image of Flevoland, an agricultural area in The Netherlands, acquired by the NASA/JPL AirSAR system (C-, L- and P-band polarimetric) on 3 July 1991. The polarimetric backscatter behaviour for homogeneous fields can be described by the Wishart distribution or its marginal distributions [22][23][24][25]. The characteristics of the physical scattering mechanisms are employed for classification in [26][27][28]. They may also be exploited in the initialization phase of model-based clustering [29][30]. In fact, they can also be applied to MRF clustering. However, it is outside the scope of this paper to discuss and compare these in detail.

In [31], the full polarimetric information is transformed to a log-normal distribution, and the validity of this is demonstrated for the Flevoland data set which used in this paper. For practical applications, it is important to note that when intensities have a log-normal distribution, this distribution transforms into a normal distribution after logarithmic scaling. Then, the classification can be performed directly on these logarithmically scaled (dB values) intensity images with multivariate Gaussian distribution assumption. Note that for an "individual homogeneous field" the complex Wishart distribution and its marginal distributions are appropriate. However, for classification of a complex scene, featuring between field variations, the class distributions (i.e. the values of all pixels belonging to a certain class) are the ones that are of primary importance. For a collection of fields from

the same class, which typically show slightly different radar backscatter mean values, the signals are shown to conform well to log-normal distributions.

In this study, 18 intensity bands from the C- and L-band of the full polarimetry model (see reference [31]) are used. In that reference, only supervised classification, where the classes statistics are known from the training set, is used. The study area has a size of 400 x 400 pixels and is taken from the original data without any aggregation process. This is a reasonable size for demonstration purposes in this case. Even then, it still requires good initial parameters for clustering in order to have good results for the area. The clustering process takes only few minutes using Matlab on a PC Pentium IV computer.

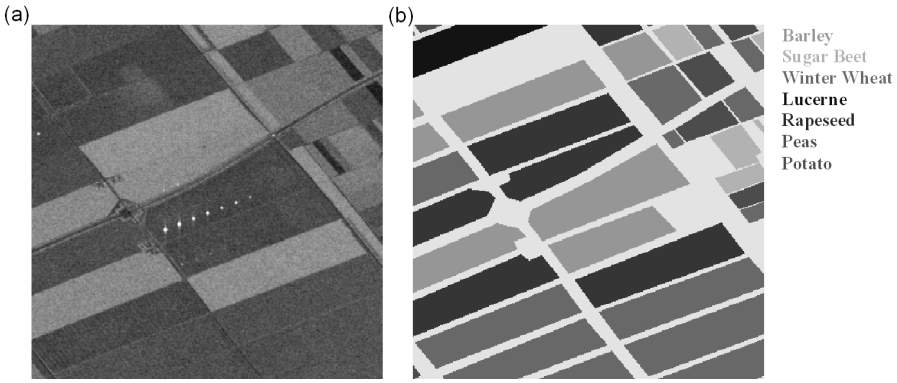


Figure 2. (a) shows the false-color image and (b) the ground-truth information of the site.

Fig. 2a shows the false-color image of the first 3 intensities of the C band. The crop type map which is the ground truth for the clustering is shown in Fig. 2b. The yellow color is a mask where the ground truth is uncertain, or not recorded. Heavily overlapping clusters are shown in Fig. 3 between Barley (Green) and Winter Wheat (Magenta) clusters. Together with sensor speckles, they are the two main problems for this image.

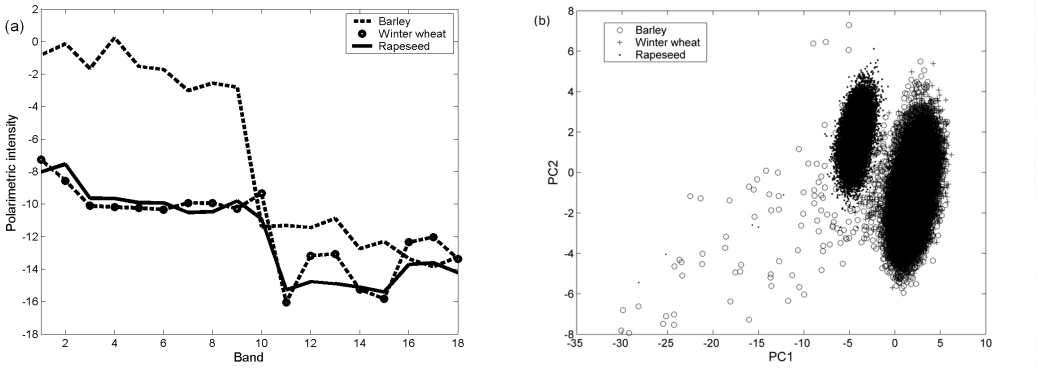


Figure 3. a) Mean spectra of objects in each of the three classes, (b) Score plot of the two first PCs of all pixels in the three classes.

For the analysis, the image is first divided to 3136 square windows (regions) with a size of 15x15 pixels. It is done so that two adjacent regions are overlapping for 50 percent, i.e. the center of one region is on the edge of the other. This produces more regions for the test, increasing the probability of a region corresponding exactly to one crop type, leading to more homogeneous regions. Indeed, after a two-level homogeneity test, the over-segmented image contains 227 homogeneous regions, as shown in Fig. 4.

Figure 4. Homogeneous regions.

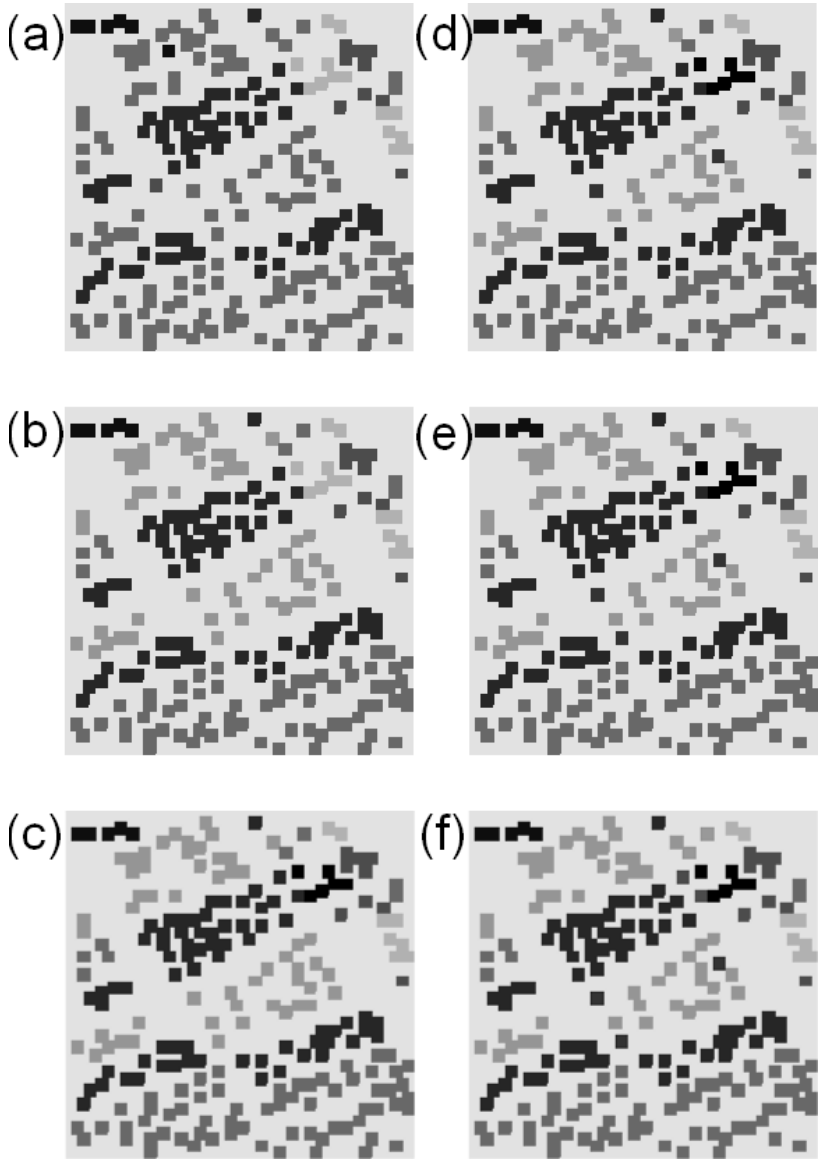


Figure 5. AHC result on homogeneous regions using Bhattacharyya distance to 5-, …, 10-cluster models.

Subsequently, homogeneous regions are combined with AHC using the Bhattacharyya distance. Statistical parameters are extracted for seven models, corresponding to [4, ..,10] clusters. As an illustration, six of these models are shown in Fig. 5. Then, at the final step,

MRF clustering is performed for each cluster-model with β=1 and a 5x5 neighboring window system. The clustering results of four selected models are shown in Fig. 6.

In this case, we know that the correct number of classes equals seven. The PLIC values for the seven models, considered in step 2, are plotted in Fig. 7. In this complex data set, the 6-cluster model shows the largest increase in PLIC value. This is to be expected since there is a large overlap between Barley and Winter Wheat clusters (Fig. 3).

The seven-cluster model obtains more than 97 % accuracy on the area for which reference information is available (the non-yellow area in Fig 2b). The accuracies of the separate clusters are 96, 95, 100, 98, 98.5, 100 and 91%, respectively. Since the clustering result is unlabelled, the clustering accuracy is calculated from the most overlapping cluster-class combination.

c)

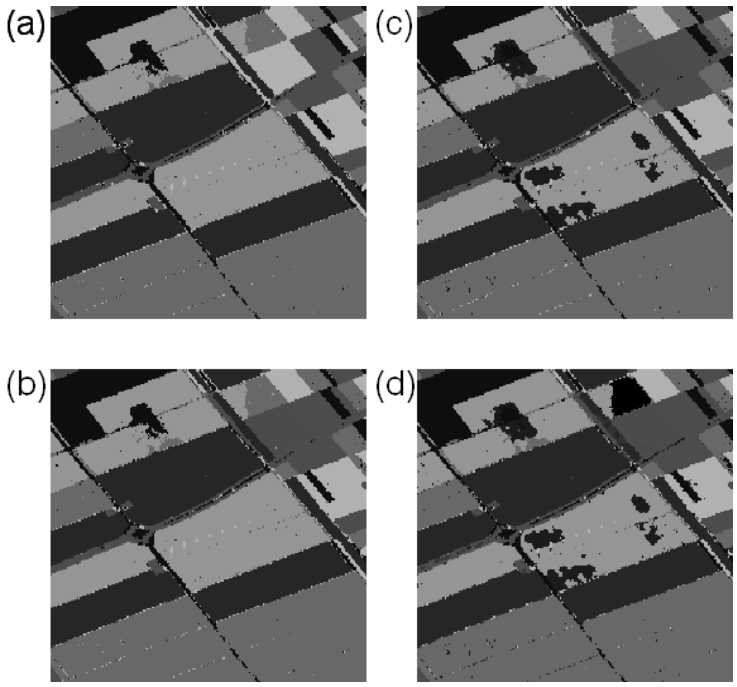


Figure 6. MRF clustering results for (a) 6-, (b) 7-, (c) 8-, and (d) 10-cluster models.

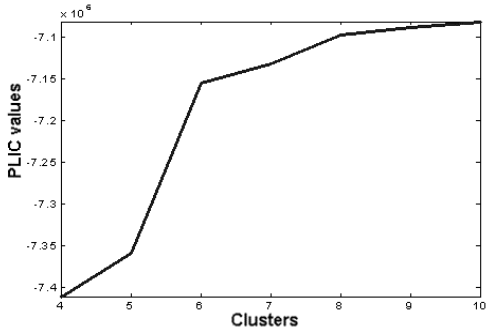


Figure 7. Plot of PLIC values for 4-, ..,10-cluster models.

The method was also compared with other often-used initialization methods, such as random initialization, K-means and fuzzy C-means clustering. The corresponding results are shown in Fig. 8a-c and the maximal total accuracies after 50 runs are 81%, 85% and 79%, respectively. A comparison has also been done with ordinary fuzzy C-means, which leads to only 44 % accuracy (Fig. 8d).

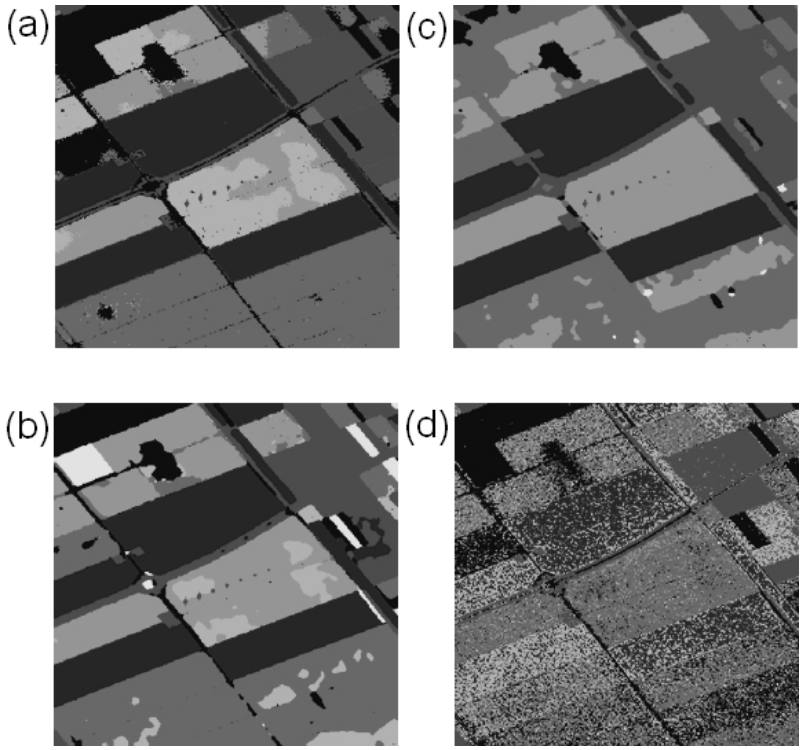


Figure 8. MRF clustering with (a) random initialization; (b) initialization by K-means; (c) initialization by the fuzzy C-means; and d) fuzzy C-means clustering.

We performed a further test using a supervised maximum-likelihood classification approach [31]. This obtained only 78.4% total accuracy. If segmentation is used as a pre-processing step, the classification accuracy is increased to 96.3%. The result is quite comparable to our unsupervised method, in which the class signatures are unknown beforehand.

As expected, not all pixels are well described by the clusters that are found. The O-image in Fig. 9 shows the outliers of the model. They partly consist of pixels from "unknown" classes (e.g. pixels in the upper-right region or road structures) or sensor speckles. One can further work on these pixels by using, for example, the incremental model-based clustering method described in [16] to identify addition classes and noise. The method takes into account the current model and new clusters in the set *O*. However, more discussion is not within the scope of this study.

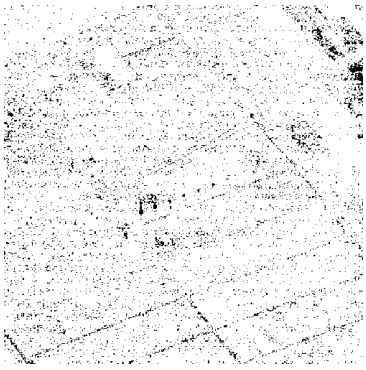


Figure 9. The O-image shows the outliers of the seven-cluster model.

In order to improve the classification results, speckle is normally reduced from the original image by de-noising schemes, such as moving average filtering or dedicated speckle

filtering. The drawback of filtering techniques is that the structure in the data may be affected. Our proposed algorithm, on the other hand, works directly on the original image. Outliers in classification results caused by speckle can be identified afterwards.

5. Conclusion and discussion

We have proposed in this work a fairly simple initialization method, which makes MRF clustering more robust and applicable for clustering of large remote sensing images, a very difficult task for any unsupervised classification method. The method works best for an image consisting of many large homogeneous regions, such as agricultural crops areas. Small and isolated clusters may not be recognized by the method. In this case, incremental model-based clustering is suggested as a post-processing step. In many cases, a good choice of the number of clusters may be identified by the use of PLIC. Prior information can also be used to determine the optimal model. The proposed method does not need pre-processing on the original image data. The method is totally unsupervised, which is the big advantage since in many cases ground truth is not available.

In this work, the method was applied to a polarimetric SAR image, utilizing the full polarimetric information content through a transformation described in [31]. The method shows excellent results. Our future work will focus on using other segmentation methods, such as region growing, in the first step. This can overcome the limitation of the current method to identify small and isolated clusters, and will significantly increase possibilities of the proposed approach on remote sensing applications.

Acknowledgements

We thank Simon Dodds for helping us to improve the English.

References

[1] J. Besag, “On the statistical analysis of dirty pictures”, *Journal R. Statistic, Soc.*, B, 1986.

[2] J. Besag, “Spatial Interaction and the Statistical Analysis of Lattice System”, *Journal R. Statistics, Soc. Ser.* B, 36, pp 192-236, 1974.

[3] W. Qian, D.M. Titterington, “Estimation of parameters in hidden Markov models”, *Philosophical Transactions of the Royal Society of London A*, vol. 337, pp. 407-428, 1991.

[4] G. McLachlan and D. Peel, “Finite Mixture Models”, Willey series in probability and statistic, Canada, 2000.

[5] C. Fraley and A. E. Raftery, “Model-based clustering, discriminant analysis, and density estimation”, J. the Amer. Statist. Asso., vol. 97, pp. 611-631, 2002.

[6] T.N. Tran, R. Wehrens and L.M.C. Buydens, “Clustering multispectral images: a tutorial”, *Chemom. Intell. Lab. Syst., in press.*

[7] A.H.S. Solberg, T. Taxt, A.K. Jain, “A Markov random field model for classification of multisource satellite imagery”, *IEEE Trans. on Geosci. Remote Sensing*, vol. 34, pp. 100 - 113, Jan. 1996.

[8] P.C. Smits, S.G. Dellepiane, ”Synthetic aperture radar image segmentation by a detail preserving Markov random field approach”, *IEEE Trans. on Geosci. Remote Sensing*, vol. 35, pp. 844 - 857, Jul. 1997.

[9] Q. Jackson, D. A. Landgrebe, “Adaptive Bayesian Contextual Classification Based on Markov Random Fields”, *IEEE Trans. on Geosci. Remote Sensing*, vol. 40, pp. 2454-2463, Nov. 2002.

[10] A. Sarkar, M. Kumar Biswas, B Kartikeyan, V. Kumar, K. L. Majumder, D. K. Pal, “A MRF Model-based Segmentation Approach to Classification for Multispectral Imagery”, *IEEE Trans. on Geosci. Remote Sensing*, vol. 40, pp. 1102-1113, May. 2002.

[11] R. Fjortoft, Y. Delignon, W. Pieczynski, M. Sigelle, and F. Tupin, "Unsupervised Classification of Radar Images using Hidden Markov Chains and Hidden Markov Random Fields", *IEEE Trans. on Geosci. Remote Sensing.*, vol. 41, pp. 675 – 686, Mar. 2003.

[12] C. Fraley, "Algorithms for Model-Based Gaussian Hierarchical Clustering", *SIAM J. Sci. Comput.*, vol. 20, pp. 270-281, 1998.

[13] A.P. Dempster, N.M. Laird and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm ", *J. R. Statist. Soc. B*, vol. 39, pp. 1-38, 1977.

[14] R. Wehrens, L.M.C. Buydens, C. Fraley and A.E. Raftery, "Model-based clustering for image segmentations and large datasets via sampling", Techn. Report no. 424, Dept. of Statistics, University of Washington, 2003.

[15] C. Posse, "Hierarchical Model-Based Clustering for Large Datasets", *Journal of Computational and Graphical Statistics*, vol. 10, pp. 464-486, 2001.

[16] C. Fraley, A. E. Raftery and R. Wehrens, "Incremental Model-Based Clustering for Large Datasets with Small Clusters", Techn. Rep. no. 439, Dept. of Statistics, University of Washington, Dec. 2003.

[17] T. N. Tran, R. Wehrens and L. M. C. Buydens, "SpaRef: A Clustering Algorithm for Satellite Imagery ", *Anal. Chim. Acta*, vol. 490, pp. 303-312, 2003.

[18] K.V. Mardia, J. T. Kent, J. M. Bibby, *Multivariate Analysis*. London, Academic Press, 1979.

[19] D. J. Hand, *Discrimination and classification*. John Wiley & Sons, 1981.

[20] R. E. Kass and A. E. Raftery, "Bayes factors and model uncertainty", *J. Amer. Statist. Assoc.,* , vol. 90, pp. 773-795, 1995.

[21] D.C. Stanford, A. E. Raftery, "Approximate Bayes Factors for Image Segmentation: The Pseudolikelihood Information Criterion (PLIC)", *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 24 ,pp. 1517-1520, 2002.

[22]. H. A. Yueh, A. A. Swartz, J. A. Kong, R. T. Shin, and L. M. Novak, "Bayes classification of terrain cover using normalized polarimetric data," J. Geophys. Res. B-12, vol. 93, pp. 15.261–15.267, 1998.

[23]. H. H. Lim et al., "Classification of earth terrain using polarimetric SAR images," J. Geophys. Res., vol. 94, pp. 7049–7057, 1989.

[24]. J. J. van Zyl and C. F. Burnette, "Baysian classification of polarimetric SAR images using adaptive a priori probability," Int. J. Remote Sens., vol. 13, no. 5, pp. 835–840, 1992.

[25]. J. S. Lee, M. R. Grunes, and R. Kwok, "Classification of multi-look polarimetric SAR imagery based on complex Wishart distribution," Int.J. Remote Sens., vol. 15, no. 11, pp. 2299–2311, 1994

[26]. J. J. van Zyl, "Unsupervised classification of scattering mechanisms using radar polarimetry data," *IEEE Trans. Geosci. Remote Sensing*, vol. 27, pp. 36–45, Jan. 1989.

[27]. S. R. Cloude and E. Pottier, "An entropy based classification scheme for land applications of polarimetric SAR," *IEEE Trans. Geosci. RemoteSensing*, vol. 35, pp. 68–78, Jan. 1997.

[28] A. Freeman and S. L. Durden, "A three-component scattering model for polarimetric SAR data," *IEEE Trans. Geosci. Remote Sensing*, vol. 36, pp. 963–973, May 1998.

[29]. J. S. Lee, M. R. Grunes, T. L. Ainsworth, L. J. Du, D. L. Schuler, and S. R. Cloude, "Unsupervised classification using polarimetric decomposition and the complex Wishart classifier," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 2249–2258, Sept. 1999.

[30]. J. S. Lee, M. R. Grunes, E. Pottier, and L. Ferro-Famil, "Unsupervised Terrain Classification Preserving Polarimetric Scattering Characteristics", *IEEE Trans. on Geosci. Remote Sensing,* vol. 42, pp. 722-731, 2004.

[31] D.H. Hoekman, and M.A.M. Vissers, "A new polarimetric classification approach evaluated for agricultural crops", *IEEE Trans. on Geosci. Remote Sensing*, vol. 41, pp. 2881-2889, Dec. 2003.

STRATEGIES FOR MIXTURE MODEL CLUSTERING OF MULTIVARIATE IMAGES

Abstract

Two novel strategies for mixture model clustering of multivariate images have been developed. Most other approaches require good guesses of the number of components (clusters) and their initial statistical parameters. In our approach, the initial parameters of mixture model clustering are determined by agglomerative clustering on homogenous regions, identified by region growing segmentation. One strategy is developed for a normal situation of mixture modelling, where the density of a cluster is modeled by a single normal distribution; the other is designed for a more complex situation, where the density of a single cluster is a mixture of several normal sub-clusters. The method is very robust to noise/outliers and overlapping clusters. It is also reasonably fast and suitable for moderate to large images. Experiments on both simple and complex data sets are presented.

Keywords: Mixture models; Clustering; Number of clusters; Spatial information

T.N. Tran, R. Wehrens, and L.M.C. Buydens, *revised for Journal of Chemometrics*.

1. Introduction

The mixture modelling approach to clustering plays a major role in exploratory data analysis in searching for groupings in the data [1][2]. The data to be clustered are usually described by a mixture of a number of Gaussian components and the clustering uses the Expectation-Maximization (EM) algorithm to fit the finite mixture model to the dataset [3][4]. However, the EM method is very sensitive to the initial estimate of the number of components and their statistical parameters (means and covariances) [5]. Several solutions in literature have addressed the problem. Fraley et al. [6][2] suggested obtaining the initial parameters values via model-based agglomerative clustering. However, "direct application of this initialization method to large datasets is often prohibitively expensive in terms of computer time and memory" [7]. Even a few thousand pixels may already be too large for convenient processing. To get out this situation, estimates of the statistical parameters can be derived from a small sample of the data. However, obtaining a representative sample is quite difficult in many cases [7]. Quite recently, for image data, the agglomerative process is sped up using segmentation techniques [8][9]. Here, an over-segmented image is produced as input to the agglomerative process. However, the final clustering result, obtained right after the agglomerative process, has the flexibility-problem [10]; once a pixel has been assigned to a cluster, it will not be considered for joining other clusters in later iterations.

Two new strategies are proposed in this paper to improve the mixture model clustering. The strategies combine agglomerative clustering and segmentation to obtain initial estimates for the subsequent mixture model clustering. Moreover, to deal with overlapping clusters and noise, the clustering result is filtered by a Markov Random Field (MRF)-based technique at the final step. The basic strategy (Strategy I) is used for data where clusters are normally distributed. However, it is frequently encountered in practice that cluster densities can be non-normal. Detecting non-Gaussian classes is a challenging task using Gaussian mixture model clustering. As suggested in [1] and recently in [17], non-Gaussian classes could be modeled by several Gaussian distributions. For this reason, we develop Strategy II for this "complex" situation. It aims to group Gaussian subclusters to form a complete component, and at the same time to retain very small clusters. Examples are given for two real-world cases: a multispectral image for minced meat, and an RGB image of St. Paulia flowers. The results are compared to other methods such as Fuzzy C-means [11] and mixture modeling clustering. In these cases, the spatial relations between pixels are ignored.

2. Previous works

2.1 Mixture model clustering

In brief, in mixture model clustering, the probability density function of the pixel x_i is given by:

$$f(x_i;\Psi)=\sum_{c=1}^{g}\pi_c f(x_i;\theta_c) \tag{1}$$

where g is the number of components, θ_c contains the means and covariances (μ_c,Σ_c) of cluster c, π_c is the mixture proportion, and Ψ contains all cluster parameters (θ) and mixture proportions (π). The form of the multivariate distribution function *f* is chosen

according to the underling distribution of the data set; usually a multivariate normal distribution is used.

If the data are not normally distributed, a mixture of normal distributions can still describe the cluster shape quite well [1]. We consider both situations in this paper.

For a dataset of n pixels, the mixture model clustering algorithm maximizes the complete-data log-likelihood function:

$$\log L(\Psi) = \sum_{c=1}^{g}\sum_{i}^{n} u_{ic} \log(\pi_c f(x_i;\theta_c)) \tag{2}$$

where u_{ic} corresponds to the conditional probability of object x_i belonging to cluster c. The Expectation-Maximization (EM) algorithm [3][4] is usually used to fit the finite mixture model to the dataset. At each iteration k, EM consists of two sub-steps, called an E-step and an M-step. The E-step (conditional Expectation step), estimating the conditional probability u_{ic}, is given by:

$$u_{ic}^{k} = P^{k}(x_i \mid c) = \frac{\pi_c^k f_c\left(x_i;\theta_c^k\right)}{\sum_{d=1}^{g}\pi_d^k f_d\left(x_i;\theta_d^k\right)} \tag{3}$$

In the M-step (Maximization step), the statistical parameters π_c and θ_c are estimated from the data [1].

Usually, several different models are fitted. To find the one that fits the data best, many different criteria can be used (see, e.g. [1]). One of the most popular criteria is the Bayesian Information Criterion (BIC) [12][13]:

$$BIC = 2\log L(\Psi) - d\log(n) \tag{4}$$

where d is the number of parameters of the model. The best model is indicated by a maximal BIC value. This corresponds to a model with few parameters that nevertheless fits well (high likelihood).

2.2 Estimation of initial cluster parameters by model-based agglomerative clustering

The quality of the clustering result by EM critically depends on the initial values, i.e. the number of clusters and their parameters. If a poor choice of initial values is made, the convergence of EM may be very slow [14], which is impractical for large image datasets.

Again, many different strategies have been proposed, e.g. several random starts, but the most reliable option seems to be to use model-based agglomerative clustering (MAC) [6]. This has the added advantage that once the cluster tree has been established, at a very low computational cost, several numbers of clusters can be assessed. MAC starts on singleton clusters, containing a single pixel. The parameters θ_{c_i}, the means and covariances (μ_c, Σ_c), for cluster c_i is now initialized by the spectral of pixel i and identity matrix I.

The algorithm then continues to join those pairs of clusters, which leads to the greatest increase in classification likelihood [2][6], L_{CL}, given by:

$$L_{CL} = \prod_{i=1}^{n} f\left(x_i;\theta_{c_i}\right) \tag{5}$$

Again, f is a multivariate Gaussian with parameters θ_{c_i} for cluster c_i to which x_i is assigned.

The number of clusters is decreased by one after each iteration. The process continues until there is only one cluster. This yields a dendrogram, presenting how cluster pairs are joined. The initial statistical parameters for mixture model clustering for several interesting models can be extracted by cutting the dendrogram at appropriate levels. Model-based clustering [2], proposed by Fraley and Raftery, essentially includes these two main steps: initialization of statistical parameters by using MAC at step one, and performing EM for the interesting models and selecting the best model using BIC at step two.

The initialization step of course can be done by using the ordinary agglomerative clustering such as single-linkage, average linkage and complete linkage. However, they have no known associated statistical model [2] that allows good estimates of the statistical parameters and the number of clusters, which is the main goal of our study.

3. Strategy I

The major drawback of initialization by agglomerative methods is that the computational demands increase rapidly with the number of samples, which makes it impractical for large data sets. In our research, we propose a solution particular to images. Instead of starting from individual pixels, the agglomerative initialization starts from a much smaller number of homogeneous areas. These areas are obtained by a simple region growing segmentation (RGS). Strategy I extracts cluster parameters for several selected models; e.g. in between 5 and 25 clusters, which are used as starting points for the EM iterations. The aim is only for reducing the computation time by discarding all the models out from this range. Normally, without any prior knowledge, this can be set to the maximum to the capacity of the computer system. However, with prior knowledge about the data this range can be much narrower.

Then, the best model is picked on the basis of the BIC criterion. Finally, the clustering of the complete image (pixels inside and outside the homogeneous areas) is obtained by MRF classification, the second new element in our strategy. It uses distances to the individual clusters as well as class information of neighboring pixels. The steps of Strategy I are given below.

Clustering Strategy I:

Step 1: Obtain homogenous regions by the region growing segmentation method.

Step2: Estimate cluster parameters for selected models by agglomerative clustering.

Step 3: Do EM for each selected model on homogenous pixels; the best model will be selected using BIC.

Step 4: Log-likelihood classification or Markov Random Field (MRF) classification on the entire image.

Note that from now on if the objects are not mentioned explicitly at any step of the algorithm, then it implies that the algorithm is applied to the pixels belonging to the homogenous regions.

Obtain homogenous regions

By definition, "image segmentation is a process of partitioning the image into non-intersecting regions such that each region is homogeneous and the union of no two adjacent regions is homogeneous" [15]. RGS starts with a number of initial seed pixels, and creates homogeneous regions by grouping adjacent pixels (or regions) if their distance (due to intensities in the channels) is below a predefined threshold. Defining this threshold is the most problematic issue in RGS. An over-fragmented image, containing many more regions than expected, is easily obtained. One reason for this is that object homogeneity is not a well-defined concept and may be described by different statistics. In reality, cluster homogeneity can be influenced by many external factors such as temperature, or experiment factors. In this work, over-segmentation is not a problem because regions will be merged later. Hence, determining a perfect threshold is not necessary. Several basic variants of RGS exist, depending on the definition of the distance between neighbor pixels and the segment of a current "seed" pixel. Here, for simplicity, we use the simple average linkage RGS where the distance between neighbor pixels and the mean of the current segment is used. The RGS algorithm employed here uses one parameter, the minimal size of a region (MINSIZE), is described below:

The RGS algorithm:

Step 1. (New segment) A still unlabeled pixel (which is not associated to any segment) is used as a seed pixel to initialize the set of seed pixels, and go to step 2. If all pixels are labeled, discard all very small regions with size < MINSIZE, and STOP.

In general, any unlabeled pixel can be used as seed pixel in this step. However, for speeding up the process, it is chosen as the first unlabeled pixel encountered when reading the image (row or column order)

Step 2. (Iterative growth) If the set of seed pixels is not empty, get one at the top of the set as a current seed pixel. All boundary pixels are eligible for merging (by reading order): they are joined to the current segment (i.e. they are labeled) AND appended to the end of the set of seed-pixels if:

1. They are unlabeled pixels,
2. The distances to the mean of the current segment are below the variance of the current segment.

If the set of seed pixels is empty, go back to step 1, otherwise loop to the beginning of step 2.

Very small regions with sizes smaller than MINSIZE may well contain noise, artefacts or spatially isolated pixels. These pixels are not important for parameter estimation purposes, and therefore they are discarded from the process until the last step of the clustering strategy. Otherwise, they may influence the clustering process. The smaller MINSIZE, the more homogeneous region will be found. It should be kept small but larger than the number of dimensions. As a rule of thumb, MINSIZE may be taken as twice the number of spectral variables in the data set. In our experience, this works well for different data sets.

Artefacts and noise are typically not present in homogeneous areas, which will improve the quality of the estimated statistical parameters in later stages of the algorithm. Note that there is a chance that some clusters are not found; this may be the case if no homogeneous areas corresponding to these clusters are identified. However, this may happen in any

clustering, albeit for different reasons. The chance is very small if MINSIZE is small enough.

Model-based clustering of homogenous regions

Step 2 and step 3 of the algorithm is in fact the model-based clustering [2] using the most general model (variable in volume, shape, and orientation - VVV) . The initialization is performed using the homogeneous regions obtained from step 1 rather than from singleton pixels. In brief, after MAC on the homogeneous regions, the statistical parameters for the range of interesting models, can be obtained. These are used to start EM for the selected models. Then, in step 3, BIC values for all selected models are calculated and the best model is identified by the highest BIC.

MRF classification for the entire image

At this point, step 4, the best model is identified with statistical parameters of all clusters. We can use these to obtain a classification for all pixels in the image, not only the pixels in the homogeneous areas, by maximal likelihood classification (one E-step) [2]. However, the result may be improved further by taking into account the spatial relation between pixels. Therefore, we propose to apply a MRF step to deal with overlapping clusters and noise [1][10][16]. Basically, in MRF clustering, the conditional probability of point xi belonging to cluster c, $P(c_i = c)$, under the neighboring system ∂_i (usually a 3 x 3 or a 5 x 5 rectangular window) is estimated by [8]:

$$P(c_i = c) = \frac{1}{Z}\exp\left[\beta \sum_{j\in\partial_i} u_{jc}\right] \tag{8}$$

where Z is a normalization constant and β is a spatial smoothness parameter. More details can be found in [1][16].

A higher (positive) β corresponds to higher spatial dependency of neighbor pixels. In practice, it is normally set in the range of [0.1, .., 4]. The more positive the value, the smoother the result image can get. However, over-smooth images could lose small and isolated parts of clusters. Therefore, the user has to find a compromise.

The EM algorithm is then adapted, leading to the the complete-data log-likelihood criterion:

$$\log L_{MRF}(\Psi) = \sum_{c=1}^{g}\sum_{i}^{N} u_{ic}\log(\pi_{ic} f(x_i;\theta_c)) \tag{9}$$

where the mixture proportions π_c (Eq. 2) are now replaced by the transition probability π_{ic} [1]:

$$\pi_{ic} = \exp\left(\beta \sum_{j\in\partial_i} u_{jc}\right) / \sum_{h=1}^{g}\exp\left(\beta \sum_{j\in\partial_i} u_{jh}\right) \tag{10}$$

Here, we need only the E-step. The conditional probabilities u_{ic}, and π_{ic} taking into account spatial information, are changed. This is different from the MRF concept in [10], where MRF is integrated directly into full E- and M- steps. Here, the statistical parameters θ_c are kept constant. The convergence is usually obtained in very few iterations; in most cases two iterations suffice.

Two clusters that overlap in the spectral domain but are in different regions of the image may be separated easily in this way. Moreover, isolated noise pixels are classified to one of the classes present in their neighborhood, which leads to a much smoother clustering result [10].

4. Strategy II

In practice, the classes to be identified by the clustering, are often not normally distributed but still can be described very well by a mixture of several normal distributions [1][17].

In this situation, it is hard to determine the best cluster-model determined by using BIC criterion, because no clear maximum may be present. It could result in many more clusters than expected. Especially, it is the case for a dataset containing a non-normal big cluster as well as several small clusters. By the likelihood criterion in agglomerative clustering (step 2)[2][6], small clusters are likely to be merged to other clusters very early. This is the explanation for the problem of detecting small clusters in large datasets using model-based clustering [7]. On the other hand, it is very hard to join sub-clusters of a big cluster into one component.

Hence, we propose strategy II, which is an extension of strategy I only in the step 3. Step 3 is extended to better identify the number of clusters and their statistical parameters for the best cluster model. Now the step 3 in Strategy I is step 3.1 in Strategy II.

After step 3.1, an intermediate best cluster model containing many clusters is obtained. At this point, sub-clusters of each component should be merged. We start with the assumption: "*if a component contains many normal clusters then they must be highly overlapping (very similar)*". Then we do the merging by looking into the degree of overlap of all pairs of clusters. One of the ways to measure the overlap is by using the Bayes error. It is often modeled by the Bhattacharyya distance, Bha, [15] below:

$$Bha\ (i,j) = \frac{1}{8}(\mu_i - \mu_j)^T \left(\frac{\Sigma_i + \Sigma_j}{2}\right)^{-1} (\mu_i - \mu_j) + \frac{1}{2}\ln\left(\frac{\left|\frac{\Sigma_i + \Sigma_j}{2}\right|}{\sqrt{|\Sigma_i||\Sigma_j|}}\right) \qquad (12)$$

where i,j are two clusters with means μ_i, μ_j and covariance matrices Σ_i , Σ_j, respectively.

The distance is positive number. A higher overlap leads to a higher Bayes error and therefore a lower Bhattacharyya distance. Step three of the strategy I is then replaced by:

Extension of step 3:

3.1. Do EM for each selected model (usually models with a large number of clusters); the best intermediate model is selected using BIC plot and for non-Gaussian dataset, it usually comes up with a very high number of clusters. This is actually the step 3 of strategy I objective to very high number of clusters.

3.2. Apply agglomerative clustering to the intermediate model using the Bhattacharyya distance, to obtain cluster parameters for a number of interesting models.

3.3. Do EM for each model and again the best model can be selected using BIC plot.

Since the number of expected clusters is far less than the normal distributions needed in the complex data set, the final cluster model of Strategy II is not the best fitted to the data and the BIC value is not the highest. However, instead of focusing on the exact description of one or two large clusters, strategy II tries to model smaller clusters, which do not influence the likelihood that much, as well.

Strategy II differs from Strategy I only in step 3. In general, Strategy II can be directly applied to the dataset without prior information about class distributions. The BIC plot at the step 3.1 determines the next steps, whether to continue with Strategy II or use Strategy I instead. If the plot shows the maximal BIC values already at a very low number of clusters, then classes are expected to be Gaussians and Strategy I can be used.

5. Results

5.1. Minced meat data set.

The first example is a multivariate image of minced meat of 318x318 pixels with 257 variables (bands) from 396 nm to 736 nm (1.3 nm for each band), recorded with the ImSpector V7 imaging spectrograph (Spectral Imaging Oulu, Findland) [18]. The incoming light is split and captured by a Sony CCD camera to obtain a color image, which will be used as the reference image (Fig. 1a). In order to reduce computation time, the number of variables is reduced to 11 bands by averaging [10]. The data set contains 4 classes: the petri disk, dark meat, light meat and fat. The difference between dark meat and light meat is caused by the amount of blood in the meat. The dark pixels represent the dark meat class and the white spots represent the fat class. The clustering of original image is reported in [10]. We demonstrate the ability of the method of dealing with noise. Therefore, White Gaussian noise with a standard deviation of 50% of the average standard deviation of the entire image is added to the spectra of the image (Fig. 1b).

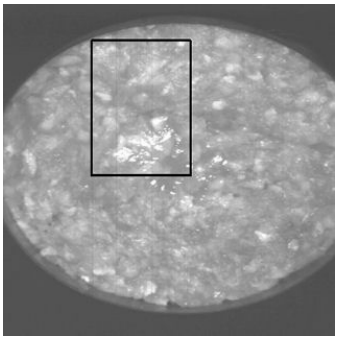
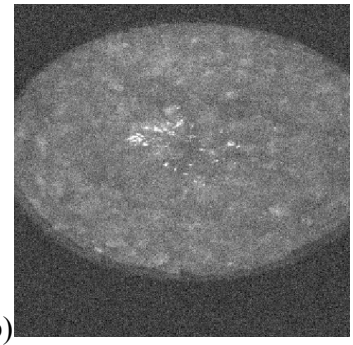
a)  b) 

Figure 1. (a) the reference CCD color image; (b) The composite image (band 2,3 and 9) of the noise image.

Noise and overlapping clusters are the two main problems of this data set. Indeed, the results obtained by using fuzzy C-means and mixture modelling by EM (with a random initialisation, repeated 50 times, an unconstrained VVV model, and ignoring spatial information) are very poor, as shown in Figures 2a and 2b, respectively.

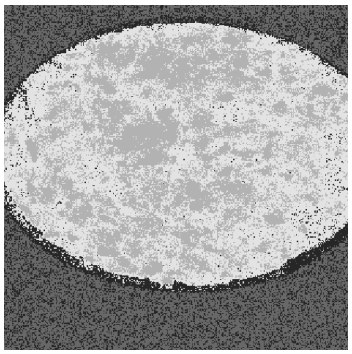
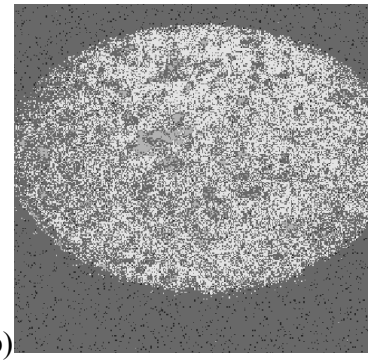
a)  b) 

Figure 2 Clustering on the noise image by (a) Fuzzy C-means on noise image; (b) Mixture models clustering by EM.

A standard solution to the noise problem is to preprocess the image by smoothing or filtering techniques. However, this tends to increase the overlap problem [10]. It is illustrated in Figure 3a, where the noisy image is filtered by the often-used median filtering technique with a 3-by-3 neighborhood. The clustering results for the filtered data by the fuzzy C-means (Fig. 3b) and the EM algorithm (Fig. 3c) are still not very good. In fuzzy C-means, the fat spots and dark meat regions are covering much of the light meat regions; the regular EM algorithm mixes the dark meat with other classes.

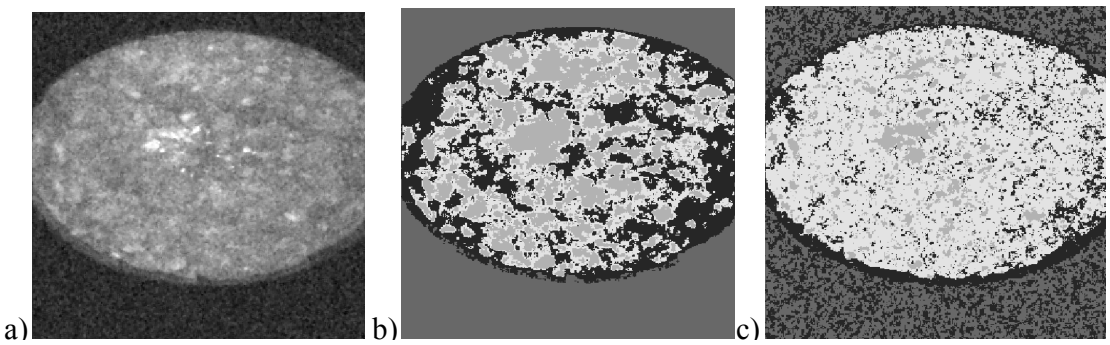


Figure 3. (a) The filtered image using the Median filtering; and clustering on the filtered image by (b) Fuzzy C-means on noise image; (c) Mixture models clustering by EM.

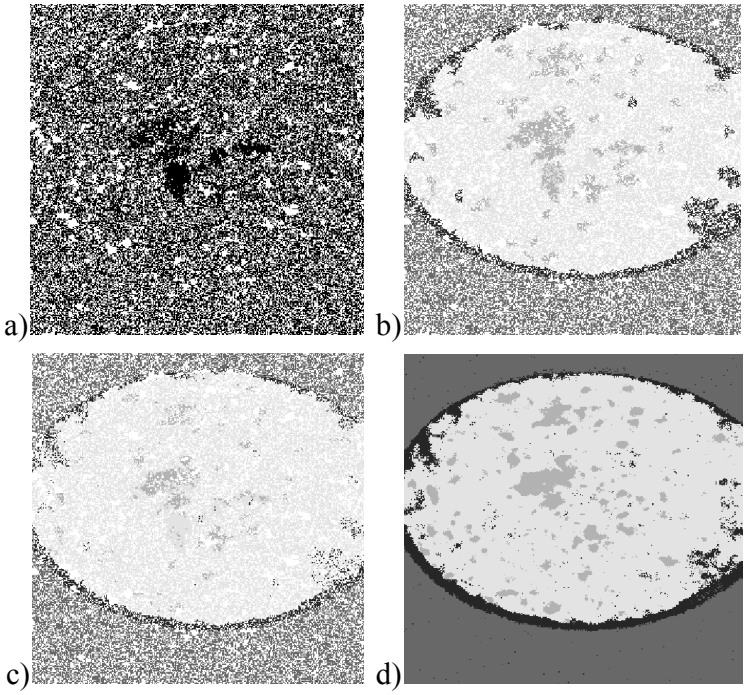


Figure 4. (a) Step 1: 86 regions have been obtained by RGS (Black regions); (b) Step 2: The four-cluster model after MHC; (c) Step 3: The four-cluster model after EM; (d) The four-cluster model after MRF classification extension to the entire image.

The first step of strategies was applied on the noisy image. 86 homogeneous regions are obtained by RGS with MINSIZE = 22 (twice the number of feature dimension). The image of these regions is plotted in Figure 4a (black areas). The white areas represent pixels

outside the homogeneous regions. In step two, MAC is applied to the homogeneous regions and seven interesting models (ranging from 2 to 8 clusters) are extracted from the dendrogram. The four-cluster model is shown in Figure 4b. Then EM is applied to all selected models in step three to obtain statistical parameters. The BIC values of several models are shown in Figure 5. After a seven cluster-model, BIC values are decreasing. The plot shows the maximal BIC values already at a four cluster-model, so classes are expected to be Gaussians and Strategy I is suitable for this dataset.

The four-cluster model has the highest BIC value, which is in agreement with the reference information. Upon obtaining the best model and the corresponding cluster parameters, the final clustering result is obtained after MRF classification on the entire image, using β=0.3 and a 5x5 neighboring system (Figure 4d). The clustering result is very good, considering the amount of noise in the image. Especially, it is important that the "fat" regions coincide with regions of light spots in the reference image (Figure 1a).

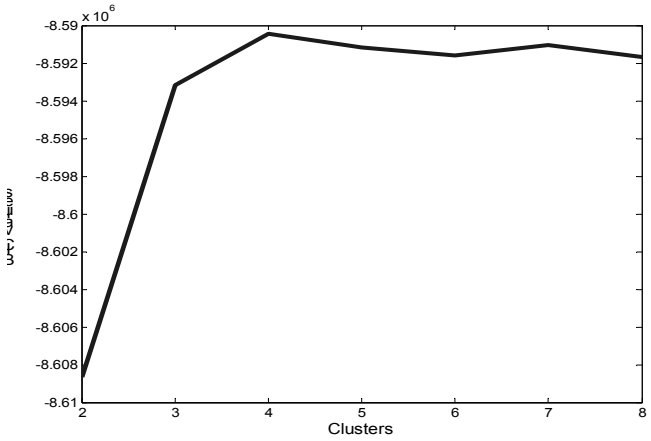


Figure 5: The BIC plot of Strategy I on the minced meat data set.

5.2. St. Paulia Flower Image Data

The RGB (3-band) image (304 x 268 pixels) of a St. Paulia flower is shown in Figure 8a. Since the yellow centers of the flower are very small (many yellow spots have sizes smaller than 4 pixels) detecting them using clustering with a small number of clusters is a challenging task. Incremental model-based clustering was proposed in [7] as one of solutions for this problem. In the current paper, Strategy II is used for this dataset. In step one, 419 homogeneous regions are obtained by RGS with MINSIZE = 6 (twice the number of feature dimensions). In step two, MAC is applied to the homogeneous regions and a wide range of numbers of clusters of [2,..,80] is selected to start step three. EM is applied to all selected models. The BIC values are shown in Figure 6. The maximal BIC value is in a high number of clusters and the best model is very difficult to determine, confirming that this is a complex dataset.

An intermediate-model of 61 clusters is selected (Step 3.1) corresponding to the highest BIC value. Step 3.2 and 3.3 are applied to the intermediate model for a range of [2,..30] clusters. Figure 7 plots the BIC values for this step (Strategy II) against the values found in step 2 (equivalent to Strategy I). The best model could be obtained by a sharp increase of BIC value. Three suitable options for Strategy II are at the locations of A, B, and C (in the BIC plot) corresponding to the models of 15, 22, and 25 clusters, while in the same range,

the models, located at D and E are suitable for the best model for Strategy I corresponding to the models of 21 and 29 clusters.

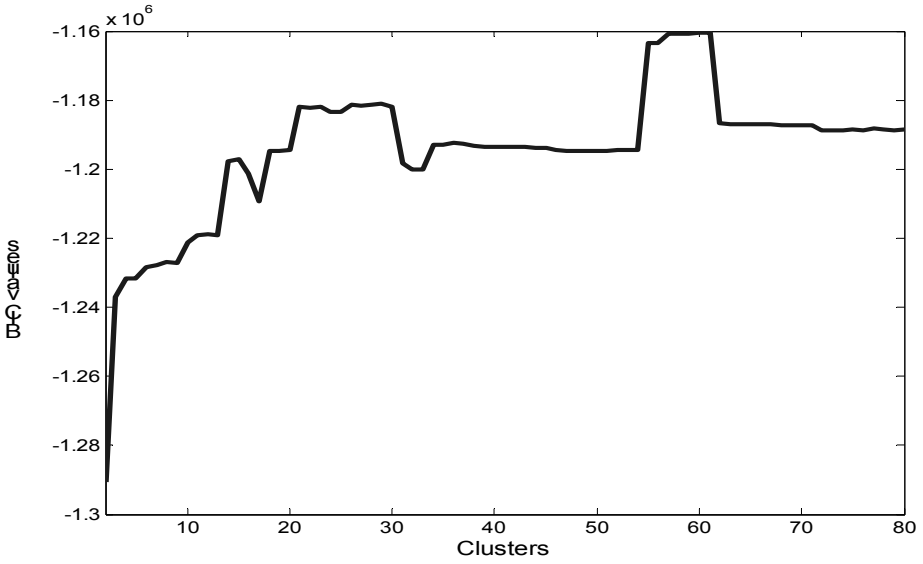


Figure 6. BIC values for cluster-models from 2 to 80 clusters of step 3.1 (Strategy I) on the image of St. Paulia flower.

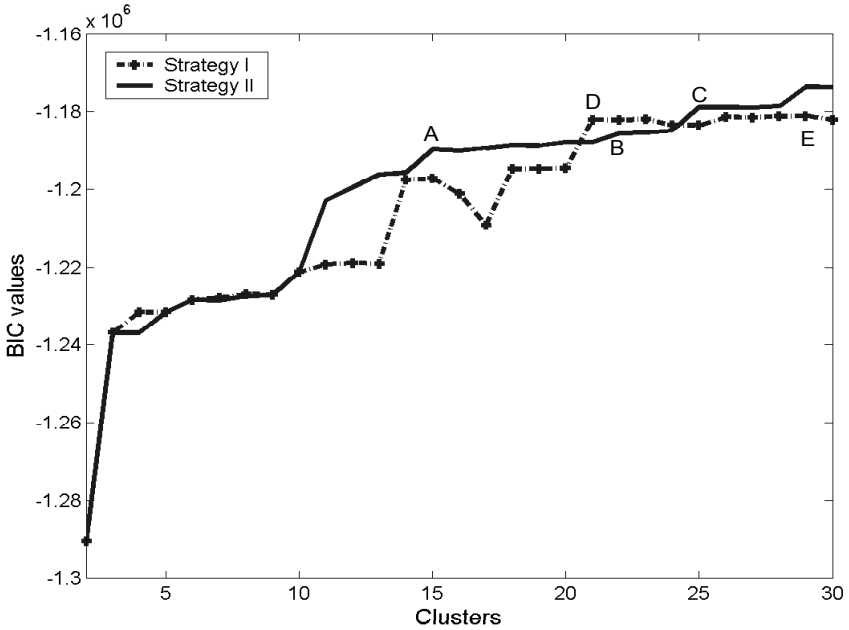


Figure 7. BIC values of Strategy II against the Strategy I on the image of St. Paulia flower.

In the final step, the statistical parameters of clusters are extracted for each option and maximal likelihood classification (one E-step) is used to obtain the corresponding clustering result. Some results for chosen models are plotted in Figure 8. The yellow centers are revealed well only by Strategy II on the B and C models corresponding to 22 and 25 clusters. The results of two models, Strategy II to 22 clusters (option B) and Strategy I to 29 clusters (option E), are shown in Figure 8b and c, respectively. The

clustering was also performed by fuzzy C-means to 30 clusters (Figure 8d), which cannot show the yellow centers either.

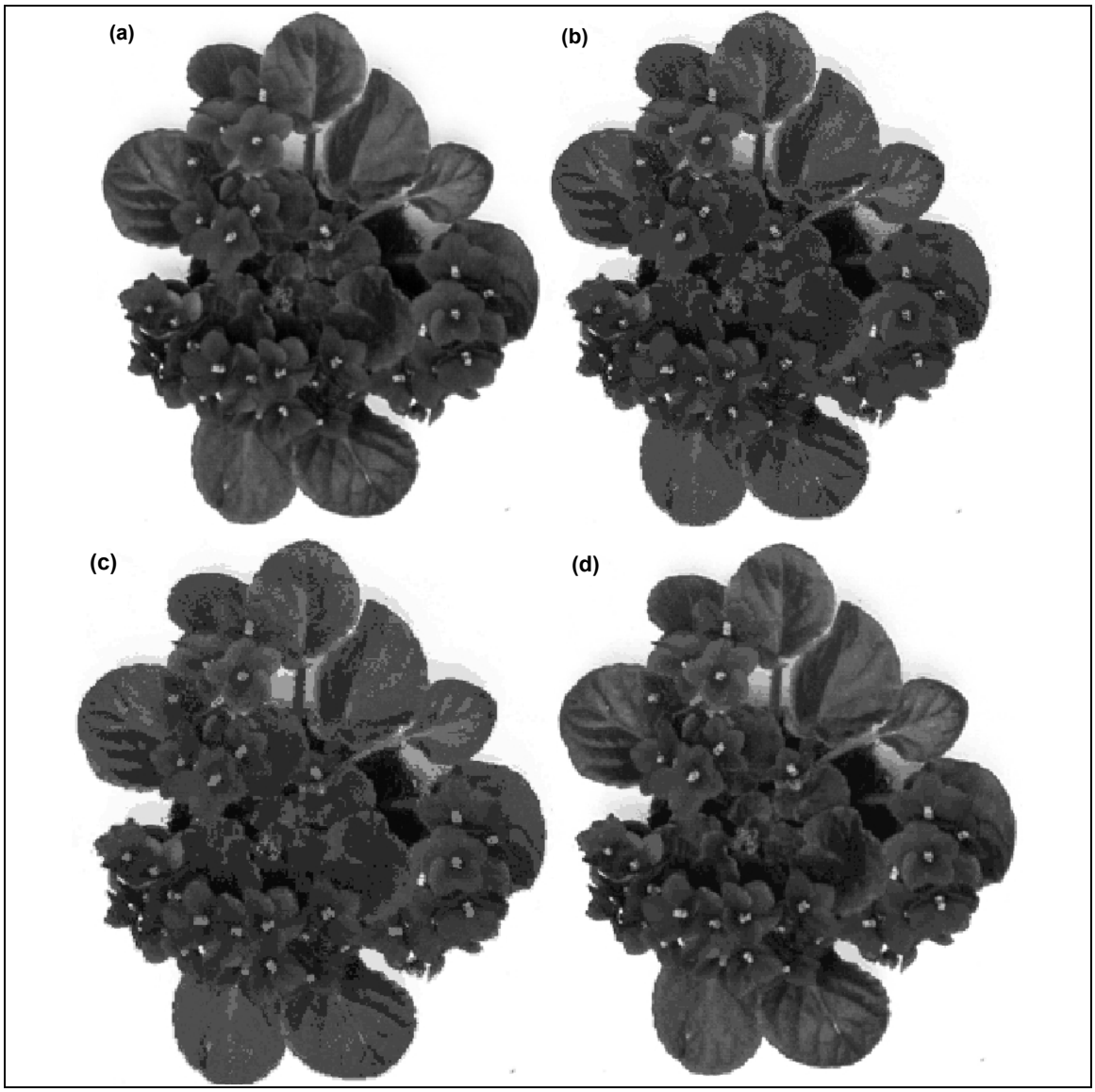


Figure 8. a) RGB Image of St. Paulia flower b) Strategy II to 22 clusters (option B), c) Strategy I to 29 clusters (option E), and d) fuzzy C-means to 30 clusters.

6. Conclusions and discussion

Two strategies (Strategy I and Strategy II) to mixture model clustering for multivariate images have been developed in this study. Strategy I is for a image data where each cluster is normally distributed, and the other for the situation where a cluster is a mixture of several Gaussian distributions. The methods minimize the need for human interaction to select values of input parameters. Spatial information is effectively used. Firstly, in the first stages it considers only homogenous regions, formed by RGS, which not only makes the process faster, but allows for reliable estimation of cluster parameters. Secondly, by employing MRF classification in the final step, it reduces the effect of noise/artifacts and the overlap problem of clusters, often present in real-world data sets. Since the number of homogenous regions is much smaller than the total pixels, the MAC process should be fast.

The EM algorithm is known to have linear of rate of convergence, which normally can be very slow due to wrong initialization [7]. In our study, the estimated initial parameters for EM should very close to the "true" signatures of the components and the EM algorithm must converge very soon, practically lest than 10 steps. Especially with Strategy II, Gaussian sub-clusters of one component and other very small clusters are very well recognized.

There are several adjustable parameters that should be set by the user. Many of these aim to reduce computation time by limiting the algorithm to an interesting range of models. The parameter MINSIZE in RGS should be small, but larger than the number of dimensions. In some applications, where the clustering result needs to be smooth, such as in the Meat dataset, MRF classification can be used in the final step. Smoothing parameters in this case can be adjusted easily since the statistical parameters of clusters are unchanged.

Without prior information, the number of clusters is suggested by BIC plots. For a simple dataset, it can be identified easily by the maximal BIC value. However, the maximal values could be at a very high number of clusters. Therefore, using the extension of Strategy II, meaningful clusters could be retained in a cluster-model with only few clusters.

The implementation of several parts of strategies I and II may be replaced by alternative procedures, in particular the thresholds that are used to focus the methods to relevant clustering models. In our experience, this hardly influences the results of the algorithms.

In essence, both two strategies are fast enough to be used for moderate-size and large multivariate images. The strategies are recommended for dataset of not very high dimension because EM algorithm could break down due to singularity problem of the estimated covariance of cluster. When the data are of high dimension, dimension reduction techniques, such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), or Self-Organizing Map (SOM), are inevitable.

On experiments, the proposed strategies gave very good performance on both real world data sets with difference types of problems.

Acknowledgements

We thank Jacco C. Noordam, Department of Production & Control Systems, Agrotechnological Research Institute (ATO), for sharing the data sets.

References

[1] G. McLachlan and D. Peel, "Finite Mixture Models", Willey series in probability and statistic, Canada, 2000.

[2] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation", J. the Amer. Statist. Asso., vol. 97, pp. 611-631, 2002.

[3] A.P. Dempster, N.M. Laird and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm ", J. R. Statist. Soc. B, vol. 39, pp. 1-38, 1977.

[4] R. J. A. Little and D. B. Rubin, "Statistical analysis with missing data". Wiley, New York, 1987.

[5] W. Seidel, K. Mosler, and M. Alker, "A cautionary note on likelihood ratio tests in mixture models", Ann. Inst. Statist. Math., vol. 52, No. 3, pp. 481-487, 2000.

[6] C. Fraley, "Algorithms for Model-Based Gaussian Hierarchical Clustering", SIAM J. Sci. Comput., vol. 20, pp. 270-281, 1998.

[7] Chris Fraley, Adrian Raftery, and Ron Wehrens, "Incremental model-based clustering for large datasets with small clusters", Journal of Computational and Graphical Statistics, vol. 14, pp. 1-18, 2005.

[8] L. Hermes, J.M. Buhmann, "Boundary-Constrained Agglomerative Segmentation", IEEE Trans. Geosci. Remote Sensing, vol. 42, pp. 1984-1995, Sept. 2004.

[9] D.W. Paglieroni, "Convergent Coarseness Regulation for Segmented Images", IGARSS Proc., Alaska, Sep. 2004.

[10] T.N. Tran, R. Wehrens and L.M.C. Buydens, "Clustering multispectral images: a tutorial", Chemom. Intell. Lab. Syst., 77/1-2, 3-17, 2005.

[11] J.C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms", Plenum, New York, 1981.

[12] G. Schwarz, "Estimating the dimension of a model", The Annals of Statistics, vol. 6, pp. 461-464, 1978.

[13] R. E. Kass, and A. E. Raftery, "Bayes factors", Journal of the American Statistical Association, vol. 90, pp. 773-795, 1995.

[14] G. J. Mclachlan, and T. Krishnan, "The EM algorithm and extensions", Wiley, 1997.

[15] N.R. Pal and S.K. Pal, "A review of image segmentation techniques", Pattern Recognition, vol 26, pp. 1277-1294, 1993.

[16] J. Besag, "On the statistical analysis of dirty pictures", Journal R. Statistic, Soc., B, 1986.

[17] Jia Li, "Clustering based on a multi-layer mixture model", Accepted by Journal of Computational and Graphical Statistics.

[18] J.C. Noordam and W.H.A.M. van den Broek, L.M.C. Buydens, "Multivariate image segmentation with cluster size insensitive ", Chemom. Intell. Lab. Syst., vol. 64 pp. 65-78, 2002.

CONCLUSION, DISCUSSION AND FUTURE PROSPECTS

7.1 Conclusion and discussion

The influence of spatial information on clustering has been extensively studied in this thesis (*Question 1*). Not only can spatial information be used for filtering/smoothing a noisy dataset as usual, but it has been shown in this research that the spatial information can also be used in clustering, firstly, to improve classification accuracy by reducing critical problems of clustering and, secondly, to speed up the clustering process for large images. Several clustering techniques have been proposed in the study for multivariate/multi-spectral images, taking into account spatial information. They are quite simple, robust with respect to the input parameters, show a fast convergence for large multi-spectral images, and most importantly, obtain more accurate results.

- More specifically, the biggest advantage of using spatial information is for extracting the initial parameters for partitional clustering (e.g. Expectation Maximization, fuzzy C-means). The initialization allows for better and faster convergence of the mixture model clustering (**Chapters 5 & 6**).
- The spatial information can be used to speed up a clustering process by selecting a representative sample out of the entire image (Chapter 6). It makes it possible to apply hierarchical-like clustering methods, such as model-based clustering, to large images (*Question 2*). Moreover, by applying clustering only on representative regions, outliers and noise are not included to the process. Then, the estimated parameters are expected to be more robust and accurate.
- The spatial information can also be integrated into the clustering procedure to reduce the influence of overlapping clusters and noise, the critical problems for spectral-only clustering (**Chapter 5 & 6**).

Many other problems can also influence the clustering. A general guideline provided in **Chapter 2** is useful for users to make a decision on the clustering method to use for a specific image dataset.

Answering to *Question 3*, high spectral dimensions and very different clustering densities are treatable by a density-based clustering named KNNCLUST, proposed in **Chapter 3**. Spatial information, however, is not considered in the method.

To answer to *Question 4*, the number of clusters is identified automatically by validation techniques in all proposed clustering approaches, except in KNNCLUST; the compactness criterion in **Chapter 4** and the Pseudolikelihood Information Criterion (PLIC) (Stanford and Raftery, 2002) and the Bayesian Information Criterion (BIC) (Schwarz, 1978) in **Chapter 5,** and **6**, respectively.

Although the proposed methods have been shown excellent results on most of experiments, some specific discussion points can be derived:

- *"No perfect clustering method for all images"* The statement is still true in our study. For example, in high-dimensional problems, KNNCLUST can work well. However, since KNNCLUST is sample-based, the method suffers from the overlapping clusters

problem. On the other hand, the problem of overlapping clusters was not a case for the modified mixture model clustering (**Chapter 5** & **6**), using spatial information. This method in turn is sensitive to very high dimensions due to the singularity problem of the estimation of covariance in the EM algorithm (**Chapter 2** & **6**). However, when the images are of high dimension, some dimension reduction strategy is recommended, e.g. by Principal Components (Smyth, 2000) or wavelet transformation (Murtagh et al., 2000).

- *"Can we use spatial information for other than partitional clustering; hierarchical and density-based clustering methods?* This is still an open question. In this study, spatial information was used during partitional clustering. However, we think that this information could also used to enrich a capability of other clustering types as well.

- The final number of clusters can be identified successfully by validation techniques in many applications, however, some sort of evaluation step still is inevitable for a more complex dataset. Visualization of the clustering result can also provide more information on the quality of the result. The first option is to map the data feature space to a latent space of at most three dimensions, for example by Principal Components Analysis (PCA). The first two or three principal components are normally used. However, the original feature space is changed, which does not ensure an unchanged cluster structure. For probability-based methods such as mixture modelling, uncertainties of probabilities can be used to build a mixture image. Uncertainty-image may be viewed for a single cluster at a time (gray-scale) (Wehrens et al., 2002), or three different clusters can be viewed together simultaneously, one cluster in each color (RGB). The overlapping areas would be in mixed colors rather than a pure RGB color.

7.2 Recommendations and further research

All our recommendations below will become subject for future research.

- Spatio-temporal images are different from multispectral images to relate to space and time together (having both spatial extension and temporal duration). The spatio-temporal image contains a sequence of still images in the time domain and, in addition to the spatial information in still images, 3D spatio-temporal pixel neighbourhood information is available. For example, in computer vision, motion analysis [Mitiche and Bouthemy, 1996, and Huang and Tsai, 1981] of video objects is an interesting application. Examples are tracking a person walking, a waving hand, a rotating wheel, ocean waves, and a flying bird in moving video. An image sequence from a video is a collection of single frames that have been recorded at consecutive points in time. This application needs the ability to study objects based on still images as well as their "movements" in time, the temporal direction. However, not many approaches employ the underlying spatio-temporal structure to classify objects with their motions on several frames simultaneously. We would recommend an extension of our proposed strategy (Chapter 6**)** to spatio-temporal data, the homogeneous regions now being defined in the spatio-temporal domain. Particularly, the region growing segmentation technique can be extended by tracking regions over time (temporal axis). Then, the initialization of clusters parameters and mixture model clustering can apply on these representative regions. The MRF refinement steps can also be modified in the same way. Other applications, like the spectroscopic data produced by 3-dimensional magnetic resonance spectroscopy (MRS) of the prostate (Simonetti, 2004) and many other applications, can also be seen as spatio-temporal image data. The most important advantage of the

proposed method is that the "objects" can be visualised in a sequence of frames of images and their "motions" and many of their interacting characteristics can be extracted. The immediate applications for example would be an enhancement of animated cartoon movies, and colouring black-and-white movies.

- More often than not, objects have a texture (a periodical repetition of a "homogenous" pattern). For example, in an image of a corn field, the corn field appears as a texture of parallel corn beds (each individual bed is identified as homogenous region). In this case, texture is very important information for clustering in order to recognize the corn field as whole. In literature, a usual solution for texture classification is by using filtering techniques, e.g. Gabor filters (Idrissa and Acheroy, 2002) or wavelet transform (Acharyya and Kundu, 2001), as feature extraction methods to map the original image space to a feature space, where a texture region can be seen as homogenous region. Then, clustering can be used. However, the current forms of the transformation methods (working only on gray images) can not be used for multispectral images. We recommend another approach by applying a clustering method on the original multispectral images to identify "homogenous" patterns, e.g. corn beds, and finally, a post-processing is needed to form all beds and neighbouring non-beds regions into a texture area by taking into account a context of texture characteristics; such as periodicity. The spatial information can be used in all steps.

References

Acharyya, M. and Kundu, M.K. (2001) An adaptive approach to unsupervised texture segmentation using M-band wavelet transform. Signal Processing 81 1337-1356.

Fraley, C., Raftery, A. and Wehrens, R. (2005) Incremental model-based clustering for large datasets with small clusters", J. Comput. Graph. Statist, in press.

Huang, T. S. and Tsai, R. Y. (1981) Image sequence analysis: Motion estimation. In T. S. Huang, editor, Image Sequence Analysis, volume 5 of Springer Series in Information Sciences, Springer-Verlag, Berlin, Heidelberg, 1-18.

Idrissa, M. and Acheroy, M. (2002) Texture classification using Gabor filters. Pattern regognition letters. 23 1095-1102.

Mitiche, A. and Bouthemy, P. (1996) Computation and analysis of image motion: A synopsis of current problems and methods. International Journal of Computer Vision, 19(1) 29-55.

Murtagh, F., J.-L Starck, and M. W. Berry (2000). Overcoming the curse of dimensionality by means of the wavelet transform. The Computer Journal 43, 107-120.

Simonetti, A. (2004) Investigation of brain tumor classification and its reliability using Chemometrics on MR spectroscopy and MR imaging data. Phd. thesis, Radboud University of Nijmegen.

Smyth, P. (2000). Model selection for probalilistic clustering with cross-validated likelihood. Statistics and Computing 10, 63-72.

Stanford, D.C., and Raftery, A.E. (2002) Approximate Bayes Factors for Image Segmentation: The Pseudolikelihood Information Criterion (PLIC). IEEE Trans. on Pattern Anal. Mach. Intell. (24) 1517-1520.

Wehrens, R., Simonetti, A.W. and Buydens, L.M.C. (2002) Mixture modelling of medical magnetic resonance data. J. Chemom. 16 274-282.

SUMMARY

In many image applications, huge image data sets are collected, which do not allow manual processing. Remote sensing is an example. Automatic analysis techniques such as clustering make it possible to analyse more and larger data sets. However, most clustering methods take only spectral information of images into account. The objective of this thesis is to study a the extension of clustering techniques to moderate and large multivariate/multi-spectral images utilizing the advantages of spatial information. The main interest is to improve the robustness of clustering methods (with respect to input parameters and the number of classes), and the accuracy by reducing the influence of the problems of overlapping clusters and noise on (but not limited to) remotely sensed images.

Chapter 2 provides a tutorial, which is a broad survey of the most basic clustering techniques for multivariate images, and gives guidelines to determine the most appropriate clustering for a particular multivariate image data set, depending on "image data problems". In many cases, partitional clustering techniques, taking into account spatial information, form the best option for a large image and can deal better with noise and outliers. The tutorial shows also a problem still remaining for clustering multivariate images, which requires a good setting of input parameters. Automatic settings do not always give a good result. In many cases, the setting can be obtained by a "trial and error" strategy and personal experience. This work is more difficult for a larger image, when more than one set of parameters may be required. Furthermore, clustering multivariate images always has to deal with the large data problem due to the development of imaging technology.

Chapter 3 presents a new clustering algorithm (KNNCLUST) to deal with the well-known problem of different densities of clusters in a high dimensional feature space for density-based clustering algorithms. KNNCLUST is a very good tool to cluster a small-sized multivariate data set provided that the clusters are not very different in size.The knn-kernel density estimation technique with Triangular and Gaussian kernels is used by KNNCLUST. KNNCLUST has only one parameter, which is the number of neighbors, k. In most cases, it is not difficult to find a range of k for which clustering results are stable. The number of clusters is automatically determined by KNNCLUST. Due to the calculation of the knn distance matrix, the computational complexity for the algorithm is quite high. However, in practice, indexing techniques could be used to improve the computation. KNNCLUST is suitable for high-dimensional data sets.

Starting from Chapter 4, spatial information is elaborately studied in clustering methodology. Spatial information actually has been used as a pre- and post-processing technique, referring to the filtering and smoothing of an image containing noise and artefacts, and the clustering result, respectively. In this thesis, spatial information can also be used in many places; initialization of the clustering parameters, during the clustering process, or filtering the clustering result at the final stage using spatial information together with the structure clusters.

Chapter 4 proposes SpaRef as a clustering algorithm for hyperspectral images, which is the first algorithm in my study taking into account spatial information. This method is a combination of K-means and Ward's method, in which the Ward's method (agglomerative

hierarchical clustering) is applied on a moderate number of highly homogenous classes, obtained by K-means (a partitional clustering). At the end, spatial information is used to correct the potential shortcomings of the Ward's method by introducing a refinement process. The proposed clustering method has the advantages to be stable, and leads to clusters with a high degree of compactness and continuity.

Chapter 5 investigates a possibility of using the spatial information in different way for initialization of Markov Random Field (MRF) clustering. The clustering is then more robust and applicable for large remote sensing images, consisting of many large homogeneous regions; such as agricultural crops areas. Small and isolated clusters may not be recognized by the method. An optimal choice of the number of clusters may be automatically identified by the use of the PLIC index.

In **Chapter 6**, two strategies (Strategy I and Strategy II) to mixture model clustering for multivariate images have been developed. Strategy I is for a image data where each cluster is normally distributed, and the other for the situation where a cluster is a mixture of several Gaussian distributions. The methods minimize the need for human interaction to select values of input parameters. Spatial information is effectively used. Firstly, in the first stages it considers only representative regions, formed by RGS, which not only makes the process faster, but allows for reliable estimation of cluster parameters. Secondly, by employing MRF classification in the final step, it reduces the effect of noise/artifacts and the overlap problem of clusters, often present in real-world data sets. Especially with Strategy II, Gaussian sub-clusters of one component and other very small clusters are very well recognized. The proposed strategies gave very good performance on both real world data sets with difference types of problems.

SAMENVATTING

In veel beeldverwerkingstoepassingen worden enorme data sets gegenereerd, die onmogelijk handmatig verwerkt kunnen worden. Een voorbeeld is remote sensing. Automatische analysetechnieken, zoals clustering, maken het mogelijk grotere en grotere data sets te analyseren. De meeste clustermethoden maken echter alleen gebruik van de spectrale informatie van de beelden. Het doel van dit proefschrift is het bestuderen van een uitbreiding van clustertechnieken voor grote multivariate/multispectrale beelden die de voordelen van de ruimtelijke informatie kan benutten. Het belangrijkste punt is de verbetering van de robuustheid van de clustermethoden (met betrekking tot invoerparameters en het aantal klassen), en de accuratesse, door de invloed van problemen zoals cluster overlap en ruis in (onder andere) remote sensing beelden te verminderen.

Hoofdstuk 2 bestaat uit een tutorial, een breed overzicht van de basale clustermethoden voor multivariate beelden, en verschaft handreikingen om de meest toepasselijke clustering voor een specifieke data set van multivariate beelden te bepalen, afhankelijk van "image data moeilijkheden". In veel gevallen vormen partitionele clustermethodes die gebruik maken van ruimtelijke informatie de beste optie voor een groot beeld omdat ze beter in staat zijn om te gaan met ruis en uitbijters. Het tutorial laat ook een niet-opgelost probleem zien voor het clusteren van multivariate beelden, afhankelijk van een goede keuze van invoerparameters. Automatische procedures geven niet altijd een goed resultaat. In veel gevallen kan de keuze worden bepaald door een "trial and error" strategie en persoonlijke ervaring. Dit is echter moeilijker voor een groot beeld, waar meer dan een set parameters nodig kan zijn. Bovendien moet bij het clusteren van multivariate beelden rekening gehouden worden met de grootte van de data set, door de voort durende ontwikkeling van beeldvormende technologie.

Hoofdstuk 3 presenteert een nieuw clustering algoritme (KNNCLUST) dat om kan gaan met het voor dichtheids-gebaseerde clustermethoden bekende probleem van clusters met verschillende dichtheden in een hoogdimensionale ruimte. KNNCLUST is een zeer goede methode om relatief kleine multivariate data te clusteren, vooropgezet dat de clusters niet al te veel van grootte verschillen. De KNN-kernel dichtheidsschatter met driehoekige en Gaussische kernels wordt gebruikt door KNNCLUST. KNNCLUST gebruikt slechts een parameter, het aantal buren k. In de meeste gevallen is het eenvoudig om een aantal waardes van k te vinden waarbij de clusterresultaten stabiel zijn.Het aantal clusters wordt automatisch door KNNCLUST bepaald. Vanwege de berekening van de KNN afstandsmatrix is de computationele complexiteit van het algoritme vrij groot. In de praktijk echter kunnen indexeringstechnieken worden gebruikt om de berekeningen te

versnellen. KNNCLUST is geschikt voor hoog-dimensionale data sets.

Vanaf **hoofdstuk 4** wordt het gebruik van ruimtelijke informatie in de clustering bestudeerd. Ruimtelijke informatie is al vaker gebruikt in de voor- en nabewerking, wat refereert naar respectievelijk het filteren en gladstrijken van een beeld met ruis en artefacten, en het clusterresultaat. In dit proefschrift wordt ruimtelijke informatie in

verschillende stadia gebruikt; het initializeren van de clustering parameters, gedurende het clusteren of het filteren van het clusterresultaat in het laatste stadium, gebruik makend van

ruimtelijke informatie samen met de cluster structuur.

Hoofdstuk 4 introduceert SpaRef als een clustering algoritme voor hyperspectraalbeelden, het eerste algoritme in dit proefschrift dat de ruimtelijke informatie meeneemt. Deze methode is een combinatie van K-means en Ward's clustering, waarbij de Ward's clustering (een agglomeratieve hierarchische clustering) wordt toegepast op een niet al te groot aantal zeer homogene klassen, verkregen door K-means (een partitionele clusteringmethode). Uiteindelijk wordt ruimtelijke informatie gebruikt om de potentiele tekortkomingen van Ward's methode te corrigeren door een verfijning toe te passen. De voorgestelde methode heeft als voordeel stabiel te zijn en te leiden tot clusters met een hoge graad van compactheid en continuiteit.

Hoofdstuk 5 onderzoekt een andere mogelijkheid om ruimtelijke informatie te gebruiken voor de initialisatie van Markov Random Field (MRF) clustering. De clustering is dan robuuster en toepasbaar voor grote remote sensing beelden die veel grote homogene gebieden bevatten, zoals landbouwgebieden. Kleine en geisoleerde clusters kunnen gemist worden door de methode. Een optimale keuze voor het aantal clusters kan automatisch gemaakt door gebruik te maken van de PLIC index.

In **Hoofdstuk 6** worden twee strategieen (I en II) ontwikkeld voor mixture modelling van multivariate beelden. Strategie I is voor data waar elk cluster normaal verdeeld is, en II voor de situatie waar een cluster bestaat uit een mengsel van normaalverdelingen. De methoden minimaliseren de noodzaak aan menselijke interactie om invoerparameters te kiezen. Ruimtelijke informatie wordt efficiënt benut. In de eerste stadia worden alleen representatieve gebieden beschouwd, verkregen door RGS, hetgeen niet alleen het proces versnelt maar ook een betrouwbare schatting van cluster parameters opleverd. Door MRF klassificatie in de laatste stap worden de in realistische data sets vaak aanwezige effecten van ruis, artefacten en cluster overlap verminderd. Gaussische subclusters van 1 component en andere zeer kleine clusters worden vooral met strategie II erg goed herkend. De voorgestelde strategieën doen het erg goed op een tweetal realistische data sets met verschillende moeilijkheden.

ACKNOWLEDGEMENT

I would like to take this opportunity to thank all from the university who made my stay here a productive and enjoyable experience.

My deepest thanks go to my supervisors, Lutgarde M.C. Buydens and Ron Wehrens for all their support, guidance, and friendship during my years at the department. I also highly appreciate for the freedom they gave me in constructing this research.

I am very grateful to many people who have collaborated with me, in particular to Dirk Hoekman, Geerjan Geerling, Jacco C. Noordam and Rob Jordans for their valuable conversations, comments and helps on sharing datasets.

Special thanks to the department secretary, Brigitte Loozen, for her help on handling many paper works and documents in Dutch.

Thanks to many other colleagues in the department for creating such a nice atmosphere to work in the department.

My parents and close relative deserve very special thanks since they have always encouraged me on my study. Many thanks to my brother and his family for taking such good care of my parents when I in abroad. Last, but certainly not least, I thank my wife for her patience and support throughout.

Nijmegen, August 2005

Curriculum Vitae

Thanh Ngoc Tran was born at 27 July 1973 at Hanoi, Vietnam. After receiving his bachelor's degree in mathematics and informatics at Thanglong University in Vietnam in 1994, he started a 5 years working contract for Asian Institute of Technology (AIT). During that time, he had been pursuing master's degree in computer science and information management at the same institute. After graduated in 1999, he worked for International Telecommunication Union (ITU), United Nation regional office in Thailand and Technical University of Eindhoven (TUe) (the Netherland). September 2001, he started Ph.D degree in image analysis from the Analytical Chemistry Department in Nijmegen University, The Netherlands, specializing in clustering algorithms for multi-spectral data/images application (but not limited) to remote sensing, satellite images.

He refereed papers for several international journals: Analytica Chimica Acta, Information Science, Journals of Remote Sensing and IEEE transaction on Geoscience and Remote Sensing.