

# Knn Density-Based Clustering for High Dimensional Multispectral Images

Thanh N. Tran, *member, IEEE*, Ron Wehrens, Lutgarde M.C. Buydens\*  
 Laboratory of Analytical Chemistry, University of Nijmegen  
 Toernooiveld 1, 6525 ED Nijmegen, The Netherlands  
 Email: {tnthanh, rwehrens, lbuydens}@sci.kun.nl

**Abstract**— High resolution and high dimension satellite images cause problems for clustering methods due to clusters of different sizes, shapes and densities. The most common clustering methods, e.g. K-means and ISODATA, do not work well for such kinds of datasets. In this work, density estimation techniques and density-based clustering methods are exploited. Density-based clustering is well-known in data mining to classify a data set based on its density parameters, where high density areas are separated by lower density areas, although it can only work with a simple data set in which cluster densities are not very different.

Our contribution is to propose the k nearest neighbor (knn) density-based rule for a high dimensional dataset and to develop a new knn density-based clustering (KNNCLUST) for such complex dataset. KNNCLUST is stable, clear and easy to understand and implement. The number of clusters is automatically determined. These properties are illustrated by the segmentation of a multispectral image of a floodplain in The Netherlands.

**Index Terms**—Clustering algorithm, density-estimation, high dimension multispectral images.

## I. INTRODUCTION

CLUSTERING is the organization of a data set into homogeneous 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 to points in different clusters. 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. Due to the technological advancement, a huge amount of data from satellite imagery is available for information processing with very high resolution in both spatial (i.e. the number of pixels) and spectral (i.e. the number of measurement values per pixel) domains. In satellite images for urban areas, a complex spatial assemblage of disparate land cover types comprises built structures, numerous vegetation types, bare soil and water bodies, which have different spectral reflectance characteristics, leading to cluster of widely different shapes, sizes and densities in the feature domain.

Clustering methods basically fall into two types: partitional and hierarchical approaches [1]. Variants of K-clustering, such as K-means and ISODATA [2] are the partitional clustering

methods that are most widely used for satellite images because they are computationally attractive. However, they are not very stable and very sensitive to outliers and they often do not work well when the clusters are of different size, shape, and density [3]. Therefore, results on satellite images are not very good. In contrast, agglomerative hierarchical clustering (AHC) is more stable but its computation and computer memory used are very expensive and, thus, it is not feasible for a large data set. Moreover, for examples, the single link approaches are very vulnerable to noise and differences in density. While group average and complete link are not as vulnerable to noise, they have trouble with varying densities and cannot handle clusters of different shapes and sizes [3].

Besides the partitional and hierarchical approaches, density-based clustering methods such as Denclust [4] and DBSCAN [5] form a third clustering type. These are often used in data mining for knowledge discovery. 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 remain separated from one another by low-density regions. Density-based clustering has advantages over K-clustering and AHC in discovering clusters of arbitrary shapes and sizes. However, it was shown that current density-based clustering works well only on a simple data set where cluster densities are similar [6].

Density estimation techniques range from histogram-based to kernel-based approaches. They become more difficult for high dimension data sets, in which the volume of data space increases dramatically [3]. In this paper, a new k nearest neighbor (knn) density rule for a high dimension multispectral data set is presented. The clustering method based on this knn density-rule is proposed called KNNCLUST. This knn density-based clustering shows advantages of working with clusters of different sizes, shapes and densities. And it is stable and easy to understand and implement. Moreover, the number of clusters is determined automatically.

The paper is organized as follows. First, we review local density estimation techniques and density-based clustering algorithms in section II. The description of the knn local density rule for high dimensional data sets and the density-based clustering KNNCLUST are given in section III. In section IV, some experimental results are presented. Finally, the conclusions are in section V.

## II. LOCAL DENSITY ESTIMATION AND DENSITY-BASED CLUSTERING

One of the simplest nonparametric density estimation methods is the histogram-based approximation of an unknown probability density function (pdf) [7]

$$pdf(x) = \frac{k_N}{h \cdot N} \quad (1)$$

where  $N$  is the total number of points,  $h$  is the length of histogram's bin and  $k_N$  is the number of points located inside the bin. There are variations of histogram-based approximation techniques that use search windows or grids instead of bins, where the volumes of search windows or grids are used instead of the bin length  $h$ . However, the approximation of the function pdf is a discontinuous function.

The density function can be made smoother by introducing a weight to each point, so that the points at the edge of the search area are less influenced to the estimator than the other points. A number of different functions are used to perform the weight calculation, given the generic name kernels or potential functions [8]. A triangular or a Gaussian kernel function is normally used,

$$pdf(x) = \frac{1}{V} \sum_{i \in D} \phi(x_i - x) \quad (2)$$

where  $D$  is the data search space,  $V$  is the volume of  $D$  and the Gaussian density function  $\phi$  is given by

$$\phi(x_i - x) = e^{-\frac{|x_i - x|^2}{\sigma^2}} \quad (3)$$

In the histogram based approach or in the kernels approach, the volume around point  $x$  was fixed. In contrast, in the  $K$ -nearest neighbor (knn) density estimation [8] the number of points  $k$  is fixed and the size of the volume around point  $x$  is adjusted to include  $k$  nearest neighbor points. The volume, therefore, is much smaller in areas of high density than in areas of low density.

Density estimation techniques have been used in many supervised classification methods, such as Bayesian classifiers and ALLOC [8],  $k$ -NN classifiers, etc [7]. In supervised classification, training points for each class are provided and the density values of given classes are calculated. The classification of a new object  $u$  is determined by the highest value of the density function  $pdf$  that estimates for all classes at the position of  $u$ .

By contrast, unsupervised classification is used when training objects are not available. A density map is produced by the local density values for every point. Histogram-based density estimation has been used in DBSCAN [5], while kernel density estimation has been used in Denclust [4]. A density-based clustering algorithm is basically a hill-climbing procedure to group attractor points (in Denclust) or core points (in DBSCAN), whose density values are higher than the minimum density threshold  $\mathcal{E}$ .

Research by M. Ankerst et al. (1999) showed that the global density threshold  $\mathcal{E}$  does not exist in a complex data set, in which local densities are very different [6]. Many solutions have been introduced such as in references [6, 9, 10], which

tried to identify  $m$  different density thresholds  $\mathcal{E}_{i=1..m}$  for different density distributions. The algorithms simply group all attractors or core points for different density threshold  $\mathcal{E}_{i=1..m}$  individually, which is similarly to a hierarchical application of Denclust or DBSCAN algorithms with the threshold  $\mathcal{E}_{i=1..m}$ . However, these solutions work well only when density distributions are well separated. In other cases, the algorithm tends to create more clusters due to overlapping densities.

These density estimation techniques become more difficult for high dimension data sets, in which the volume of data space increases dramatically when dimensionality increases but the number of points remains the same. L. Ertoz, et al. (2002) provided a solution by using a knn density estimation technique for clustering in which the local densities of adjacent points are compared by the gradient of density functions, such as the number of shared neighbor points in this case [3]. A more shared neighbor points leads to a lower gradient. Therefore, the density gradient does not have to take into account the volume of the search space. However, this clustering method still requires the threshold of the gradient to be defined [10].

## III. KNN DENSITY-BASED CLUSTERING

### A. KNN density-based rule for high dimension data set

Given a dataset  $D = [x_1, \dots, x_N]$  and the set of clusters presented in the dataset  $C_{i=[1..C]}$ , the knn query of the point  $x$  is the set of the nearest neighbors of the point  $x$   $K_m(x) = \{x_j\}_{j=[1..k]}$ . If the data volume defined by the query is  $V_k(x)$ , the density of the cluster  $c_i$  at the point  $x$  is defined by the ratio of the number of the  $k$  nearest neighbor points in  $c_i$  to the volume  $V_k(x)$ . [7]

$$f(x, c_i) = \frac{size(K_m(x) \cap C_i)}{V_k(x)} \quad (4)$$

Suppose that point  $x$  is a new point which is not yet assigned to any cluster. If cluster memberships for all points in  $Knn(x)$  are known, then  $x$  can be assigned to cluster  $c$  in which

$$f(x, c) = \underset{i=[1..C]}{Max}(f(x, c_i)) \quad (5)$$

The volume  $V_k(x)$  for the query  $Knn(x)$  is the same for all  $f(x, c_i)$  and therefore the point  $x$  is assigned to the cluster  $c$  where  $Knn\_rule$  is hold:

$Knn\_rule$ :

$$1. \ size(K_m(x) \cap C_c) = \underset{i=[1..C]}{Max}(size(K_m(x) \cap C_i)) \quad (6)$$

2. *If there are more than one clusters containing the same max points, then the point  $x$  has to be assigned to the closest cluster  $c$ , which is defined by the nearest distance of the furthest point in the cluster to the point  $x$ , which is called the 'complete rule'.*

Although the 1<sup>st</sup> rule of  $knn\_rule$  (eq. 6) is not new, since it is used in the  $Knn$  supervised classifier, we would like to build

a clear link between *knn\_rule* and density estimation techniques and to extend it to unsupervised classification methods.

The 2<sup>nd</sup> rule of *knn\_rule* actually is the merging criterion in the complete-linkage clustering merging the point *x* to the set of clusters. This rule can be extended to the min distance, which is equivalent to the criterion of single-linkage clustering, or the mean distance, equivalent to the criterion of average-linkage clustering.

Interesting is that *knn\_rule* does not depend on the dimension of the dataset. Our goal of using the density estimation technique is not to estimate a ‘real’ distribution of the data set, as attempted by kernel techniques, but to assign with this rule the data object to the ‘right’ group of the data set.

*B. Knn density-based clustering (KNNCLUST)*

In this section, the methodology of the *knn* density-based clustering called KNNCLUST is described. The algorithm works as follows.

First, KNNCLUST starts by assigning each point to individual cluster. The algorithm requires the number of nearest neighbors –*k*–, which is the only parameter of the algorithm. Then, the *knn* query – $K_{nn}(x)$ – is calculated for all points.

At each iterative step, the *k*-nearest-neighbor rule – *knn\_rule* - is applied for every point *x* which is assigned to the nearest obtained cluster *c*. Each point has a chance to be assigned or to be re-assigned to the most suitable cluster at this procedure. This step is repeated until convergence, when no or only few points are changing from one cluster to another. The higher convergence rate, the more accurate result obtains and time needs for the algorithm, and vice visa. The flowchart of KNNCLUST is given in Fig. 1.

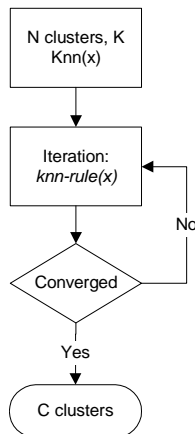


Fig.1. KNNCLUST flowchart.

In KNNCLUST, the number of clusters is automatically identified upon convergence. The *Knn* query is limited only to ‘neighbor’ clusters to reduce the size of the *knn* distance table and allows the algorithm to work with a large data set. KNNCLUST uses the *knn\_rule* instead of ‘traditional’

distances in other methods without any additional density constraint (e.i. a density threshold  $\mathcal{E}$  in Denclust and DBSCAN), and thus this method can deal well with clusters of different sizes, shapes and densities. In general, noise and outliers must be removed before clustering. If they are still present in the dataset, they will not influence the clustering result and will be joined to the closest cluster by *knn\_rule*.

KNNCLUST requires only one parameter –*k*– which should be smaller than the smallest size expected for any cluster. The smaller *k*, the more detail there is in the clustering and the more clusters can be obtained. By contrast, with higher values for *k*, the clustering result is ‘smoother’ and a smaller number of clusters is obtained.

The complexity of KNNCLUST depends mainly on the calculation of the  $Knn(x_{i=1..N})$  query which is very expensive,  $O(N.N.log(N))$  for an image of *N* data points.

We are currently studying the potential of using spatial information, i.e. the location of an object in the image, to define a search space for a *knn* query for a point *x*, instead of the whole image. By using that, the complexity of the queries is dramatically reduced to  $O(N.w.log(w))$  where *w* is the size of the search space in the spatial domain.

IV. RESULTS

As an example, we will use a multispectral satellite image recorded by a Compact Airborne Spectrographic Imager (CASI) scanner from the Natural Environment Research Council (NERC) that was taken at 1536 m over an area in the Klompenwaard, the Netherlands during August 2001. The CASI has provided 10 bands for this study from 437 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 90 x 765 m<sup>2</sup>. The original multispectral data were mean centered and compressed via a principal components analysis to the first four principal components, which account for more than 99.8 % of the spectral variance, in order to reduce computation time [8].

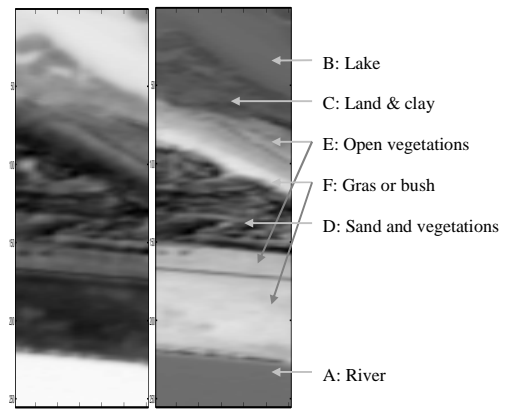


Fig. 2. Gray-scale image of PC1 and PC2.

Fig. 2 shows the gray-scale images of the first two principal components. Basically, six main object patterns have been defined: a lake, grass or bush, etc.

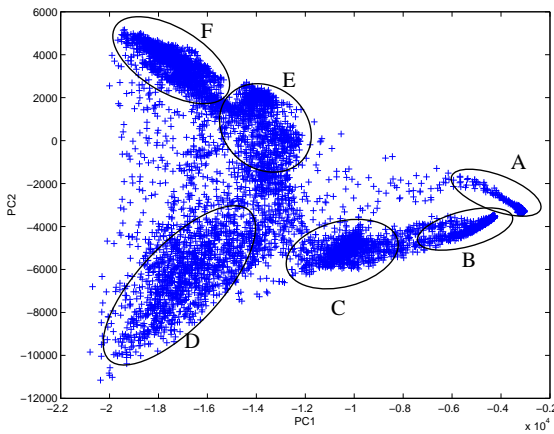


Fig. 3. Score plot of PC1 and PC2.

Fig. 3 clearly shows that clusters are different in shapes, sizes, and densities; e.g. cluster A and B are very small with a long shape containing approximately 1000 points, which is similar to the other cluster sizes.

Six clusters were found by KNNCLUST. This was the case for all values of  $k$  that were tested: [450, 500, 550, 600, 650, 700]. For the comparison with  $k$ -means, this method was initiated by six random cluster centers. The results of the KNNCLUST image looks much smoother, mainly because the open vegetation area, E, is more homogeneous.  $K$ -means incorrectly joins the lake (B) and the river (A), and divides E into two clusters.

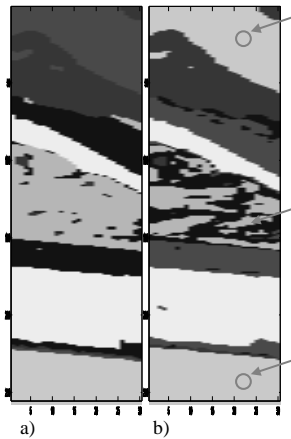


Fig. 4. Six clusters obtained by a) KNNCLUST with  $k=550$ , b)  $K$ -means with random seeds to six clusters.

Just like with hierarchical clusterings, the order of the points may influence the clustering results of KNNCLUST. The data points are randomly re-ordered in order to test this behavior in KNNCLUST for all values of  $k$ : [300, 350, 400, 450, 500, 550, 600, 650, 700]. The Rand index [11] was used to compare the clustering result before and after the re-ordering procedure. This index is based on counting the pairs of points on which the two clustering agree/disagree. Perfect agreement corresponds with an index of 1; the lower bound of the index

is zero. The obtained Rand indices were higher than 0.96 for the cases of the same number of clusters, i.e. six or seven. It indicates that KNNCLUST is not very sensitive to the point order for this data set.

For the purpose of qualitative comparison,  $I$  index has been considered, which measures the ratio of within-cluster variation and between-cluster variation [12]. Lower  $I$  index values indicate better results. This index is not designed for a data set with clusters of different shapes such as in this data set, but it might provide an idea about the stability and the compactness of clustering results.

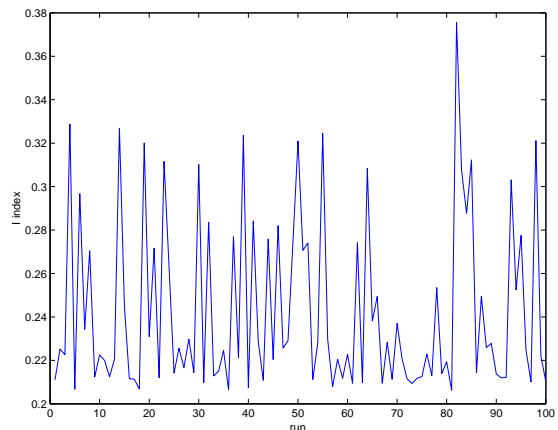


Fig. 5.  $I$  index changes of  $K$ -means in 100 runs.

Fig. 5 indicates the ' $I$ ' index values of  $K$ -means for six clusters in 100 runs with random initial cluster position. It clearly shows that  $K$ -means is not stable, where the minimum value of the ' $I$ ' index is 0.2063 and the maximum value is 0.3756. On the other hand, the minimum value of the ' $I$ ' index for KNNCLUST is 0.2074 when the  $k$  value is 700, and the maximum value is 0.2081 when the  $k$  value is 450, which are comparable to the best case obtaining by  $K$ -means. The small variance of the ' $I$ ' index indicates that KNNCLUST is not sensitive to the values of  $k$ .

### V. CONCLUSION

This paper has presented a new  $knn$  density-based rule - the method of comparing densities - for a high dimensional data set. A new  $knn$  density-based clustering called KNNCLUST has developed based on this rule. The  $knn\_rule$  makes a clear link between density-estimation techniques to unsupervised classification. The method can obtain clusters different in size, shape and density. The algorithm is clear and easy to understand and implement. The method has the advantage of being stable and needs only a minimum of user input. Besides, the number of clusters is automatically determined. The complexity of this 'basic' KNNCLUST to the multispectral images is still very high. Our future research focuses on integrating spatial information of the image to improve the performance in huge multi-spectral images, e.g. satellite images. KNNCLUST has successfully experimented on the multi-spectral CASI image.

## ACKNOWLEDGMENT

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

## REFERENCES

- [1] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review", *ACM Computing Surveys*, vol. 31, no 3, pp. 264-323, Sep. 1999.
- [2] G.H. Ball, D.J. Hall, ISODATA, "A novel method of data analysis and pattern classification", Springfield, Stanford, 1965.
- [3] L. Ertoz, M. Steinbach and V. Kumar, "A new shared nearest neighbor clustering algorithm and its applications", AHPCCRC, Tech. Rep. 134, 2002.
- [4] 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.
- [5] 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.
- [6] M. Ankerst, M. M. Breunig and H-P Kriegel, "OPTICS: Ordering Points to Identify the Clustering Structure". in Proc. *ACM SIGMOD*, 1999, pp. 49-60.
- [7] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 1999.
- [8] B.G. M. Vandeginste, D.L. Massart, L.M.C. Buydens, S.De Jong, P.J. Lewi and J. Smeyers-Verbeke, *Handbook of Chemometrics and Qualimetrics, Part B*. Elsevier, 1998, pp. 225-227.
- [9] 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, issue 2, pp. 151-167, 2002.
- [10] J. A. Richards, X. Jia, *Remote Sensing Digital Image Analysis*. Springer, 1999.
- [11] L. Hubert and P. Arabie, "Comparing Partitions", *Journal of Classification*, vol. 2, pp. 193-218, 1985.
- [12] Brereton R. G., "Multivariate pattern recognition in chemometrics, illustrated by case studies", Elsevier, 1992, pp. 179-204.