Advances in the segmentation of multi-component microanalytical images

Jérôme Cutrona\textsuperscript{1,2}, Noël Bonnet\textsuperscript{1,2}, Michel Herbin\textsuperscript{2}, Ferdinand Höfer\textsuperscript{3}

\textsuperscript{1}UMRS INSERM 514 (IFR53). Hôpital Maison Blanche. 45, rue Cognacq Jay. 51092 REIMS Cedex. France.
\textsuperscript{2}Laboratoire d’Etudes et de Recherches en Informatique (CReSTIC-LERI). University of Reims. France.
\textsuperscript{3}Research Institute for Electron Microscopy, Graz University of Technology, Steyrergasse 17, A-8010 GRAZ. Austria.

Abstract
Considering the problem of segmenting multi-component microanalytical images through pixel clustering, we first highlight some limitations of classical clustering algorithms (C-means and fuzzy C-means). Then, we describe a new algorithm we have contributed to develop: the Parzen-watersheds algorithm. This algorithm is based on the estimation of the probability density function of the whole data set in the feature space (through the Parzen approach) and its partitioning using a method inherited from mathematical morphology: the watersheds method. Next, we introduce a fuzzy version of this approach, where the pixels are characterized by their grades of membership to the different classes. Finally, we show how the definition of the grades of membership can be used to improve the results of clustering, through probabilistic relaxation in the image space. The different methods presented are illustrated through an example in the field of electron energy loss mapping, where four elemental maps are concentrated in a single chemical phase map.

P.A.C.S: 02.70.-c  02.50.Sk  68.55.Nq  42.30.Va  07.05.P

Keywords: multi-component images; segmentation; clustering; relaxation; probability density function; fuzzy logic.

1. Introduction
The improvement of microanalytical techniques allows us to record, simultaneously or not, the maps of several elements constituting the specimen under study. The observation of these different maps permits us to interpret the content of the studied area of the specimen qualitatively. But quantitative information remains difficult to infer. One step towards a more quantitative interpretation will consist in concentrating the information contained in the different elemental maps into a single labelled map, where each label is associated with a
Each class of pixels is characterized by a set of concentrations, one for each of the studied elements.

There are several ways to go from the set of elemental maps to the unique class map. A first group of methods is interactive and sometimes called Interactive Correlation Partitioning. It consists in selecting the different classes of pixels interactively, within the two- or three-dimensional scatterplot, and back-mapping the selected areas into the real image space.

The second group of methods, on which we will concentrate here, is automatic and we called it Automatic Correlation Partitioning. It consists in automatically grouping similar pixels (that possess similar elemental composition) into one class, the number of classes being a priori unknown. This process is known as clustering in the artificial intelligence and data analysis communities [1].

To the best of our knowledge, the first attempt to introduce the idea of clustering in the framework of imaging in microanalysis is in [2]. In this work, an algorithm well-known in data processing was introduced: the K-means algorithm. A little bit later, Bonnet et al. [3] expanded on this idea and presented more detailed applications of the K-means and the fuzzy C-means algorithms, a fuzzy version of the K-means algorithm.²

In parallel to their application to microanalytical data sets, clustering techniques are also evolving independently and we contributed to the development of a new technique that overcomes some drawbacks of the previous ones. In this paper, we illustrate the capabilities of the new technique and its fuzzy version in the framework of multi-component microanalytical imaging.

The outline of the paper is the following. In the next section, we summarize the C-means and fuzzy C-means algorithms. We also emphasize their weaknesses. In the following section, we present the new algorithm we have developed (and called the Parzen-watersheds algorithm) and its fuzzy counterpart. We show how these algorithms surpass the classical algorithms. Then, we illustrate the application of the different algorithms to a set of four elemental maps recorded through electron energy-loss mapping. Finally, we draw some conclusions.

---

¹ Or voxels for three-dimensional microanalytical techniques.
² The K in “K-means” and the C in “fuzzy C-means” have exactly the same meaning, i.e. the number of classes. So, in the remaining of the paper, we will use “C-means” and “fuzzy C-means”, in order to avoid confusion.
2. Classical clustering algorithms: C-means and fuzzy C-means.

We start with a set of N elemental maps $EM$, each one of size $K = width \times height$:

$$ EM_n(k), \quad n \in \{1, \ldots, N\}, k \in \{1, \ldots, K\} $$

(1)

Each pixel $k$ of these maps can thus be described by a vector $V_k \in \mathbb{R}^N$:

$$ V_k = [EM_1(k), \ldots, EM_n(k), \ldots, EM_N(k)]^T, \quad k \in \{1, \ldots, K\} $$

(2)

The purpose of the clustering process is to group pixels into a limited set of classes: $c \in \{1, \ldots, C\}$. This must be done on the basis of the data set alone, without the help of any additional training set.3

The classical C-means algorithm aims at grouping objects (here, objects are pixels) in order to minimize the objective function:

$$ J_{C\text{-means}} = \sum_{c=1}^{C} \sum_{k=1}^{K} d^2(V_k, X_c) $$

where $X_c$ is the centre of class $c$ and $d$ is the distance between object $k$ and $X_c$.

The algorithm itself was fully described in [2] and is not reproduced here.

One of the problems with this algorithm, as with many clustering algorithms, is that the number of classes, $C$, must be fixed in advance. When this parameter is not known, one has to run the algorithm several times with a varying number of classes and to deduce the true number of classes from the minimization/maximization of a validity criterion, or figure of merit. In [3], several validity criteria were investigated.

Along the iterations of the C-means algorithm, each pixel is associated to one class in a binary (yes/no) way. However, in the community of data processing and artificial intelligence, it was found through several arguments that fuzzy grades of membership are a better alternative: an object belongs simultaneously to the different classes [4]:

$$ \begin{cases} 
0 \leq \mu_{k,c} \leq 1 \\
\sum_{c=1}^{C} \mu_{k,c} = 1 
\end{cases} $$

(4)

where $\mu_{k,c}$ is the grade of membership of object $k$ to class $c$.

The fuzzy C-means algorithm is the extension of the (hard) C-means algorithm in the framework of fuzzy logic. Again, an objective function has to be minimized iteratively. This function is:

3 Other classification algorithms are based on the use of a training set. We do not consider them in this paper.
\[ J_{\text{Fuzzy C-means}} = \sum_{c=1}^{C} \sum_{k=1}^{K} \mu_{k,c}^m d^2(V_k, X_c) \]  (5)

where \( m \) is a fuzzy parameter, generally taken as \( m = 2 \).

The minimization can be obtained through an iterative alternate process (as for the C-means algorithm):

- given the centres of the classes, \( X_c \), compute the grades of membership of the pixels to these classes:

\[ \mu_{k,c} = \frac{\left( d^2(V_k, X_c) \right)^{m-1}}{\sum_{c'=1}^{C} \left( d^2(V_k, X_{c'}) \right)^{m-1}} \]  (6)

- given the grades of membership, compute the fuzzy centres of the classes:

\[ X_c = \frac{\sum_{k=1}^{K} \mu_{k,c}^m V_k}{\sum_{k=1}^{K} \mu_{k,c}^m} \]  (7)

The iterative process stops when convergence is attained. Then, defuzzification has to take place in order to attach class labels to the different pixels:

\[ \text{object } k \rightarrow \text{ class } c, \quad c = \text{Arg Max}_{c} \left\{ \mu_{k,c} \right\} \]

As for the C-means algorithm, validity criteria have to be defined in order to estimate the true number of classes. Many such figures of merit have been devised for the fuzzy C-means algorithm (Cf. [3] and references therein).

**Main limitation of the C-means and fuzzy C-means algorithms:**

Both algorithms rely on the computation of distances between an object and the centre of a class. Whatever the type of distance chosen, this implies that some assumptions are made on the shape of the classes, represented by clusters in \( \mathbb{R}^N \). If we consider, for instance, the classical Euclidean distance, the implicit assumption is that clusters are hyper-spherical and have similar extension. Figure 1 exemplifies the failure of clustering when these assumptions are not fulfilled. Figure 1a shows a case with two hyper-spherical clusters (two disks here, in a two-dimensional feature space) with similar extension: the Euclidean distance allows solving the partitioning problem correctly. Figure 1b shows the case of an hyper-spherical cluster and an hyper-elliptical cluster. Figure 1c shows the case of two hyper-spherical clusters with a dissimilar extension. Figure 1d shows the case of non convex clusters. For
these last three cases, the Euclidean distance does not permit to get the correct clusters: the boundaries found by classical clustering algorithms are represented by the vertical bars. It is fair to say that more sophisticated distances, such as the Mahalanobis distance [5] for instance, could in principle cope with cases b and c of figure 1, since the underlying assumption is then that of hyper-elliptical clusters. But with such distances, the minimization process is more sensitive to the initialisation parameters, has more difficulties to converge and often converges towards local minima instead of the global minimum. In addition, case d of figure 1 cannot be handled by such algorithms.

3. The Parzen-watersheds algorithm and its fuzzy version

With the aim of providing a clustering algorithm that does not make assumptions concerning the shape and extension of clusters, we have developed the Parzen-watersheds clustering algorithm [6,7,1].

The method first consists in estimating the probability density function (pdf) of the whole data set in the feature space $\mathbb{R}^N$. This can be done according to the Parzen method [8]: the point distribution (one object corresponds to one point in $\mathbb{R}^N$) is transformed into a quasi-continuous distribution through convolution by a smoothing kernel:

$$pdf(\vec{z}) \approx \frac{1}{h^N} \sum_{k=1}^{K} \text{Ker} \left( \frac{\vec{z} - \vec{z}_k}{h} \right), \vec{z} \in \mathbb{R}^N$$

(8)

where Ker is a smoothing kernel of size $h$.

The estimated pdf is generally characterized by the presence of several modes, or local maxima, separated by valleys, or local minima. We consider that each of the modes corresponds to a class of objects. Thus, the following step consists in estimating the positions of the boundaries between the different classes, in the feature space $\mathbb{R}^N$. Although this task is trivial for $N = 1$, it is much less trivial for a N-dimensional feature space, i.e. $N > 1$.

Our approach has been to use a procedure originating from image processing: the watersheds function of mathematical morphology [9]. This procedure allows the catchment basins associated to each mode of the pdf to be defined. The watersheds, where the catchment basins meet, constitute the boundaries between the different classes.

---

4 In supervised classification procedures, the probability density function is evaluated for each training class separately. Here, all classes are mixed and the pdf concerns the mixture of all classes.

5 We chose Gaussian kernels.

6 Our implementation is limited to $\mathbb{R}^3$. 
Once the different classes, and the boundaries which separate them, are defined in the feature space $\mathbb{R}^N$, the last step consists in going backwards the real image space. This can be done easily because one knows where any pixel $k$ was mapped in the feature space. Thus it suffices to carry back the label $c$ found in the feature space in the real space.

As for any clustering method, the question of selecting a number of classes is relevant. Within our framework, the number of classes is the number of modes of the estimated pdf. The number of modes itself is related to the extension (width) of the smoothing kernel, $h$. Our approach to estimate the number of classes, $C$, consists in looking at the number of modes, $M$, as a function of the parameter $h$. The decreasing curve $M = f(h)$ displays some plateaus, which give stable solutions for the number of modes, and hence of classes.\(^7\) In [10], we also investigated another approach, based on the concepts of stability of the number of modes when resampling the data set, using the bootstrap technique [11].

**Extension to a fuzzy approach:**

For the same reasons as the C-means algorithm could be improved through fuzzy logic concepts, we expected that our Parzen-watersheds algorithm could be improved with the replacement of hard labels by grades of memberships [12].

The first step of the fuzzy procedure is the same as previously: the estimation of the global pdf is performed with the Parzen technique. But, instead of computing the zones of influence of the different modes (via the watershed technique), each point $p$ of the feature space receives a grade of membership to the different classes. For this, the cost of the travel from the considered point $p$ to each mode $m_c$ is computed as $\text{cost}(p,m_c)$:

$$\text{cost}(p,m_c) = \sum_{p',p''\in S_{p,m_c}} d_{m_c}(p',p'')$$

where $d_{m_c}(p',p'') = \frac{|pdf(p') - pdf(p'')|}{pdf(m_c)}$

and $S_{p,m_c}$ is the less costly path from $p$ to $m_c$ and $p'$, $p''$ are two neighbour points of the feature space along this path.

\(^7\) It should be stressed that there is no unique solution to a clustering problem. There is a hierarchy of possibilities, corresponding to different scales at which the data set can be handled. The different plateaus correspond to these different possibilities, from which the user has to make his personal choice.
Then, the grade of membership of point \( p \) to class \( c \) (corresponding to mode \( m_c \)) can be computed as:

\[
\mu_{p,c} = \frac{\left( \text{cost}(p, m_c) \right)^{-1}}{\sum_{c' = 1}^{C} \left( \text{cost}(p, m_{c'}) \right)^{-1}}
\]

(11)

The costs of the travels to the different modes are computed for all the points in the feature space, according to Dijkstra’s algorithm [13].

Illustration of the interest of the fuzzy version compared to the non-fuzzy version: probabilistic relaxation.

With the non-fuzzy Parzen-watersheds algorithm, the final result is obtained in the form of a labelled map where each label is the number of the class to which any pixel belongs. This label is obtained as a result of the clustering process in the feature space. Of course, as any clustering procedure, the one we employ is subject to classification errors, even if it was designed in order to minimize the number of classification errors, compared to classical methods.

The fuzzy version can help to reduce the classification errors a little bit more, along the following way. After the computation of the grades of membership in the feature space, we do not convert these grades of membership into labels, through defuzzyfication. Instead, we transfer these grades of membership to the pixels, in the real (image) space. We thus get as many maps as we have selected classes, i.e. \( C: \mu_{k,c}, \quad c \in \{1, \cdots, C\}, k \in \{1, \cdots, K\} \).

Now, one possibility to correct (at least partly) the errors made at the clustering level is to take into account the fact that the probability that neighbouring pixels belong to the same class is high. This can be done through relaxation procedures [14].

Probabilistic relaxation procedures are procedures that modify the grades of membership of a pixel by taking into account the grades of membership of other pixels in a given neighbourhood:

\[
\mu_{k,c}^{(n+1)} = \frac{\mu_{k,c}^{(n)}(1 + q_{k,c}^{(n)})}{\sum_{c' = 1}^{C} \mu_{k,c'}^{(n)}(1 + q_{k,c'}^{(n)})}
\]

(12)

where \( q_{k,c} \) is a correction factor, \( n \) and \( n+1 \) are successive iteration numbers.
\[ q_{k,c}^{(n)} = \sum_{p \in \mathcal{N}(p)} \sum_{c'} K(c, c'), \mu_{k,c}^{(n)} \]  

(13)

where \( K(c, c') \) is a compatibility coefficient between class \( c \) and class \( c' \).

We have selected two possibilities for \( K \) \[15, 12\]

a) \( K(c, c') = \delta_{c,c'} \)  

(14)

With this option (\( \delta \) is the Kronecker symbol), the relationships between the different classes are not taken into account. Probabilistic relaxation is only performed within individual classes.

b) \( K(c, c') = \rho(c, c') = E \left[ \left( \frac{\mu_{k,c} - \bar{\mu}_c}{\sigma_c} \right) \left( \frac{\mu_{k,c'} - \bar{\mu}_{c'}}{\sigma_{c'}} \right) \right] \),  

(15)

\( \rho \) is the correlation coefficient between the two fuzzy maps \( c \) and \( c' \). Here, probabilistic relaxation is performed taking into account the grades of membership to all the classes of pixels in the neighbourhood.

4. **Illustration through a real data set in the field of electron energy loss mapping.**

Figure 2 represents four elemental maps (C, Cr, Ni and Ti) of a Ni-based alloy.\(^8\) These maps were recorded by means of energy filtering transmission electron microscopy, according to methods previously described \[16, 17\]. This four-dimensional data set could be processed directly by some of the methods described above. However, in order to concentrate the useful information and facilitate the visualization of the different intermediate processes, we chose to perform dimensionality reduction first. For this, we used Correspondence Analysis (CA), one of the techniques available for linear dimension reduction. The three eigen-images obtained after Factorial Analysis of Correspondence are displayed in figure 3. Since the first two eigenvalues represent 90\% of the information contained in the whole data set (65\% and 25\%, respectively), we will work with these two components in the following. This helps us to look at a reduced feature space (\( N = 2 \)). Figure 4a shows the scatterplot built from the first two eigenimages. One can easily see four main clusters of points in this feature space, indicating the existence of four natural classes. In order to go towards automatic analysis of the feature space, we proceed to pdf estimation by the Parzen technique. A Gaussian kernel was used. One typical result is shown in figures 4b and 4c. Figure 4b is a two-dimensional display and figure 4c is a three-dimensional display, where the height is proportional to the estimated pdf.

\(^8\) This data set was recorded at the Research Institute for Electron Microscopy, Graz University of Technology, Austria, and kindly provided by Dr. F. Hofer.
Figure 4d is the plot of the number of modes ($M$) as a function of the smoothing parameter, i.e. the standard deviation of the Gaussian smoothing kernel. A large plateau is obtained for $M = 4$, indicating that clustering into four classes is reasonable. Figure 5a shows the result of applying the watersheds technique to the estimated pdf with four modes. Every grey level represents a different class. Figure 5b displays the result of propagating the labels found in the feature space into the real image space. This is the final result we expect from the clustering procedure. From this result, other quantification approaches can be applied in order to obtain quantitative results characterizing the studied area of the specimen.

Figure 6 shows the grades of membership obtained for the four classes in the reduced feature space. The grades of membership of pixels to the four classes, in the feature space, are shown in figure 6a. The corresponding grades of membership of pixels, in the image space, are shown in figures 6b. Figure 6c displays the grades of membership of pixels after probabilistic relaxation. As can be seen, the contrast is enhanced (grades of membership become closer to zero or one), which means that relaxation improves the certainty in the classification. The class labels obtained after defuzzification are shown in figures 7c and 7d, respectively, and should be compared to the results obtained through the watersheds procedure (figures 5b and 7a) and through the fuzzy method without relaxation (figure 7b). Figure 7c corresponds to within-class relaxation and figure 7d corresponds to within- and between-class relaxation.

**Conclusion**

In this paper, we showed that new clustering techniques may be more powerful than classical ones for clustering multi-component images, for microanalysis. The reason for this is that these new techniques make fewer assumptions concerning the shape and extension of clusters in the feature space. As an example, we described a method we contributed to develop, called Parzen-watersheds. We also described a fuzzy extension of this method. With the fuzzy version, each pixel is characterized by a grade of membership to the different classes. One advantage of this is that the grades of membership computed on the basis of the feature vector alone can be corrected by probabilistic relaxation in the real space. After probabilistic relaxation, a defuzzification process allows to transform the grades of memberships to labels.

The whole process was illustrated through an example concerning a limited number of elemental maps obtained through electron energy-loss mapping.
It should be stressed that the same process could also be applied to other types of microanalytical data sets. For instance, it could be applied to multi-spectral data sets, where each pixel is characterized by a full spectrum instead of a few concentration values. In this situation, the only difference is that the feature vector $V_k$ is a high-dimensional one. Therefore, dimension reduction is mandatory in this case. Otherwise, the process remains identical.
References

This journal issue was never printed, for technical reasons. So, the paper is only available via the Internet, at:
http://www.aber.ac.uk/~ecmwww/journal/smi/pdf/smi97-01.pdf
or
www.univ-reims.fr/INSERM514/bonnet/Publications/smi97-01.pdf
**Legends of figures**

**Figure 1**

Illustration of the limitations of classical clustering algorithms taking into account the concept of centre of class. a) two hyper-spherical clusters with similar extension b) one hyper-spherical cluster and one hyper-elliptical cluster c) two hyper-spherical clusters with different extension d) two non convex clusters.

In any case, the crosses represent the centres of the two classes and the vertical bar represents the boundary found by classical algorithms between the two clusters. The result is correct in case “a” only.

**Figure 2**

Four elemental maps of a Ni-based alloy, obtained through the processing of electron energy-loss filtered images. From this multi-component image, the aim is to concentrate the information into a single phase image.

**Figure 3**

First three factorial images obtained after applying Correspondence Analysis to the four elemental maps displayed in figure 2.

**Figure 4**

a) Scatterplot built with the first two factorial images. Every point represents one or several pixels. b) Probability density function of pixels estimated in the feature space span by the first two factorial components. Arrows indicate the position of the four modes. c) Same as b), represented as a three-dimensional plot. d) Plot of the number of modes of the pdf as a function of the kernel size. A large plateau can be observed for four modes, indicating that the number of modes can be estimated equal to four.

**Figure 5**

a) Partition of the feature space according to the four modes of the estimated pdf, using the watersheds approach. b) Partition of the image space, through back-mapping the labels obtained in the feature space.

**Figure 6**
a) Grades of memberships to the four classes in the feature space.
b) Grades of membership of pixels to the four classes, in the image space, before relaxation.
c) Grades of membership of pixels to the four classes, after probabilistic relaxation.

**Figure 7**

Labels of pixels, representing the final phase map:

a) Obtained through the watersheds procedure (this figure is identical to figure 5b)

Obtained through the fuzzy method:

b) Without probabilistic relaxation (this figure is similar to figure 5b)
c) With probabilistic relaxation (within classes)
d) With probabilistic relaxation (within and between classes)
Figure 1
Figure 2
Figure 4
Figure 6
Figure 7