



# Clustering huge data sets for parametric PET imaging

Hongbin Guo<sup>a,\*</sup>, Rosemary Renaut<sup>a</sup>, Kewei Chen<sup>b</sup>, Eric Reiman<sup>b</sup>

<sup>a</sup> Department of Mathematics and Statistics, Arizona State University, Tempe, AZ 85287-1804, USA

<sup>b</sup> Good Samaritan PET Center, Good Samaritan Regional Medical Center, Phoenix, AZ 85006, USA

Received 7 December 2002; received in revised form 20 March 2003

## Abstract

A new preprocessing clustering technique for quantification of kinetic PET data is presented. A two-stage clustering process, which combines a precluster and a classic hierarchical cluster analysis, provides data which are clustered according to a distance measure between time activity curves (TACs). The resulting clustered mean TACs can be used directly for estimation of kinetic parameters at the cluster level, or to span a vector space that is used for subsequent estimation of voxel level kinetics. The introduction of preclustering significantly reduces the overall time for clustering of multiframe kinetic data. The efficiency and superiority of the preclustering scheme combined with thresholding is validated by comparison of the results for clustering both with and without preclustering for FDG-PET brain data of 13 healthy subjects.

© 2003 Elsevier Ireland Ltd. All rights reserved.

*Keywords:* Dynamic PET; Neuroimaging; Clustering

## 1. Introduction

This paper concerns the estimation of quantitative information from sequences of positron emission tomographic (PET) images of the brain. Quantitative rather than qualitative characterization has the potential to provide important in vivo measurements of dynamic physiological and biochemical processes which can then be used to track or identify disease processes.

### 1.1. The parametric model

To illustrate our discussion we consider a model of flurodeoxyglucose (FDG) tracer dynamics for the estimation of the cerebral metabolic rate of glucose as a measure of the functional activity in the brain. This is set up as a differential model consisting of three

compartments: the FDG concentration in plasma,  $u(t)$ ; the FDG concentration in tissue,  $y_1(t)$ ; and the phosphorylated FDG (FDG-6-phosphate) concentration in tissue,  $y_2(t)$ . The dynamics of the tracer are then described by the initial value problem

$$\begin{aligned}\frac{dy_1}{dt} &= K_1 u(t) - (k_2 + k_3)y_1(t) + k_4 y_2(t); \\ \frac{dy_2}{dt} &= k_3 y_1(t) - k_4 y_2(t); \quad y_1(0) = 0, \quad y_2(0) = 0.\end{aligned}\tag{1}$$

Here the system rate constants are interpreted as following:

- $K_1$  is the transport rate from the blood to the extra-vascular space;
- $k_2$  is the transport rate back from the extra-vascular space to the blood;
- $k_3$  is the phosphorylation rate of the intra-cellular FDG by hexokinase enzymes to FDG-6-phosphate;

\* Corresponding author. Tel.: +1-480-965-0401.

E-mail address: [hb\\_guo@asu.edu](mailto:hb_guo@asu.edu) (H. Guo).

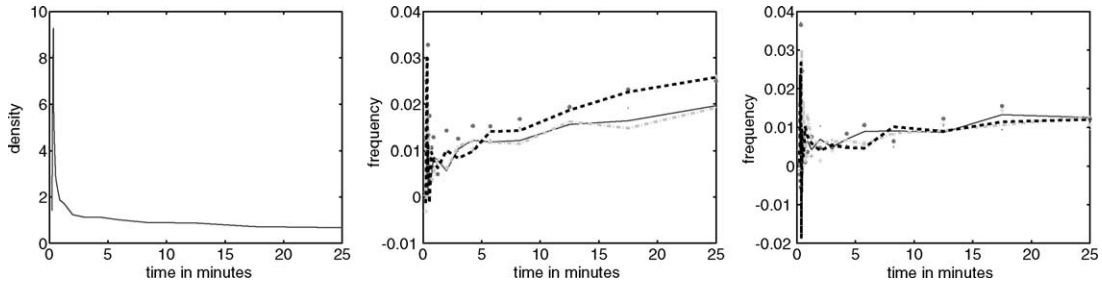


Fig. 1. Representative input function  $u(t)$  on the left, in the middle and right representative output data  $y(t)$  for a random selection of six active and inactive voxels, respectively. Note that the data are measured at times 0.1, 0.2167, 0.25, 0.2833, 0.3167, 0.35, 0.3833, 0.4167, 0.45, 0.55, 0.7167, 0.9, 1.25, 2, 3, 4.25, 5.75, 8.25, 12.5, 17.5, 25 and 45 min.

- $k_4$  is the dephosphorylation rate of the intra-cellular FDG-6-phosphate back to FDG.

Given the input function  $u(t)$ , we obtain the solution  $y(t) = y_1(t) + y_2(t)$ , by rewriting the system (1) as a second-order differential equation, and expressing  $y(t)$  as a convolution of  $u(t)$  with an exponential function  $f(t)$ , determined by the coefficients of the differential system, see for example, Huang et al. (1980). Based on this model, we are interested in estimating the kinetic parameters  $K_1$ ,  $k_i$ ,  $i = 2, 3, 4$ , and the local cerebral metabolic rate of glucose proportional to

$$K = \frac{K_1 k_3}{k_2 + k_3}. \quad (2)$$

To do this we need an estimate of the output function  $y(t)$  which can be compared with the analytic expression for  $y$  given  $u$  such as to obtain the unknown parameters.

In parametric studies the tissue concentration  $y(t) = y_1(t) + y_2(t)$  is the model output, taken to be the image intensity of a reconstructed PET image and we suppose that the FDG plasma concentration, the model input, is known. Typically, it can be obtained non-invasively by PET imaging of the blood supply through the carotid artery in the lower portion of the brain (Chen et al., 1998), or invasively by taking arterial or venous samples. Generally, the goal is the determination of the parametric values at the voxel level, and therefore  $y$  is the value of the intensity at a given voxel and Eq. (1) is assumed to describe the voxel level dynamics. We thus suppose that we have a sequence of volumetric PET images of the brain, taken at times  $t_i$ ,  $i = 1, \dots, n$ , where  $n$  is typically small, 22 in this study, and the time intervals  $\Delta t_i = t_{i+1} - t_i$

are not constant across the range  $t_1, \dots, t_n$ . There are many approaches for the solution of the optimization problem for the determination of the kinetic rate parameters by the matching of the output time activity curves (TACS) with the analytic representation of the solution, see for example, Feng et al. (1993). Here we do not focus on these techniques but on the data that is used in these algorithms.

Consider Fig. 1, which gives examples of the input function and the measured output for random choices of voxels, respectively.

There are several points to note about the data shown in these figures:

- The output data are very noisy. We know that the data indicate intensity values which correspond to densities, and should be positive.
- The data points indicate the unequal time intervals over which the PET data is obtained.
- The initial time intervals are very short in order to attempt to catch the first peak as the tracer reaches the tissue, which is crucial for catching the initial exponential increase of tracer in tissue, and hence assists with correct determination of the kinetic rates.
- The data are only illustrated through time  $t_{n-1}$ . The last time interval is very long, 20 min, and the output changes little over this interval which is, however, crucial for estimating the long time decay term in the output, and hence contributes significantly to the estimation of the kinetic rates.

These observations serve to elucidate the difficulties associated with determination of parametric images describing FDG tracer dynamics in the brain. In order to obtain statistically reliable estimates of kinetic

rate parameters at either the region of interest (ROI) or voxel level, it is important that we find reasonable estimates for the range of the parameters by initially dealing with the noise in the images. We consider an approach based on clustering of voxels which demonstrate similar dynamic behavior.

### 1.2. Clustering of the data

Clustering of data can be viewed as a form of segmentation of the data. The application of segmentation techniques has a rich history in the processing of magnetic resonance images (MRI), see for example, the overview by Clarke et al. (1995), but has not been widely used for parametric PET imaging, primarily because of the low signal-to-noise ratios (SNR). In particular, in dynamic PET, the SNR of the first images is much lower than that of the final images because of the need to use a short time interval to catch the early peak in the transport of tracer to the tissue. Recent research results, presented by several research groups, demonstrate that the incorporation of effective clustering as a preprocessing step prior to parametric estimation offers the opportunity to improve the accuracy of voxel level quantification (Kimura et al., 2002; Zhou, 2000). Kimura et al. assume  $k_4 = 0$ , and separate the character of parameter  $K_1$  from that of  $k_2$  and  $k_3$  by normalizing the data. The method of principal components is used to cluster the normalized data, including information for  $k_2$  and  $k_3$ . The parameter  $K_1$  is estimated voxel by voxel after  $k_2$  and  $k_3$  have been estimated cluster by cluster. Zhou uses the classical hierarchical average linkage (HAL) clustering method to generate a subspace of admissible TACs at the voxel level. Ashburner et al. (1996) use the assumption that the PET data satisfy a Gaussian distribution, and Acton et al. (1999) use fuzzy clustering. Then typical approaches for handling the poor SNR include the solution of the parametric estimation problem subject to constraints derived from the cluster solutions (Zhou et al., 2001), and/or the introduction of a weighted functional for the minimization, typically with weights proportional to the time interval for the measurement of the given frame of data. While these approaches are to some degree successful they are limited in their ability to provide accurate and efficient solutions for large four-dimensional data sets.

Here we propose a two-stage preclustering step to reduce the cost of the global clustering. Numerical results are presented which validate the preclustering technique as an effective preprocessing step which significantly improves the efficiency of clustering for parametric data.

### 1.3. Paper outline

In the remaining sections of the paper we present first a brief review of the key aspects of algorithms for clustering of data: the choice of distance measurement between clusters, how to add or remove elements from clusters, how to initialize clusters and how to decide the number of clusters. We then provide our rationale for the preclustering of the data and describe the preclustering algorithm, which can serve as a preprocessing step for any clustering algorithm. A brief outline of the algorithm for preclustering is presented in Appendix A. In Section 4, we use FDG dynamic data from 13 healthy individuals to show the improvement in efficiency and accuracy due to the inclusion of the specific preclustering phase as compared with results of clustering without preclustering. We conclude that the presented technique offers a novel cost effective and accurate option for quantification of functional PET data.

## 2. Segmentation of multivariable data by clustering

We consider the general case in which we have a set of multivariable data, vectors  $\mathbf{x}$  of length  $n$  with components  $x_l$ ,  $l = 1, \dots, n$ . Generally, the components of  $\mathbf{x}$  represent features of the data set, which is to be clustered, or *grouped* according to the similarity of these features. In our case the vectors  $\mathbf{x}$  are the measured TACs at each voxel. These vectors can be considered as the columns of a matrix of size  $n \times m$ , where  $m$  is very large. For example, with a relatively coarse resolution for three-dimensional (3D) brain data, we may expect to have 31 slices of two-dimensional (2D) data measured on a  $128 \times 128$  grid, yielding a total data set of  $2^{19}$  entries, each of which is a vector of values of time measurements. Some of the data lie outside the field of view of the imaged object, and thus the actual data set has size  $m$ , where  $m < 2^{19}$ . The

number of time frames is typically small, a moderate estimate would be 22 frames,  $n = 22$ .

We suppose that the underlying imaged object, here the brain, consists of a finite number of tissue types, the case for FDG modeling, say  $k$  different tissues which have different biochemical, or physiological, properties. The goal is to determine these underlying biochemical properties from representative TACs of the  $k$  different tissue types. With clustering of the TACs into  $k$  groups we intend to obtain groups such that ultimately TACs which are in the same group should be more similar, relative to some measurement of similarity, to each other than to TACs in other clusters. A representative member of each cluster, for example its mean TAC, may then be used to provide quantitative estimates of kinetic rate constants of the cluster. Further processing of this data for information at the voxel level is considered in other works and will not be emphasized here.

There are many excellent references which review the field of clustering, see for example, Jain and Dubes (1988), Kaufman and Rousseeuw (1990), Bezdek (1981). These algorithms differ in many ways such as the way that similarity between clusters is measured, how to add or remove elements to/from clusters, their initialization and whether assignment is revocable or not.

### 2.1. Distance measurement between TACs

We suppose that  $d(\mathbf{x}, \mathbf{y})$  is a measurement of the dissimilarity between the multivariable vectors  $\mathbf{x}$  and  $\mathbf{y}$ . For example, a typical choice would be the usage of

$$d(\mathbf{x}, \mathbf{y}) = \left( \sum_{l=1}^n |(x_l - y_l)w_l|^p \right)^{1/p}, \quad (3)$$

a weighted Minkowski  $p$ -norm which satisfies the usual mathematical requirements for a norm, and where the choice  $w_l = 1$  for all  $l$  would suppose that each feature  $x_l$  has equal significance. For  $p = 2$  this is the weighted sum of squares, or Euclidean distance, and for  $p = 1$  the weighted Manhattan norm. This distance may be calculated elementwise, i.e. for all elements in a given cluster as compared to all elements in another cluster, or with respect to a *representative* element of a cluster, such as the cluster centroid. The hierarchical centroid linkage (HCL), Sokal and

Michener (1958) uses the centroid  $\mu_I$  for group  $I$  as the representative group vector and the distance between clusters is given by the distance between the group vectors,  $D_{IJ} = d(\mu_I, \mu_J)$ . The distance of a single point to a single cluster is  $d(\mu_I, \mathbf{x}_j)$ . Zhou (2000) uses HAL, where the average distance between clusters  $I$  and  $J$ , of sizes  $n_I$  and  $n_J$ , respectively, given by

$$D_{IJ} = \frac{1}{n_I n_J} \sum_{i \in I} \sum_{j \in J} d(\mathbf{x}_i, \mathbf{y}_j) \quad (4)$$

is used to measure an average distance with respect to all pairs of elements (Sokal and Michener, 1958). For PET it is imperative that weighting is included in order to account for the difference in the SNR of the different time frames (see Fig. 2) due to the increasing time windows used in functional PET. We thus weight elements  $x_l$  with  $w_l = \Delta t_l$ , where  $\Delta t_l$  is the width of the  $l$ th time window.

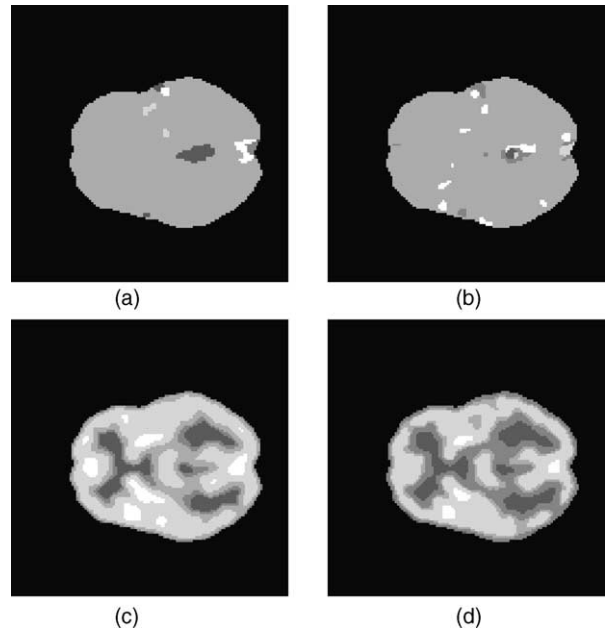


Fig. 2. Illustration of clustering to five clusters for slice 16 of an FDG-PET image of a healthy individual with different distances. From right to left and top to bottom these are: (a) Manhattan distance, (b) Euclidean distance, (c) weighted Manhattan distance and (d) weighted Euclidean distance.

## 2.2. Forming the clusters

Given a quantitative measure  $d$  of similarity or dissimilarity between TACs as defined by a given distance choice, one needs to determine how clusters should actually be formed. There are several options: hierarchical algorithms perform successive merging or splitting of the data according to a similarity measurement. For example, in the hierarchical linkage algorithm (HL), the algorithm is initialized via the calculation of a distance matrix of size  $m$ , where the element in the  $i$ th row and  $j$ th column represents the distance between objects  $i$  and  $j$  induced by the distance  $d(\mathbf{x}_i, \mathbf{x}_j)$ . Then at the first stage the two objects for which the  $ij$  entry is least are merged, hence forming the first cluster. The entries of the matrix are updated to reflect the distance of elements from this new cluster, hence reducing the size of the matrix by one. The process is repeated, pulling out entries into a new cluster at the next stage and continues, sweeping through the updated matrix at each step to determine the appropriate linking of objects. An algorithm which obtains clusters in this way is called a single linkage (Florek et al., 1951), or nearest neighbor, because the minimum requires only that one pair of elements is close, possibly overemphasizing the effects of outlier data. Linkage, called complete linkage (Sorensen, 1948), according to a maximum of  $d(\mathbf{x}, \mathbf{y})$  over all pairs  $(\mathbf{x}, \mathbf{y}) \in I \otimes J$  generates very compact clusters which tend to ignore outlier data.

A clear problem with these algorithms is that the initial matrix requires very high storage for the given brain data, due to the size of  $m$ , say  $2^{15}$  if just one-sixteenth of the data represents non-zero TACs. This problem is partially avoided if the number of clusters is assumed and initial elements for these clusters are chosen, then, if  $k$  is relatively small, the initial distance matrix is commensurately smaller. Moreover, because these algorithms are *irrevocable*, once an element is assigned to a cluster it cannot be reassigned, their effectiveness depends on a good initialization of the cluster elements.

## 2.3. Cluster initialization and number of clusters

In all of the above cases we may expect that the algorithms are made more efficient if we not only have a good estimate of the anticipated number of clusters

but also that we have a good initialization approach which includes as much of the known information as available. Here we do not focus on the former requirement, since it is usually anticipated that there will be no more than perhaps five different tissue types impacting the clustering. Instead, we present a novel automatic initialization of the clusters, *thresholding with preclustering*.

## 3. Methods

### 3.1. Histogram-based thresholding

Clustering of 4D PET data is prohibitive in time and memory. We seek to make the clustering of this data feasible in two ways: first by identifying active voxels, those in which the strongest kinetic activity is occurring as indicated by the voxels with highest intensity, and second by preclustering these active voxels. We illustrate the rationale of the first step by consideration of histograms for the density frequencies of multidimensional data at different time frames, see Fig. 3. We see that the final frame most clearly shows that there are differences in voxel activity over time: in the early frames there is significant evidence of noise from the reconstruction process, specifically negative values of the density because the sampling interval is small. This indicates that there are voxels in these frames with low tracer activity and thus very poor SNR for such small sample intervals. While we could chose to perform an initial characterization of voxels from any frame if some structural information were available, here we propose the automatic separation into active and inactive voxels using thresholding for the final frame. This first step is very straightforward, and simply relies on the identification of a separation between the two distributions using a mixture model based on thresholding, and will not be described in this report.

### 3.2. Preclustering

The initial thresholding into active and inactive voxels does not provide enough information to precluster the data, since clustering should include features over the entire TAC and not just the final frame. We assume, when preclustering is used prior

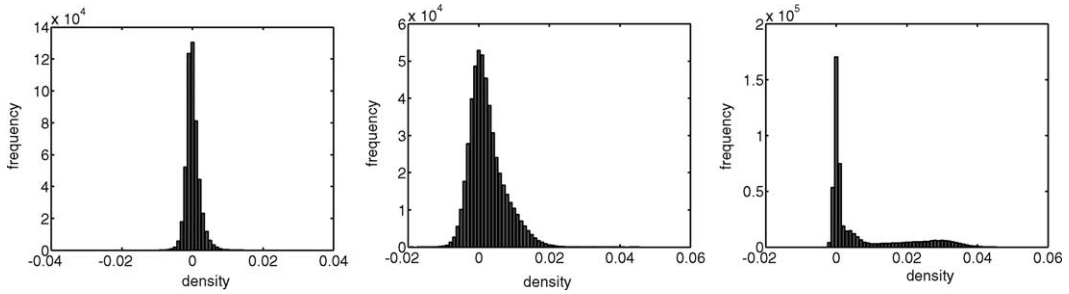


Fig. 3. Histograms of densities summed over all three spatial dimensions for time frames, from left to right, at the beginning, a middle frame at time 0.9 min and the last frame at time 45 min.

to an irrevocable algorithm, that the linkage order in the early stages has little effect on the final clusters, provided that the number of clusters is small relative to the overall size of the data set, which is certainly the case in this situation. Thus we propose a fast approach for the initial clustering of the data, followed by standard and more accurate clustering at the later stages.

To illustrate our idea, we analyze the histogram for the active voxels at the last time frame, see Fig. 4. The density which occurs with highest frequency is attained at a density around 0.03, with frequency 5934. If we identify a set,  $\mathcal{N}_0$ , of voxels in this frame with density near 0.03 we can calculate a mean TAC from the TACs for the voxels in  $\mathcal{N}_0$ . This mean TAC can then be used to find an initial cluster,  $\mathcal{G}_1$ , of voxels with TACs near this mean. Note that at this point we search over all voxels, not only those in  $\mathcal{N}_0$ , and distances are measured with respect to the entire TAC and not only the last frame. Removing voxels in  $\mathcal{G}_1$  from the histogram of the final frame we obtain the histogram in the right portion of Fig. 4

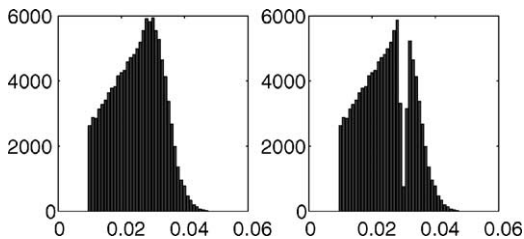


Fig. 4. Histogram of density of active voxels from a scan at the last time interval. The scan has 31 total slices, and each slice is  $128 \times 128$ . The left figure is the original histogram, and the right the histogram after removal of the initial cluster.

which now has two peaks on two separated intervals. We can repeat this process on each subinterval to find several preclusters for initialization of the cluster algorithm.

The specific parameter-dependent algorithm is provided in Appendix A, along with a modification which improves the preclustering by an additional step in which each non-clustered voxel is assigned to its closest cluster, over existing clusters. The ideas of the algorithms are described in the following subsections.

### 3.2.1. Parameter-dependent algorithm

Denote the density of voxels  $q$  in the chosen frame by  $f(q)$  and denote the frequency of the occurrence of  $f(q)$  by  $F(f(q))$ . The average frequency of  $f$  over density interval  $[a, b]$  is denoted by  $F_{\text{mean}}[a, b]$ . We introduce an integer labeling associated with each voxel, which not only identifies the voxels under consideration, but is then subsequently used to identify the cluster to which the voxel is assigned. Hence define, initially, labeling array  $B(q) \times B(q) = 0$  for inactive voxels and  $B(q) = 999$ , otherwise. The choice 999 is arbitrary but represents a significant overestimate on the maximum number of clusters anticipated. The initial interval for preclustering, the current active interval  $[a, b]$ , is defined such that  $a \leq f(q) \leq b$  for  $B(q) = 999$ .

The algorithm requires the specification of parameters,  $\kappa, \beta, \gamma_{\text{max}}$  and  $\gamma_{nk}$ , which set the tolerances for extraction of the preclusters, as follows. We define the density corresponding to the maximum frequency on the current interval  $[a, b]$  by  $f_{\text{max}}[a, b] = \arg \max_{f(q) \in [a, b]} F(f(q))$ . A set of voxels with density

near  $f_{\max}$  is defined by the set

$$\mathcal{N}_0 = \left\{ q : (f(q) \in [a, b]) \right. \\ \left. \& \left( \frac{|f(q) - f_{\max}[a, b]|}{f_{\max}[a, b]} \leq \kappa \right) \right. \\ \left. \& \left( \frac{F(f(q))}{F_{\text{mean}}[a, b]} \geq \beta \right) \& B(q) = 999 \right\}. \quad (5)$$

The first condition insures that we look only at voxels with density in the chosen interval on the histogram. The second condition, dependent on  $\kappa$ , seeks voxels with densities near the density with maximum frequency, measured relatively. Finally, the third condition, dependent on  $\beta$  seeks only voxels whose density has high frequency.

Associated with each entry  $q$  of  $\mathcal{N}_0$  is a TAC  $x_q$  for voxel  $q$  and we can calculate the average TAC( $\mathcal{N}_0$ ) over all  $q \in \mathcal{N}_0$ . At the next step we need to find all voxels with TACs which are close to TAC( $\mathcal{N}_0$ ) =  $\bar{\mathcal{N}}_0$ , measured with respect to a tolerance given by  $\gamma_{\max}$ , which is an estimation of the maximum relative radii of the preclusters. The initial cluster at stage  $nk$  of the preclustering process is then defined by the set

$$\mathcal{G}_{nk} = \left\{ q : \frac{d(x_q, \bar{\mathcal{N}}_0)}{d(0, \bar{\mathcal{N}}_0)} < \gamma_{\max} \& B(q) = 999 \right\}. \quad (6)$$

We refine the estimate of  $\mathcal{G}_{nk}$  by calculating its average TAC, updating its center to this new average, and thus determining its new radius. We do this recursively until the center becomes stable and such that  $\mathcal{G}_{nk}$  has a radius  $\gamma_{nk}$  which makes it sufficiently dense with respect to the frequency of occurrences of its el-

ements. Finally, we update the labeling of the voxels according to  $B(q) = nk$  for voxels  $q \in \mathcal{G}_{nk}$ , which removes these voxels from further process during the preclustering phase. We also calculate new intervals for further preclustering, the intervals  $[a, c]$  and  $[c, b]$ , where  $c$  is the average for  $f(q)$ :  $q \in \mathcal{G}_{nk}$ .

### 3.2.2. Parameter selection

The final clustering is most sensitive, in terms of effectiveness and efficiency, to the parameter  $\gamma_{nk}$  which determines the effective width and density of the  $\mathcal{G}_{nk}$ . Consider the case when all  $\gamma_{nk}$  are taken large: the voxels are quickly clustered, the preclustering produces fewer but larger clusters, and there is little work remaining for the final clustering. The limit of large  $\gamma_{nk}$  corresponds to an initial cluster which is the whole brain but the limit as all  $\gamma_{nk} \rightarrow 0$ , creates empty initial clusters. There is clearly a trade-off in accuracy and efficiency between smaller and larger choices for  $\gamma_{nk}$  and their choice is purpose dependent. If the goal is individual voxel clustering, then they should be really small so that the clustering is as accurate as possible. But if the goal is to generate mean TACs for subsequent refinement at the tissue level, then  $\gamma_{nk}$  can be chosen to be relatively large. In the latter case we have found that an appropriate choice can significantly reduce the size of the data set for the final clustering, while maintaining good accuracy in the calculation of the tissue TACs. An example of clustering 3D data is illustrated in Fig. 5, in which the data set was preclustered into 25 clusters (containing in total 91,215 voxels) and 27,917 isolated voxels, yielding a reduction in the size of the final data set for clustering by more than 75%.

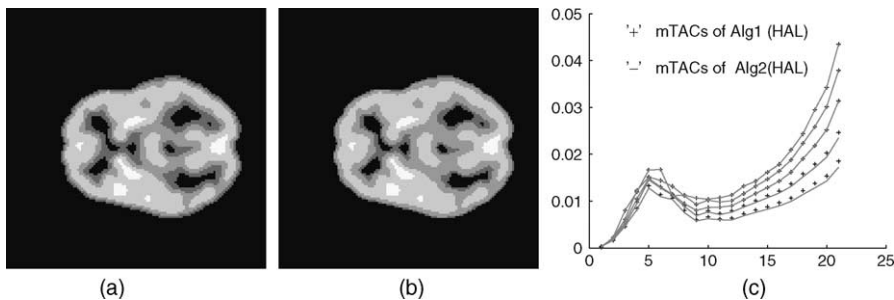


Fig. 5. 3D clustering: slice 16 of subject 4277, and the resulting average TACs for each cluster on the whole brain. From left to right (a) Algorithm 1, (b) Algorithm 2 and (c) the mean TACs for each of the five clusters.

Table 1

Computational cost with and without preclustering, measured in hours (h), minutes (min) and seconds (s)

Algorithm	2D (average)		3D (4277)	
	HCL	HAL	HCL	HAL
1	7 s	68 s	12 min 37 s	10 h 20 min
2	6 s	50 s	46 s	8 h 10 min
HL	2 h 1 min	2 h 33 min	NA	NA

Cost for 3D data of subject 4277, and for 2D data averaging over all 13 subjects. NA indicates that the algorithm did not complete because of memory requirements.

#### 4. Experimental results

PET-FDG brain data from 13 healthy subjects were used for the evaluation of both HAL and HCL algorithms with and without preclustering. In these experiments we use parameters which we have found to be optimal,  $\beta = 1$ ,  $\kappa = 0.05$  and  $\gamma_{\max} = 0.085$ , and which should be suitable as default values for similar PET studies. We would assume some initial effort be applied to assess this default choice for other studies, probably dependent on scanner resolution, time interval of final scan, and tracer under consideration. Once determined for a given data set and trial, the pa-

rameters should be held fixed, thus introducing only a small initial overhead. Experiments use Matlab13. In all presented figures the notation *mTAC* denotes the mean TAC for the specific cluster. In Figs. 5–7, where results are reported for more than one algorithm, the ‘—’ indicates the continuous line, and the other results, indicated by ‘+’ and/or ‘.’ symbols are superimposed.

##### 4.1. Test 1: 3D data

This test is used to show the difficulty of standard clustering of 3D data. For the data from subject 4277

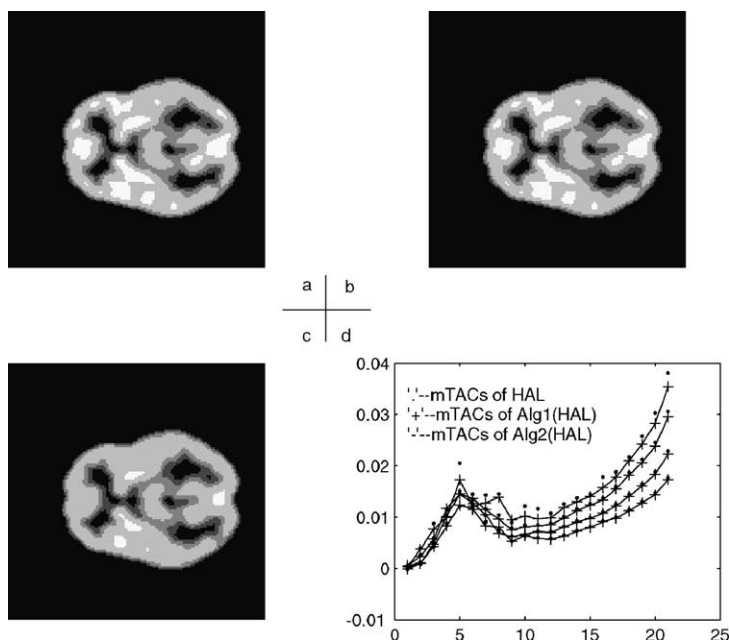


Fig. 6. Clustering of slice 16 for subject 4277. From left to right, and top to bottom: (a) HAL, (b) Algorithm 1, (c) Algorithm 2 and (d) the mean TACs for the clusters.

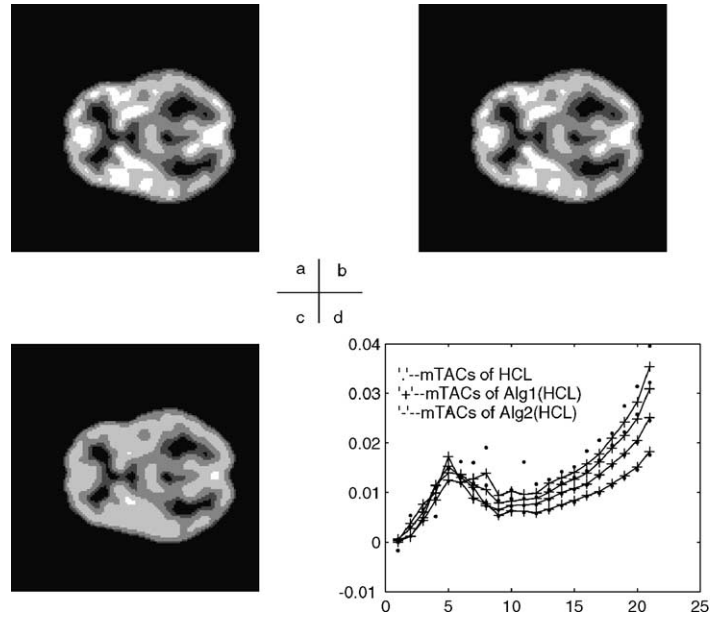


Fig. 7. Clustering of slice 16 of subject 4277. From left to right, and top to bottom: (a) HCL, (b) Algorithm 1, (c) Algorithm 2 and (d) the mean TACs for the clusters.

we selected 103,581 active voxels by thresholding the initial data. The resulting data set is still too large for standard clustering either using Matlab or SAS, on our Solaris Sun System 8. While the use of Mat-

lab or SAS clearly limits the algorithmic efficiency, it is also our goal to examine methods which are readily and easily implemented. After preclustering we have 25 clusters which are then reduced each to five

Table 2

Average and centroid linkage algorithms: HAL and HCL, with and without preclustering by Algorithms 1 and 2: the average relative difference  $\nu_{01}$ ,  $\nu_{02}$  between the mean TACs generated by HAL and Algorithm 1, and Algorithm 2, respectively and between both Algorithms 1 and 2,  $\nu_{12}$ , and the average relative difference  $\mu_{01}$ ,  $\mu_{02}$  between the mean TACs generated by HCL and Algorithm 1, and Algorithm 2, respectively and between both Algorithms 1 and 2,  $\mu_{12}$

Subject	$\nu_{01}$	$\nu_{02}$	$\nu_{12}$	$\mu_{01}$	$\mu_{02}$	$\mu_{12}$
4290	3.1E-02	3.2E-02	9.8E-04	2.8E-02	2.8E-02	7.6E-04
3004	1.4E-02	8.9E-03	5.5E-03	6.4E-03	1.2E-02	5.4E-03
4264	2.7E-02	2.7E-02	7.1E-05	4.8E-02	4.8E-02	1.5E-04
4277	3.6E-02	3.6E-02	2.0E-04	4.9E-02	4.9E-02	2.0E-04
4248	3.0E-02	3.1E-02	2.5E-04	4.6E-02	4.6E-02	1.9E-04
4143	1.1E-02	1.1E-02	2.3E-04	4.2E-03	4.2E-03	2.4E-04
4164	2.7E-02	2.7E-02	6.2E-04	5.3E-02	5.1E-02	1.6E-03
4115	4.5E-02	4.5E-02	4.5E-04	5.3E-02	5.3E-02	2.2E-04
4214	1.4E-02	1.4E-02	3.0E-04	1.6E-02	1.6E-02	3.4E-04
4222	2.3E-02	2.2E-02	4.9E-04	2.4E-02	2.5E-02	5.3E-04
3577	3.9E-02	3.9E-02	3.7E-04	4.0E-02	4.0E-02	1.5E-04
3246	3.1E-02	3.1E-02	8.2E-04	4.1E-02	4.1E-02	1.2E-03
4179	2.4E-02	2.6E-02	2.1E-03	2.3E-02	2.4E-02	1.9E-03

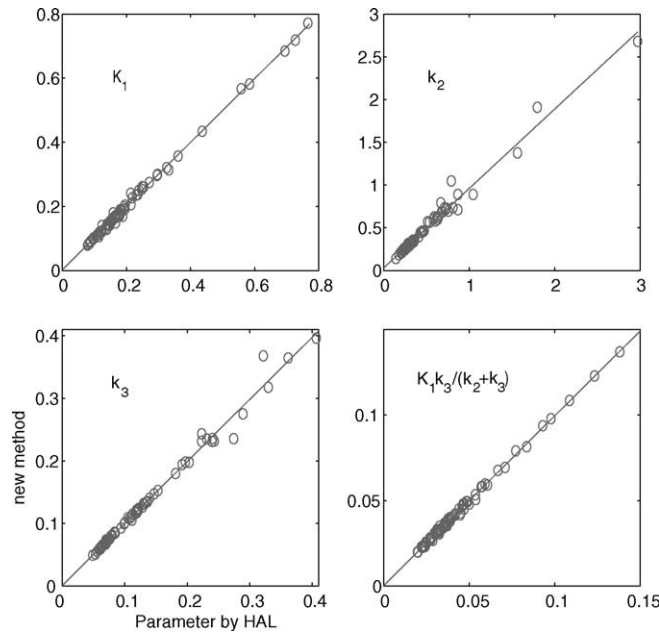


Fig. 8. Pairwise test for parameters  $K_1$ ,  $k_2$ ,  $k_3$  and Eq. (2), regression analysis estimating by preclustered HAL against HAL.

clusters using the standard HAL, see Fig. 5. Computational costs in terms of running times are reported in Table 1.

#### 4.2. Test 2: Comparison for 2D with and without preclustering

We validate the preclustering by Algorithms 1 and 2 as compared to the standard clustering for both HAL and HCL. The costs are also reported in Table 1 and the results in one case, subject 4277, are illustrated in Figs. 6 and 7. Results for all 13 subjects are presented in Table 2.

#### 4.3. Test 3: Statistical significance

To provide a statistical test of the reliability of the preclustered mean TACs we report results of pairwise testing for the calculation of the kinetic parameters, with  $k_4 = 0$ . The generalized linear least squares (GLLS), Feng and Huang (1996) method was used for estimating the parameters,  $K_1$ ,  $k_2$  and  $k_3$  from the average TACs generated by HAL and

Algorithm 1. These parameters and Eq. (2) are well estimated, see Fig. 8. For example, regression for  $K_1$  yields slope 0.994, intercept 0.002 and  $P < 10^{-8}$ .

## 5. Conclusions

We have demonstrated the feasibility of clustering techniques for the improvement of the calculation of parametric data from series of PET images. Because of special issues with time series data arising from sequences of PET images obtained at unequal time intervals, the weighting of distance measurements used for clustering is crucial. To deal with the vast quantity of data obtained from 3D volume measurements, it is also essential that the data are subjected to preclustering. A new technique for preclustering is described and validated. Numerical results presented demonstrated the validity, and efficiency of clustering for dynamic PET analyses. Future work will focus on increased resolution in clustering for identification of the input data from nonactive pixels.

## Acknowledgements

All authors were supported by the Arizona Alzheimer's Research Centre with funding from the Arizona Department of Health Services and the Alzheimer's Disease Core Centre, Phoenix, AZ. The second author acknowledges the award of the John von Neumann Visiting Professorship by the Technical University of Munich, Germany.

## Appendix A

**Algorithm 1** (HL cluster with preclustering).

- (1) Initialization phase
  - Input parameters  $\beta, \gamma_{\max}, \kappa$ . Set  $nk = 1$ .
  - Filter out voxels with insignificant values:
    - Use thresholding to identify active voxels in the last frame.
    - Set  $B(q) = 0$  for all inactive voxels. **Else**  $B(q) = 999$ .
  - Find the first active interval  $[a, b]$  such that  $0 \leq a \leq f(q) \leq b$ .
- (2) Repeatedly work on active intervals until either no more active intervals are available or a maximum iteration number has been reached as follows:
  - **For** active interval  $[a, b]$ 
    - Calculate  $f_{\max} = \arg \max_{f(q)} F(f(q))$ , for  $f(q) \in [a, b]$ .
    - Find the current initial cluster  $\mathcal{N}_0$  according to (5).
    - **If**  $|\mathcal{N}_0| < 2$  determine that current interval is inactive.
    - **Else**
      - Calculate the mean TAC for  $\mathcal{N}_0, \bar{\mathcal{N}}_0$ .
      - Find the  $nk$ th precluster  $\mathcal{G}_{nk}$  corresponding to relative radius  $\gamma_{nk}$ .
      - If**  $|\mathcal{G}_{nk}| < 2$  determine that current interval is inactive.
      - Else**
        - Set  $B(q) = nk$  for  $q \in \mathcal{G}_{nk}$ .
        - Form new active intervals  $[a, c], [c, b]$ ,  $c = \text{average of } f(q) : q \in \mathcal{G}_{nk}$ .
        - Update cluster number  $nk = nk + 1$ .
    - **End if**
  - **End for**
  - **End for**

- (3) Do HL clustering on the reduced data set, i.e.  $\mathcal{G}_{nk}$  and all voxels for which  $B(q) = 999$ .

**Algorithm 2** (HL clustering with preclustering and classification).

- (1) Carry out all steps in [Algorithm 1](#) except the last HL clustering step.
- (2) For all preclusters  $\mathcal{G}_j$  calculate the mean TAC,  $\bar{\mathcal{G}}_j, j = 1, \dots, nk$ .
- (3) For all voxels  $q: B(q) = 999$  calculate  $d(\mathbf{x}_q, \bar{\mathcal{G}}_j) j = 1, \dots, nk$ .
- (4) Merge voxel  $q$  with cluster  $\mathcal{G}_{j^*}$ , such that  $d(\mathbf{x}_q, \bar{\mathcal{G}}_{j^*}) = \min_j d(\mathbf{x}_q, \bar{\mathcal{G}}_j)$ .
- (5) Join updated clusters by HL clustering until we reach the given number of clusters.

## References

- Acton, P.D., Pilowsky, L.S., Kung, H.F., Ell, P.J., 1999. Automatic segmentation of dynamic neuroreceptor single-photon emission tomography images using fuzzy clustering. *Eur. J. Nucl. Med.* 26 (6), 582–590.
- Ashburner, J., Haslam, J., Taylor, C., Cunningham, V., Jones, T., 1996. A cluster analysis for the characterisation of dynamic PET data. In: Myers, R., Cunningham, V., Bailey, D., Jones, T. (Eds.), *Quantification of Brain Function Using PET*. Academic Press, San Diego, CA, pp. 301–306.
- Bezdek, J.C., 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York.
- Chen, K., Bandy, D., Reiman, E., Huang, S.-C., Lawson, M., Feng, D., Yun, L.-S., Palant, A., 1998. Noninvasive quantification of the cerebral metabolic rate for glucose using positron emission tomography,  $^{18}\text{F}$ -fluorodeoxyglucose, the Patlak method, and an image-derived input function. *J. Cereb. Blood Flow Metab.* 18, 716–723.
- Clarke, L.P., Velthuizen, R.P., Camacho, M.A., Heine, J.J., Vaidyanathan, M., Hall, L.O., Thatcher, R.W., Silbiger, M.L., 1995. MRI segmentation: methods and applications. *Magn. Reson. Imag.* 13 (3), 343–368.
- Feng, D., Huang, S., 1996. An unbiased parametric imaging algorithm for nonuniformly sampled biomedical system parameter estimation. *IEEE Trans. Med. Imag.* 15 (4), 512–518.
- Feng, D., Wang, Z., Huang, S.-C., 1993. A study on statistically reliable and computationally efficient algorithms for generating local cerebral blood flow parametric images with positron emission tomography. *IEEE Trans. Med. Imag.* 2 (2), 182–188.
- Florek, K., Lukaszewicz, J., Perkal, J., Zubrzycki, S., 1951. Sur la liaison et la division des points d'un ensemble fini. *Colloquium Mathematicae* 2, 282–285.

- Huang, S.-C., Phelps, M.E., Hoffman, E.J., Sideris, K., Selin, C.J., Kuhl, D.E., 1980. Noninvasive determination of local cerebral metabolic rate of glucose in man. *Am. J. Physiol.* 238 (E), 69–82.
- Jain, A.K., Dubes, R.C., 1988. *Algorithms for Clustering Data*. Advanced Reference. Prentice-Hall, Englewood Cliffs, NJ.
- Kaufman, L., Rousseeuw, P.J., 1990. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Series in Probability and Statistics. Wiley/Interscience, Wiley, New York, Chichester, Brisbane, Toronto, Singapore.
- Kimura, Y., Senda, M., Alpert, N., 2002. Fast formation of statistically reliable FDG parametric images based on clustering and principal components. *Phys. Med. Biol.* 47 (3), 455–468.
- Sokal, R.R., Michener, C.D., 1958. A statistical method for evaluating systematic relationships. *Univ. Kansas Sci. Bull.* 38, 1409–1438.
- Sorensen, T., 1948. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content. *Det. Kong. Danske Vidensk. Biol. Skr. (Copenhagen)* 5 (4), 1–34.
- Zhou, Y., 2000. Model fitting with spatial constraint for parametric imaging in dynamic PET studies. Ph.D. thesis, UCLA, Biomedical Physics, with Sung-Cheng Huang advisor.
- Zhou, Y., Huang, S.-C., Bergsneider, M., Wong, D.F., 2001. A nonlinear ridge regression with spatial constraint for generation of parametric images in dynamic PET studies. *J. Nucl. Med.* 42 (Suppl. 5), 100.