

# Clustering for three dimensional Kinetic PET Data \*

Hongbin Guo  
Rosemary Renaut  
Department of Mathematics  
Arizona State University  
Tempe, AZ 85287-1804  
hb\_guo@asu.edu  
renaut@asu.edu

Kewei Chen  
Good Samaritan PET Center  
Good Samaritan Regional Medical Center  
Phoenix, AZ 85006  
kchen@math.la.asu.edu

## Abstract

Clustering for three dimensional kinetic positron emission tomography (PET) data is considered. Recently, a fast preprocessing clustering technique for kinetic PET data was introduced by Guo et al. 2003. It is, however, still limited with respect to efficiency for three dimensional PET data. Here we present a two level clustering process which combines a slice by slice two dimensional clustering and a classic hierarchical clustering. We compare this method with our previous algorithm by clustering FDG-PET brain data of 12 healthy subjects. The new method significantly reduces the overall time and memory demand without loss of quality. September 29, 2004

## 1 Introduction

We consider a specific positron emission tomography (PET) problem, estimation of the functional activity in the brain through a three-compartmental model of fluoro-deoxyglucose (FDG) tracer dynamics. This is set up as a differential model consisting of three functions: 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

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

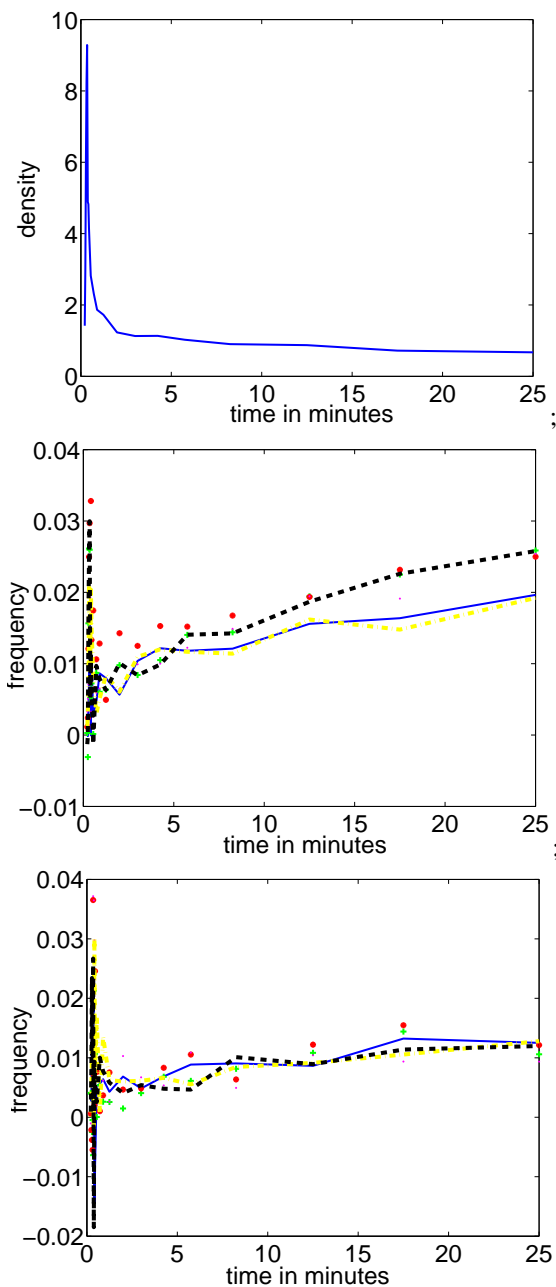
\*This work has been supported by NIH grant R21 EB02533-01 and in part by the Arizona Alzheimer's Disease Research Center which is funded by the Arizona Department of Health Services, and R. Renaut was also supported by NSF grant 0222327.

$$y_1(0) = 0, \quad y_2(0) = 0.$$

Here  $K_1, k_2, k_3$  and  $k_4$  are the system rate constants. Given the input function  $u(t)$ , and output function, the time activity curve (TAC),  $x(t) = y_1(t) + y_2(t)$ , we are interested in estimating the kinetic parameters  $K_1$  and  $k_i, i = 2, 3, 4$ . The TAC values,  $x(t)$ , are the observed intensity values of the reconstructed three dimensional PET image. Thus, while  $u(t)$  is assumed the same throughout the brain,  $x(t)$  varies for each unit of volume in the image, namely from voxel to voxel. Equivalently  $x(t)$  is shorthand for  $x_{ijk}(t)$ , where the index  $(i, j, k)$  indicates the position of the voxel in the volume. In turn the variation of  $x_{ijk}(t)$  reflects the variation in the kinetic parameters through the volume.

For a kinetic PET study, one is presented with a sequence of volumetric PET images of the brain, taken at times  $t_l, l = 1, \dots, n$ , where  $n$  is typically small, and the time intervals  $\Delta t_l = t_{l+1} - t_l$  are not constant across the range  $t_1, \dots, t_n$ . From the four-dimensional data,  $x_{ijk}(t)$  over all values of  $i, j, k$ , the goal is estimation of the kinetic rate constants in (1). In our study we use 22 PET images. Figure 1 gives examples of the input function and the measured output for random choices of voxels, resp.. The data are only illustrated through time  $t_{n-1}$ . The last time interval is very long, 30 minutes, 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.

Estimation of kinetic rate constants at either the region of interest (ROI) or voxel level 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. ROI analysis is a common approach to overcome the high level of noise in the voxel kinetics. Generally the defined ROI represents the extent of a specific tissue, the average of TACs over all voxels of the ROI is used to estimate kinetic parameters of this ROI.



**Figure 1. Representative input function  $u(t)$  on the top, in the middle and bottom representative output data  $x(t)$  for a random selection of 6 active and inactive voxels, resp.. 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, measured in minutes.**

Voxel level quantification has been hindered in the past by poor signal to noise ratio (SNR) in the TACs. Recent research results presented by several research groups, however, demonstrate that effectively clustering TACs as a pre-processing step prior to parametric estimation offers the opportunity to improve the accuracy of voxel level quantification, Kimura *et al* [6] and Zhou, [11].

There are three aspects of the difficulties with clustering PET data. A major difficulty is the inherent low SNR in the PET images due to many unavoidable characteristics such as scatter, poor spatial resolution, partial volume effects, and attenuation. Second, the result of clustering TACs should provide representative TACS which are associated nonlinearly with the chemical characteristics of the brain. In particular, unlike the case for structural image modalities, e.g. MRI and CT, there is not necessarily direct correspondence between anatomical and functional data. Finally, due to the size of the data sets, implementation of some clustering algorithms presents computational challenges. To overcome these difficulties various clustering techniques have been considered. Wong *et al*, [10], use a *k*-means-like method, Kimura *et al*, [6], use principal components, Zhou, [11], uses the classical hierarchical average linkage (HAL) clustering, Ashburner *et al*, [2], use the assumption that the PET data satisfy a Gaussian distribution, and Acton *et al*, [1], use fuzzy clustering. 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.

Let us compare *k*-means with hierarchical linkage (HL) for clustering  $m$  voxels. If the number of groups is pre-determined, *k*-means provides a fast technique for clustering (cost  $\mathcal{O}(m)$ ), and it tends to generate groups of comparable size. In contrast, the HL method, for which time and memory cost grows on the order of  $\mathcal{O}(m^2)$ , presents results with a flexible number of groups, and can distinguish small groups, which is particularly of interest in this PET application. The HL method is, however, too expensive for this PET data. Thus to accelerate the speed and to save memory, we introduced two fast HL methods in [5]. There, Algorithm 1 gives results comparable with standard HL, but is still expensive for real applications. Algorithm 2 performs much faster, and the average TACs of each cluster have good quality, but the resulting clusters are not as close to those of the standard HL.

In this report, we discuss a new algorithm based on Algorithm 1 in [5] which is cost competitive and generates high quality results. Specifically, we evaluate whether it is suitable to carry out certain aspects of the clustering on a slice by slice basis, rather than using the complete volume, in order to use repeatedly several data sets of smaller size rather than one large data set. The algorithm design is presented in section 2, our experimental results in section 3, followed

by conclusions in section 4.

## 2 Methods

### 2.1 Data structure

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 real values of time measurements. Some of the data lie outside the field of view of the imaged object, and thus the actual data set is of 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  $k = 5$  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. 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.

### 2.2 Algorithms

In this subsection, we review the fast HL algorithm, i.e. Algorithm 1 in [5]. The standard HL algorithm was reviewed in Guo *et al.*, [5]. In clustering the resulting segmentation is potentially highly dependent on the definition of dissimilarity, here the distance between two TACs. We define the distance between two voxels

$$d(\mathbf{x}, \mathbf{y}) = \sum_{l=1}^n |x_l - y_l|^p \Delta t_l, \quad (2)$$

which for  $p = 1$  is a weighted Manhattan distance and for  $p = 2$  a weighted Euclidean distance, in each case using the  $\Delta t_l$  weight. On the one hand, weighting is imperative due to the increasing size of the time windows,  $\Delta t_l$ , in functional PET, otherwise temporal information, kinetic information, may be lost. Moreover, it is used to account for the difference in the SNRs of the different time frames. On the other hand, Rudin *et al.*, [8], demonstrated that the total variance (TV) norm ( $p = 1$ ) is better than the Euclidean norm in

image restoration, especially for edge detection, and thus we base our algorithm on the weighted Manhattan distance. For comparison, see Figure 2.

We use the centroid, the average TAC, to represent a cluster, and call the corresponding clustering method the hierarchical centroid linkage (HCL), [9]. If we denote the centroid of cluster  $I$  by  $\mu_I$ , the distance between clusters is given by the distance between the average vectors,  $D_{IJ} = d(\mu_I, \mu_J)$ . The distance of a single point to a single cluster is  $d(\mu_I, \mathbf{x}_j)$ .

#### 2.2.1 Thresholding and Preclustering

Because the cost of the HL method grows on the order of  $O(m^2)$ , we introduce two techniques to reduce the size of the unclustered data set.

**Assumption 1: The reconstructed image of the last time frame is more stable than that of other frames and has relatively higher correlation with tissue function.**

First we identify active voxels, those in which the strongest kinetic activity is occurring, as indicated by the voxels with highest intensity. This is accomplished by **thresholding** on the final frame through its histogram. In general, the number of active voxels is about 20 percent of the total number of voxels of the 3-dimensional image.

**Assumption 2: Changing the linkage order in early stages of hierarchical linkage clustering has little impact on the later stages and the output.**

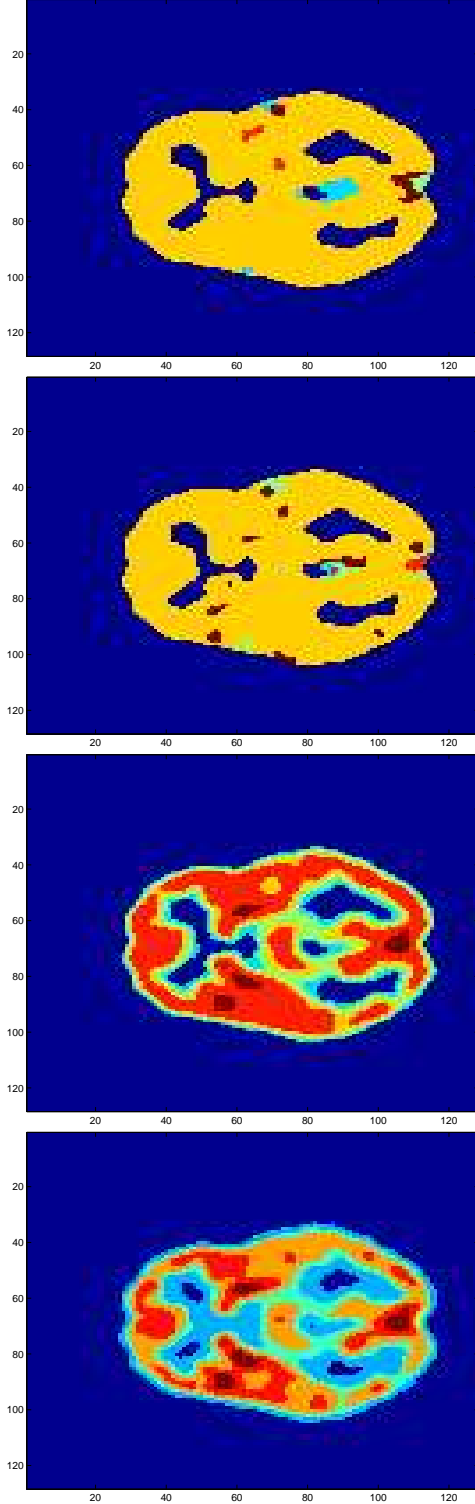
At the early stages of clustering the clusters are dense, i.e., the TACs are very close. Thus the order in which the TACs are grouped is initially irrelevant, and efficiency is more important. Thus we search for the initial clusters from the last time frame of data. We identify a set of voxels in the last frame with concentration near the highest frequency, and calculate a mean TAC from the TACs for the voxels in this set. This mean TAC can then be used to find a **precluster** of voxels with TACs near this mean. This procedure is repeated until no reasonable dense precluster is available, and then all the preclusters and remaining active voxels are passed to a HL method. We present the skeleton of the algorithm as follows, for details see [5].

**Algorithm 1** Whole Brain Volume (Algorithm 1 in [5].)

1. Identify active voxels by thresholding on last frame.
2. Initialize the cluster membership by preclustering.
3. Perform hierarchical linkage on the reduced data set.

#### 2.2.2 Slice by Slice Algorithm

At the first step of the whole brain algorithm for a FDG brain data set of  $128 * 128 * 31 = 507,904$  voxels, and 21 frames, thresholding, selects 119,132 active voxels. Step 2



**Figure 2. Illustration of clustering to 5 clusters for slice 16 of an FDG-PET image of a healthy individual. From top to bottom these are: Manhattan distance, Euclidean distance, Weighted Manhattan distance and Weighted Euclidean distance, (2).**

preclusters 91,215 voxels to 25 pre-clusters, leaving 27,917 isolated active voxels, and a total 27,942(=27,917+25) objects still to be clustered by HL, step 3. Here we suggest the use of Algorithm 1 slice-by-slice, and consider the HL algorithm applied at two levels. At the first level HL is applied for each slice independently, significantly reducing the sizes of the data sets to be clustered. At the second level HL is applied for the entire volume on the existing slice by slice objects, but again the data set to be clustered is now of reduced size. At the first level, on each slice, we suppose the data are clustered to  $N_i$  groups,  $i = 1, 2, \dots, S$ , where  $N_i$  should be significantly larger than the number of tissue types, say  $N_i = 20$ .

**Algorithm 2** Slice By Slice for  $S$  Slices

*Cluster each slice to  $N_i$  groups,  $i = 1, 2, \dots, S$ . Perform HL clustering on whole brain.*

1. *Identify active voxels by thresholding on last frame.*
2. *For each slice precluster the identified active voxels (step 2 Algorithm 1).*
3. *For each slice perform HL on the reduced data set to  $N_i$  groups (step 3 Algorithm 1).*
4. *Take all groups ( $\sum_{i=1}^S N_i$ ) and perform final HL step on whole brain.*

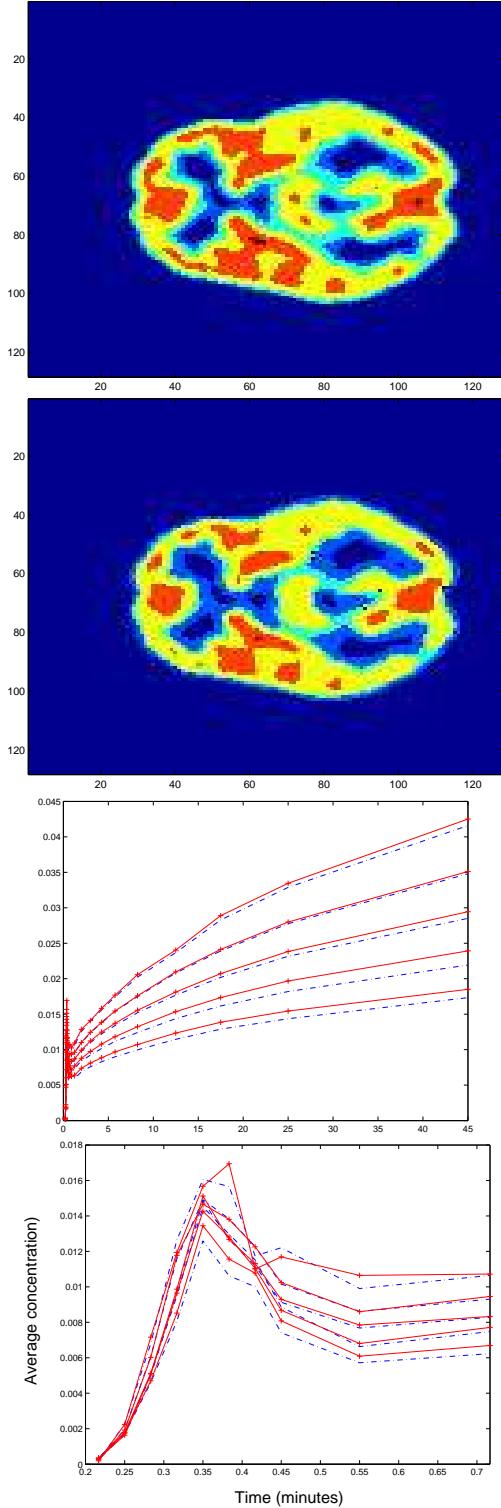
At steps 2 and 3, each slice data set is smaller than the equivalent step in Algorithm 1. At step 4, the whole brain HL, the data set is yet smaller,  $\sum_{i=1}^S N_i$ . The approach is consistent with the spirit of **Assumption 2**.

**3 Numerical Experiments**

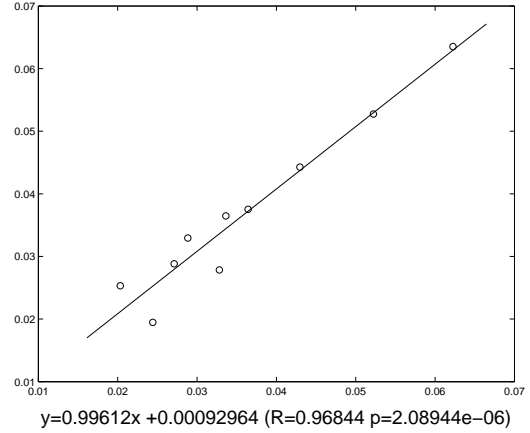
PET-FDG brain data from 12 healthy subjects are used for evaluation of kinetic parameters of mean TACs generated by the algorithms. Each data set has  $S = 31$  slices and  $n = 22$  frames of size  $128 \times 128$ . We delete the first frame because it is too noisy. For the implementation of HL clustering, we always use HCL. In our validation we obtain results by Algorithm 1 and 2. In Algorithm 2, we set  $N_i = 20$  for all slices. We calculate not only microparameters,  $K_1, k_2, k_3$ , but also parameter  $K$ ,

$$K = \frac{K_1 k_3}{k_2 + k_3}, \tag{3}$$

which is proportional to the local cerebral metabolic rate of glucose, a gold standard in Alzheimer’s disease research. For calculating these parameters, we use general linear least squares (GLLS) method with fixed  $k_4 (= 0)$ , [4], and the input functions are estimated from 3 later time stage venous samples, [3]. All tests are run with MATLAB 6.5 on a Solaris Sun System 8.



**Figure 3.** From top to bottom these are: slice 16 of a clustered 3D volume, Algorithm 1 and Algorithm 2; components of mean TACs, entire time window and windowed to time interval (0, 0.7167), solid line with '+' for Algorithm 1 and dot dash line for Algorithm 2.



**Figure 4.** Pair test for parameter  $K$ , Algorithm 2 against 1.

**Table 1.** Kinetic parameters Alg. 1/Alg. 2

$K_1$	$k_2$	$k_3$	$K$
0.117 / 0.122	0.295 / 0.275	0.089 / 0.086	0.0271 / 0.0288
0.127 / 0.128	0.228 / 0.204	0.082 / 0.081	0.0336 / 0.0365
0.132 / 0.133	0.163 / 0.157	0.079 / 0.078	0.0430 / 0.0443
0.133 / 0.133	0.121 / 0.119	0.078 / 0.078	0.0522 / 0.0527
0.139 / 0.141	0.092 / 0.090	0.075 / 0.074	0.0623 / 0.0635
0.072 / 0.050	0.281 / 0.326	0.145 / 0.208	0.0244 / 0.0195
0.050 / 0.050	0.304 / 0.203	0.208 / 0.208	0.0203 / 0.0253
0.086 / 0.079	0.196 / 0.242	0.121 / 0.132	0.0328 / 0.0278
0.080 / 0.086	0.230 / 0.195	0.131 / 0.121	0.0288 / 0.0329
0.102 / 0.103	0.184 / 0.176	0.102 / 0.101	0.0364 / 0.0375

### 3.1 Results

The average time for Algorithms 1 and 2 are 42 minutes 20 seconds and 2 minutes 18 seconds, thus confirming the expectation of improved efficiency of Algorithm 2. We pick a reasonable number of groups for each subject, here we always choose 5. Detailed results for one case are presented in Figure 3. Table 1 presents the comparison of parameters over 10 mean TACs. To validate the results we perform pairwise testing between the algorithms over the 10 mean TACs for parameter  $K$ , Figure 4, where  $R$  is the correlation coefficient and  $p$  is the probability that  $R = 0$ .

## 4 Conclusions

To obtain reliable estimates of parametric rate constants which describe the function of different tissues in the brain, it is essential to improve the SNR by clustering of four dimensional data. Typically this data set is vast and an efficient algorithm is required. In the design of an algorithm

that is both efficient and effective, it is crucial that the origin of the data is considered. Here, based on Assumption 1, the initial data set is reduced in size by thresholding to the most active voxels on the final frame. Further improvements in efficiency are realised by introducing a slice-by-slice preclustering based on Assumption 2. It is our hope that the adopted methodology of combining model knowledge to improve standard clustering algorithm efficiency can be implemented for other applications.

Numerical results presented demonstrate the validity, and efficiency of the approach. Moreover, the slice by slice technique leads itself readily to a parallel implementation. Postprocessing of the data to obtain voxel level dynamics is discussed by Negoita, [7].

## References

- [1] P. D. Acton, L. S. Pilowsky, H. F. Kung, and P. J. Ell. Automatic segmentation of dynamic neuroreceptor single-photon emission tomography images using fuzzy clustering. *European Journal of Nuclear Medicine*, 26(6):582–590, 1999.
- [2] J. Ashburner, J. Haslam, C. Taylor, V. Cunningham, and T. Jones. A cluster analysis for the characterisation of dynamic PET data. In R. Myers, V. Cunningham, D. Bailey, and T. Jones, editors, *Quantification of brain function using PET*, pages 301–306, San Diego, 1996. Academic Press.
- [3] K. Chen, D. Bandy, E. Reiman, S.-C. Huang, M. Lawson, D. Feng, L.-S. Yun, and A. Palant. Noninvasive quantification of the cerebral metabolic rate for glucose using positron emission tomography, 18F-fluorodeoxyglucose, the Patlak method, and an image-derived input function. *J. Cereb. Blood Flow Metab.*, 18:716–723, 1998.
- [4] D. Feng and S. Huang. An unbiased parametric imaging algorithm for nonuniformly sampled biomedical system parameter estimation. *IEEE Trans. Med. Imag.*, 15(4):512–518, 1996.
- [5] H. Guo, R. Renaut, K. Chen, and E. Reiman. Clustering huge data sets for parametric PET imaging. *BioSystems*, 71(1-2):81–92, 2003.
- [6] Y. Kimura, M. Senda, and N. Alpert. Fast formation of statistically reliable FDG parametric images based on clustering and principal components. *Phys. Med. Biol.*, 47(3):455–468, 2002.
- [7] C. Negoita. *Global Kinetic Imaging Using Dynamic Positron Emission Tomography Data*. PhD thesis, Arizona State University, 2003.
- [8] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [9] R. R. Sokal and C. D. Michener. A statistical method for evaluating systematic relationships. *Univ. Kansas Sci. Bull.*, 38:1409–1438, 1958.
- [10] K.-P. Wong, D. Feng, S. R. Meikle, and M. J. Fulham. Segmentation of dynamic PET images using cluster analysis. *IEEE Trans. Nuclear Science*, 49:200–207, 2002.
- [11] Y. Zhou. *Model Fitting with Spatial Constraint for Parametric Imaging in Dynamic PET Studies*. PhD thesis, UCLA, 2000. Biomedical Physics, with Sung-Cheng Huang advisor.