Neuroinformatics (2019) 17:103–114
https://doi.org/10.1007/s12021-018-9384-y

SOFTWARE ORIGINAL ARTICLE

QModeling: a Multiplatform, Easy-to-Use and Open-Source Toolbox for PET Kinetic Analysis

Francisco J. López-González[1,2] · José Paredes-Pacheco[1,2] · Karl Thurnhofer-Hemsi[1,3] · Carlos Rossi[3] · Manuel Enciso[3] · Daniel Toro-Flores[1] · Belén Murcia-Casas[4] · Antonio L. Gutiérrez-Cardo[1,5] · Núria Roé-Vellvé[1]

Published online: 28 June 2018

Abstract

Kinetic modeling is at the basis of most quantification methods for dynamic PET data. Specific software is required for it, and a free and easy-to-use kinetic analysis toolbox can facilitate routine work for clinical research. The relevance of kinetic modeling for neuroimaging encourages its incorporation into image processing pipelines like those of SPM, also providing preprocessing flexibility to match the needs of users. The aim of this work was to develop such a toolbox: QModeling. It implements four widely-used reference-region models: Simplified Reference Tissue Model (SRTM), Simplified Reference Tissue Model 2 (SRTM2), Patlak Reference and Logan Reference. A preliminary validation was also performed: The obtained parameters were compared with the gold standard provided by PMOD, the most commonly-used software in this field. Execution speed was also compared, for time-activity curve (TAC) estimation, model fitting and image generation. QModeling has a simple interface, which guides the user through the analysis: Loading data, obtaining TACs, preprocessing the model for pre-evaluation, generating parametric images and visualizing them. Relative differences between QModeling and PMOD in the parameter values are almost always below 10^{-8}. The SRTM2 algorithm yields relative differences from 10^{-3} to 10^{-5} when k_2' is not fixed, since different, validated methods are used to fit this parameter. The new toolbox works efficiently, with execution times of the same order as those of PMOD. Therefore, QModeling allows applying reference-region models with reliable results in efficient computation times. It is free, flexible, multiplatform, easy-to-use and open-source, and it can be easily expanded with new models.

Keywords Kinetic analysis · PET · Parametric images · SRTM · Patlak · QModeling

Introduction

Kinetic modeling is the basis for most methods of quantification of physiological properties based on dynamic PET data. Functional parameters like glucose metabolism (Patlak et al. 1983; Wienhard 2002) or neuroreceptor binding to appropriate tracers (Klumpers et al. 2007; Lammertsma et al. 1996b; Olsson et al. 2004) are routinely assessed at the pixel level, by applying widely-used kinetic models to images from well-established dynamic PET protocols (Bullich et al. 2010; Coello et al. 2017; Frankle et al. 2006; Keramida et al. 2017; Rodriguez-Vieitez et al. 2017; Salinas et al. 2015). The obtained parametric images are used in human and small animal studies of normal and pathologic physiology, disease progression and drug development (Bartmann et al. 2010; Lopes et al. 2017; Sérrière et al. 2014; Yo-Han et al. 2017).

The complexity of the involved calculations makes it necessary to use specific software. In the clinical research context, a free and easy to use toolbox, working with predefined, widely-used kinetic models would facilitate this kind of routine work. It would be particularly useful if set in a well-known environment providing tools for segmentation, normalization, and statistical analysis. Flexibility in the

Francisco J. López-González and José Paredes-Pacheco contributed equally to this work.

✉ Francisco J. López-González
fj.lopez@fguma.es

[1] Molecular Imaging Unit, Centro de Investigaciones Médico-Sanitarias, Fundación General de la Universidad de Málaga, Málaga, Spain

[2] Molecular Imaging and Medical Physics Group, Department of Psychiatry, Radiology and Public Health, Universidade de Compostela, Galicia, Spain

[3] Department of Computer Languages and Computer Science, Universidad de Málaga, Málaga, Spain

[4] Internal Medicine, Hospital Virgen de la Victoria, Málaga, Spain

[5] Nuclear Medicine, Hospital Regional Universitario, Málaga, Spain

processing pipeline would also be an advantage, so that users could adapt it to their own needs and resources, such as the availability, or lack of it, of an MRI scan for each individual, for instance. Furthermore, since there are also non-neurological applications to PET kinetic modeling, the best option for a new toolbox would be to allow the analysis of PET images for other purposes as well, both for humans and for animal models. For all these reasons, we have created QModeling (http://www.fil.ion.ucl.ac.uk/spm/ext/#QModeling), a free, easy-to-use, versatile and multiplatform software package for the kinetic analysis of PET studies. It is meant to be a toolbox within Statistical Parametric Mapping (SPM) (Wellcome Department of Imaging Neuroscience, London, UK; http://www.fil.ion.ucl.ac.uk/spm/), aimed at, but not restricted to, neuroimaging studies, allowing the application of predefined, commonly used kinetic models. It works locally, without the need of uploading sensitive data to online servers. It is also open-source, which can contribute to the continuous development, sharing and improvement of this kind of tools. Although there are already some packages offering kinetic analysis for PET imaging (Burger and Buck 1997; Fang et al. 2010; Gunn et al. 2016; Muzic and Cornelius 2001; Oikonen et al., n.d.; TKMF Program, n.d.), this is, to the best of our knowledge, the first one to provide all these features at a time.

The fact that SPM allows incorporating external toolboxes like this one in an easy way, which has allowed enriching it for various applications (Grotegerd et al. 2014; Keator et al. 2006; Schrouff et al. 2013; Tabelow et al. 2015), has also motivated the inclusion of QModeling in this environment. As SPM works on MATLAB, this has also set this programming environment as that of our software. This is the only cost users have to assume, as a MATLAB license is necessary for its usage.

In its initial version, our new software implements four reference tissue models: Simplified Reference Tissue Model (SRTM) (Lammertsma and Hume 1996), Simplified Reference Tissue Model 2 (SRTM2) (Wu and Carson 2002), Patlak Reference Tissue Model (Patlak et al. 1983) and Logan Reference (Logan et al. 1996). These are models with a reference region, which do not require the invasiveness and imprecision of blood sampling, while obtaining accurate and reproducible measurements on a large number of PET tracers (Lammertsma et al. 1996). SRTM and SRTM2 are among the most frequently used models in neurotransmission studies. Patlak is used for studies with tracers undergoing irreversible trapping, like fluorodeoxyglucose (18F-FDG), the most widely used PET radioligand. The Logan Plot is used for PET tracers with expected reversible union. It is widely used as an initial approach in the characterization of the kinetics of new radioligands, due to its short calculation times (Schain et al. 2017). The numeric results of the implemented methods, together with computation times, have been compared to those of a widely-used kinetic modeling software, PMOD, as a preliminary validation test.

Materials and Methods

Models and Implementation

SRTM was implemented following the algorithm described in (Gunn et al. 1997; Lammertsma and Hume 1996). Briefly, the SRTM operational equation is

$$C_T(t) = R_1 C_R(t) + R_1\left\{\frac{k_2}{R_1} - \frac{k_2}{1+BP}\right\} C_R(t) \otimes e^{-[k_2/(1+BP)+\lambda]t} \quad (1)$$

Where $C_T(t)$ is the concentration in a region of interest, R_1 is the ratio of the delivery in the tissue region of interest compared to that in the reference region (ratio of influx), $C_R(t)$ is the concentration in the reference region, k_2 is the efflux rate constant from the tissue of interest to plasma, BP is the binding potential, and λ is the physical decay constant of the isotope. Symbol ⊗ stands for the convolution operator.

In this approach, Eq. (1) is rewritten as

$$C_T(t) = \theta_1 C_R(T) + \theta_2 B_i(t) \quad (2)$$

where $\theta_1 = R_1$, $\theta_2 = k_2 - R_1 k_2/(1+BP)$, and $B_i(t) = C_R(t) \otimes e^{-k_{2a_i} t}$ with $k_{2a} = k_2/(1+BP) + \lambda$.

Thus, basis functions B_i (*bf*) are used to linearize the model and simplify the fitting process. The values for k_{2a} are taken in a logarithmic range, in an interval that the user can modify. The total number of basis functions can also be chosen. The basis functions are calculated through a convolution product. In order to increase precision in the convolution, $C_R(t)$ is usually resampled by linear interpolation to a finer grid. This improves precision but it may imply longer execution times.

Equation 2 is solved using linear least squares for each basis function, via a QR decomposition. Finally, the basis function yielding best results is chosen.

Since $k_2' = \frac{k_2}{R_1}$ in Eq. (1) is the clearance rate constant from the reference region, it has only one true value, and it should not be necessary to estimate it for every pixel, as SRTM does. The SRTM2 implementation (Wu and Carson 2002), uses this property to accelerate the calculations. Its approach is similar to that of SRTM, but it uses a fixed value for k_2' parameter. In our implementation, the value for k_2' can be established in two ways: either it is specified by the user, or it is estimated just once, by applying SRTM to a TAC of a receptor-rich region, using the appropriate reference region as well. From here on, the obtained value is considered fixed. This avoids estimating it for every voxel.

The Patlak Reference model (Patlak and Blasberg 1985) follows the equation:

$$\frac{C_T(t)}{C_R(t)} = K\frac{\int_0^t C_R(u)du}{C_R(t)} + V \quad (3)$$

Where $K = \frac{k_2k_3}{(k_2+k_3)}$, and $V = \frac{k_2k_3}{k_2^{REF}(k_2+k_3)} + \left(\frac{k_2k_3}{k_2^{REF}(k_2+k_3)}\right)\frac{k_2k_3}{k_2^{REF}(k_2+k_3)}\frac{C_1(t)}{C_R(t)}$, with k_2 the efflux rate constant from the tissue of interest to plasma, k_3 the rate constant from the tissue of interest to the tissue with irreversible binding and k_2^{REF} the rate constant from the tissue of reference to plasma and $C_1(t)$ the concentration of non-metabolized tracer in tissue.

A least squares linear fit allows resolving this model by estimating K and V.

The Logan Reference Tissue method (Logan et al. 1996) has the following operational equation:

$$\frac{\int_0^t C_T(\tau)d\tau}{C_T(t)} = DVR\frac{\int_0^t C_T(\tau)d\tau + C_T(t)/k_2'}{C_T(t)} + b \quad (4)$$

This resembles a linear equation, where the slope is the distribution volume ratio (DVR), i.e. the ratio between the distribution volumes in a receptor-rich region and in a non-receptor-containing region. The binding potential is calculated as $BP = DVR - 1$. The error term $b$decreases over time, and this dependence also becomes negligible after some time t^*. When only the values obtained after t^* are taken into account, a linear regression allows estimating DVR and b.The parameter representing the average tissue to plasma efflux, k_2', has to be specified as an input for the model.

Input Data

Besides the PET dynamic study, the models need some additional information to run. The reference region and a region with specific tracer uptake must also be specified. This can be done by loading an image file with masks for these regions of interest (ROIs), or by directly loading the TACs for these ROIs. A set of model-specific parameters must also be indicated before starting the analysis. If these parameters are not specified by the user, the program gives them default values. The parameters for each model are listed below.

SRTM: k_{2a}minimum, k_{2a}maximum, Number of *bf*, Resampling, Threshold.

k_{2a}minimum, k_{2a}maximum establish the limits of the logarithmic range for the exponent of the basis functions. The number of *bf* sets the interval at which these values will be considered. A larger number of basis functions, taken at smaller intervals, may enhance precision, at the expense of execution time. Resampling sets the interval for the grid of time values used to calculate the convolution product, $C_R(t)$. A larger value may improve precision, at the cost of longer execution times. The Threshold is defined as a percentage of the maximum intensity in each parametric image, and it is used to discard image voxels with lower values.

SRTM2: The parameters are the same as for SRTM, but, in this case, the value of k_2' can be either set to a fixed value, or estimated by the program. QModeling estimates k_2' by applying SRTM to a user-defined ROI, which avoids estimating k_2' for every voxel.
Patlak Reference: t^*, Max. Error, Threshold.

Parameter t^* defines the beginning of the temporal range used in the linear fit. If the residual sum of squares exceeds the user-specified maximum error (Max. Error), the fit is repeated with t^* set at the next frame of the study. Alternatively, the user can set t^* at a fixed point. The Threshold is, again, a percentage of the maximum intensity in each parametric image, used to exclude lower values.

Logan Reference: t^*, Max. Error, Threshold, k_2'.

Parameters t^*, Max. Error and Threshold have the same role as in the previous case, and k_2', is the average tissue to plasma efflux.

Preprocessing and Estimation of Parametric Images

Parameter estimation is performed in two steps. The first one is preprocessing. The selected kinetic model is applied to the mean TAC of the user-specified ROI with specific uptake. Then a fitted TAC is calculated with the obtained parameters and shown to the user, together with the original one. The fitted parameters and their standard errors (SE) are also presented. This allows checking if the initial values of the input parameters (Resampling, Threshold, etc.) are good enough to obtain a reasonable fit. In addition, for the SRTM and the SRTM2, the correlation matrix is included as a measure of model identifiability. Using the mean TAC of a ROI with specific uptake allows this pre-evaluation to be robust. The obtained parameters can subsequently be changed and the preprocessing repeated until the TAC fit is satisfactory, to optimize the input parameters before starting the voxel-level fitting.

The second step consists in fitting the model to the images voxel by voxel, using the checked input values. Parametric images are then generated.

Software

As mentioned, QModeling is based on SPM, which works on MATLAB, and was developed for the main operating systems (Windows, Linux, MacOS). The efficiency of MATLAB for matrix-level operations makes it a good option for the fitting process and the generation of parametric images in a reasonable

processing time. The Graphic User Interface (GUI) for QModeling was also programmed under MATLAB, due to its ease of use.

Preliminary Validation and Performance Analysis

The results provided by QModeling were evaluated by comparison with PMOD. Since PMOD is one of the most widely used kinetic modeling packages, and it has been thoroughly tested, we considered it a proper basis for a preliminary validation test. For this evaluation, both programs processed three [^{11}C]-FMZ-PET studies to which SRTM, SRTM2 and Logan were applied and three [^{18}F]-NaF-PET studies to which the Patlak model was fitted.

The [^{11}C]-FMZ-PET studies were obtained for three patients referred to our center for evaluation of temporal lobe epilepsy. They consist of 21 frames (6 × 5 s, 3 x 10s, 4 x 60s, 2 × 150 s, 2 × 300 s, 4 × 600 s) with 128 × 128 × 47 voxels, 1.95 × 1.95 × 3.27 mm in size. For the preprocessing, the pons was chosen as reference region (of about 900 voxels, depending on the study) and the occipital lobe as region of specific uptake (including around 8000 voxels).

The [^{18}F]-NaF-PET images were obtained from a research project approved by the local committee on human experimentation and all procedures followed were in accordance with the ethical standards of the Helsinki Declaration of 1975, as revised in 2000. These studies consist of 43 frames (25 × 5 s, 4 x 30s, 14 × 240 s) with 128 × 128 × 47 voxels (5.27 × 5.27 × 3.27 mm). In this case the aorta was chosen as reference region (including around 100 voxels, depending on the study) and a bone region drawn on the vertebrae (of about 30 voxels) as a specific uptake area.

The input parameters for the preprocessing were set to default values used in PMOD for SRTM, SRTM2 and Logan ($k_{2a}min$=0.006, $k_{2a}max$=0.6, number of bf = 100, Resampling = 5 and Threshold = 3% for SRTM and SRTM2; unfixed k_2' for SRTM2; unfixed t*, Max. Error = 10%, Threshold = 3% and k_2' =0.15 for Logan). Default values were also used for Patlak (unfixed t*, Max. Error = 10%) except for the Threshold, which was set to 0%, instead of 3%, for a better definition of the bone tissue. Results obtained from PMOD were taken as a *gold standard*. Two types of data were compared: the parameters obtained in the preprocessing step and the final parametric images. In the parametric images of SRTM and SRTM2, only positive values were taken into account, in order to avoid comparing areas outside the body, with zero values. A similar restriction was used for Patlak and Logan images, where only voxels where K > 0.001, for Patlak, or BP > 0, for Logan, were analyzed. No other thresholds were applied to the parametric images.

Absolute and relative errors were measured for the evaluation. For the case of parametric images, instead of voxel-by-voxel comparison of the images, the mean and maximum value of all voxels were considered for the comparison.

A comparative performance analysis between QModeling and PMOD was also carried out. The speed of TAC generation, model fitting and image generation were checked. This was done for all models: SRTM, SRTM2, Patlak and Logan. As the estimation of the k_2' parameter is done differently in QModeling and in PMOD, SRTM2 was tested twice: first with a fixed initial k_2' value and then with an estimated k_2' value.

Moreover, this comparison was repeated for different values of the input parameters, in order to see the dependency of the computation time on each of them. This was firstly done with the default values. Afterwards, the comparison was performed by re-estimating the processing time after modifying each of the studied parameters, leaving the rest at their default values. Thus, in the preprocessing for SRTM and SRTM2, 10 and 400 *bf* have been used. The tested Resampling values were 1 s and 2 s. For the k_{2a} parameter, the minimum values tried were 0.001 and 0.009 and the maximum values considered were 0.1 and 0.9. For Patlak and Logan Reference, the t^* values assayed were 5, 10 and 20 min, for fixed t^*. This test was done using the same PET studies as in the previous section.

Four PCs with different technical specifications and versions of MATLAB were used to measure computation times:

PC 1: Ubuntu 16.04 LTS (64 bits); Intel Core I7-6700 k (4.00 GHz); RAM 32 GB DDR4; HD 250 GB SSD; MATLAB r2015b; SPM12; PMOD v3.310.
PC 2: Ubuntu 16.04 LTS (64 bits); Intel Core i7 4790 K (4.00GHz); RAM 32 GB DDR3; HD 1 TB SSD; MATLAB R2016b, SPM12, PMOD v3.310.
PC 3: Os X Yosemite (10.10.1); Intel Core i5 (2.7 GHz); RAM 8 GB DDR3; HD 1 TB; MATLAB r2015b; SPM12; PMOD v3.310.
PC 4: Windows 7 Professional (64 bits); AMD Turiom 64 × 2 Mobile Technology TL-60 (2 GHz) DDR3; HD 120 GB; MATLAB r2012a; SPM12; PMOD v3.310.

Results

Implementation: Graphic User Interface

The main GUI of QModeling can be seen in Fig. 1. The program has been organized into four submodules, which can be accessed through this GUI (see boxes at the left and lower sides): Data loading and TAC generation; Model selection and preprocessing; Generation of parametric images andDisplay of parametric images.

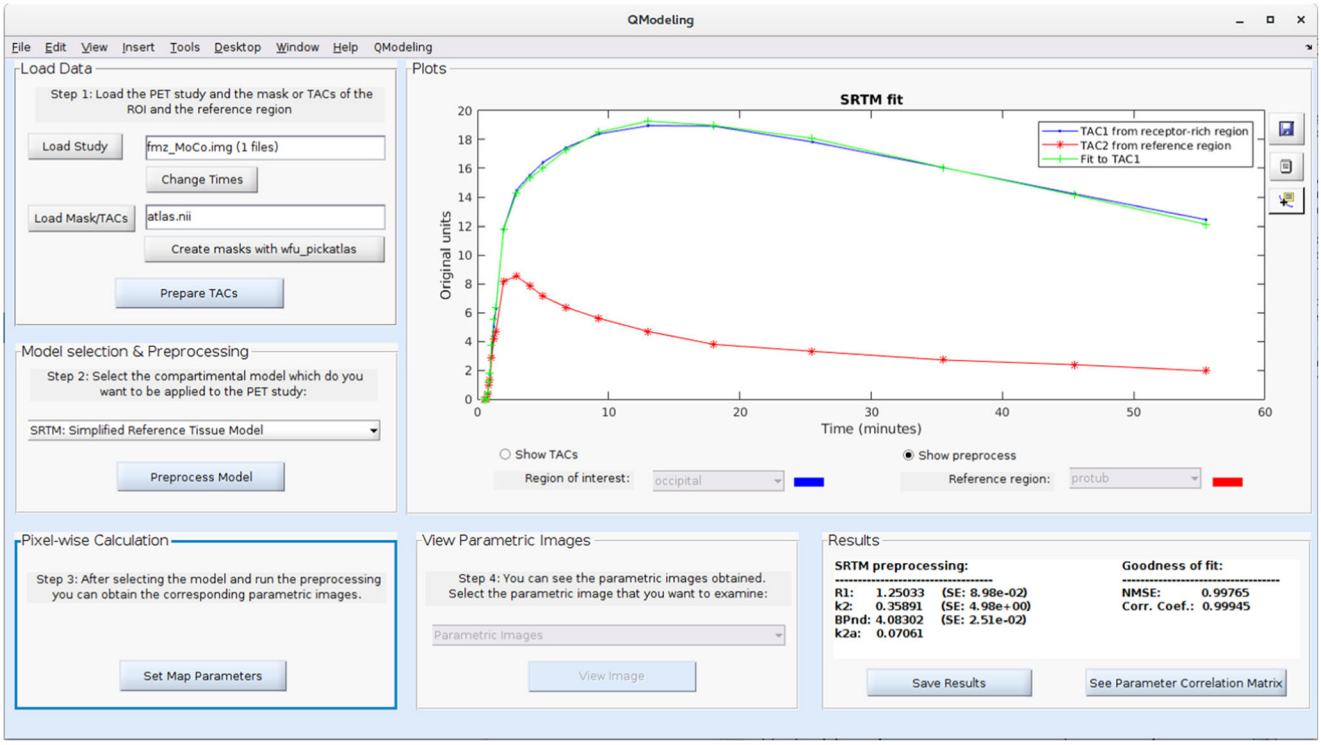

Fig. 1 QModeling GUI. In the figure, the PET study and a mask indicating both the reference region and a region with specific uptake have already been loaded. The TACs have been generated by clicking on "Prepare TACs", and they are displayed in the Plots frame (red and blue curves). The fitted curve is shown in green, and the fitted parameters can be seen in the Results frame. Preprocessing was executed with the default values

The main frame also contains a results zone, with a larger area where the graphs are displayed and a smaller one where the numerical results of the analysis are presented.

The workflow through these modules is illustrated in the flowchart in Fig. 2.

The first module is responsible for reading the data. A PET study must be loaded, together with information concerning the reference region and a ROI with specific uptake. This consists of either an image mask indicating these ROIs, or, alternatively, the TACs for these two regions. The program is prepared to read images in ANALYZE (.hdr/.img) or NIfTI format (.nii). If a mask is loaded, a text file with the numeric identifier of each of the ROIs in the mask and the associated name will also be searched. The user can later specify which of these ROIs is to be used as a reference region, and which one will be used as a region with specific uptake for the preprocessing. The option to open the toolbox WFUPickAtlas (Maldjian et al. 2003) is also included, provided it is already installed as an SPM toolbox, to facilitate the generation of masks.

After these data are read, the user has the option to see a table showing the acquisition time for each frame (button "Change times") in a new window. Here, the user can check if the information of the times is correct and load an alternative one if necessary.

If a mask has been loaded, a TAC is generated for each ROI it contains. Then the user can select one reference region and one ROI to apply the model to, from a drop-down list. The associated TACs are then displayed in the results area of the GUI (Plots frame).

At this point the preprocessing module is enabled. It is used to estimate the model parameters in the user-specified region of specific uptake.

In this module, the user chooses the kinetic model to be applied. When the "Preprocess Model" button is clicked on (Fig. 1), the program opens a configuration window (Fig. 3). Here, the user can select the TACs and the values of the input parameters to be used. The kinetic model is then fitted to the TAC of the region of specific uptake. A new TAC for this region is generated, based on the fitted parameters. Finally, both the fitted and the measured TACs are shown in the Plots area, and the parameters with the standard errors are displayed in the Results section. Furthermore, the Normalized Mean Squared Error (NMSE) and the Correlation Coefficient are also shown, as measurements of the goodness of fit. In addition, a specific button in the results section allows viewing and saving the correlation matrix when SRTM or SRTM2 are used. Thus, the user can check whether the model is properly fitted or if it is necessary to change the input parameters.

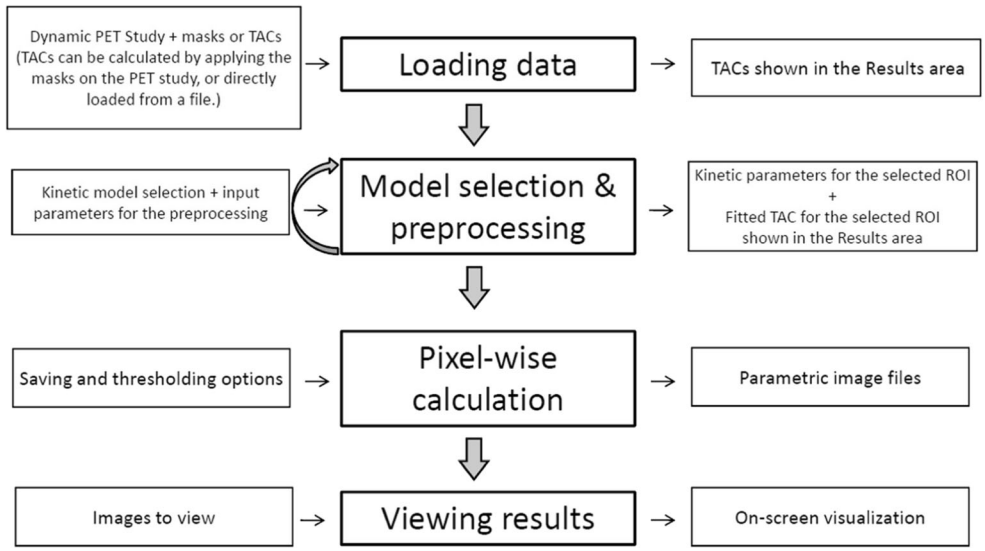

Fig. 2 Workflow for QModeling. First, a PET study is loaded, together with masks for the ROIs involved (reference region and region with specific uptake) or with the TACs for the relevant ROIs. If a mask has been loaded, QModeling generates these TACs. In the next step, the study is preprocessed with the kinetic model specified by the user. This gives the kinetic parameters for the ROI with specific uptake as well as the fitted TAC. This process can be repeated after changing the input parameters, until the result is satisfactory. Afterwards, parametric images can be calculated, and finally visualized using the viewer

During this initial fitting process, the parameter configuration window stays open. When it is closed, the last calculated values are taken as valid and saved. Then, the image generation module is enabled.

This module applies the selected model to the entire image. It also allows selecting the parametric images to be generated, applying thresholds to the resulting parameter values, and specifying image format and saving options (Fig. 4a).

Once images are generated, the last module is enabled (Fig. 4b). It allows selecting an image and then displays it in a specific window. The user can open multiple images simultaneously in different windows. To build this module, we used a library of functions developed by Jimmy Shen for NIfTI and ANALYZE image processing (Shen 2014).

Implementation: Command Line

The program can also be run from the MATLAB command line. All the functions included in the first three modules (Data loading and TAC generation; Model selection and preprocessing; Generation of parametric images) can be executed from it. Three different instructions can be typed in the command line, one for each module, and the selected model and the input parameters need to be specified between parentheses, as it is customary in MATLAB. Users can

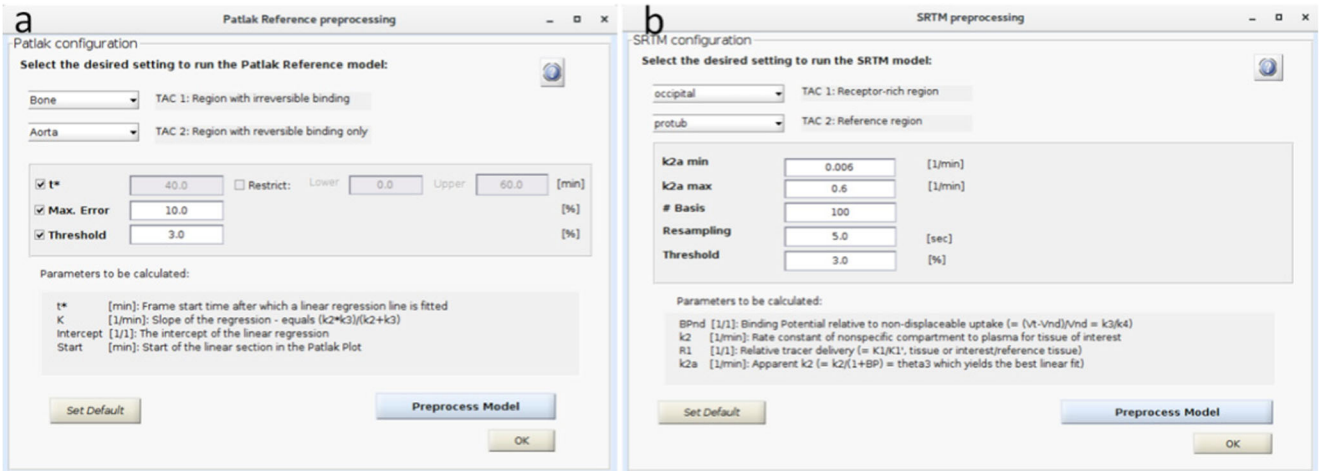

Fig. 3 Preprocessing configuration windows for Patlak Reference (a) and SRTM (b). When the window is opened, the previously displayed TACs are loaded. After specifying the input parameters, and pressing "Preprocess Model", the model is fitted, this window is minimized and the results are presented in the main screen

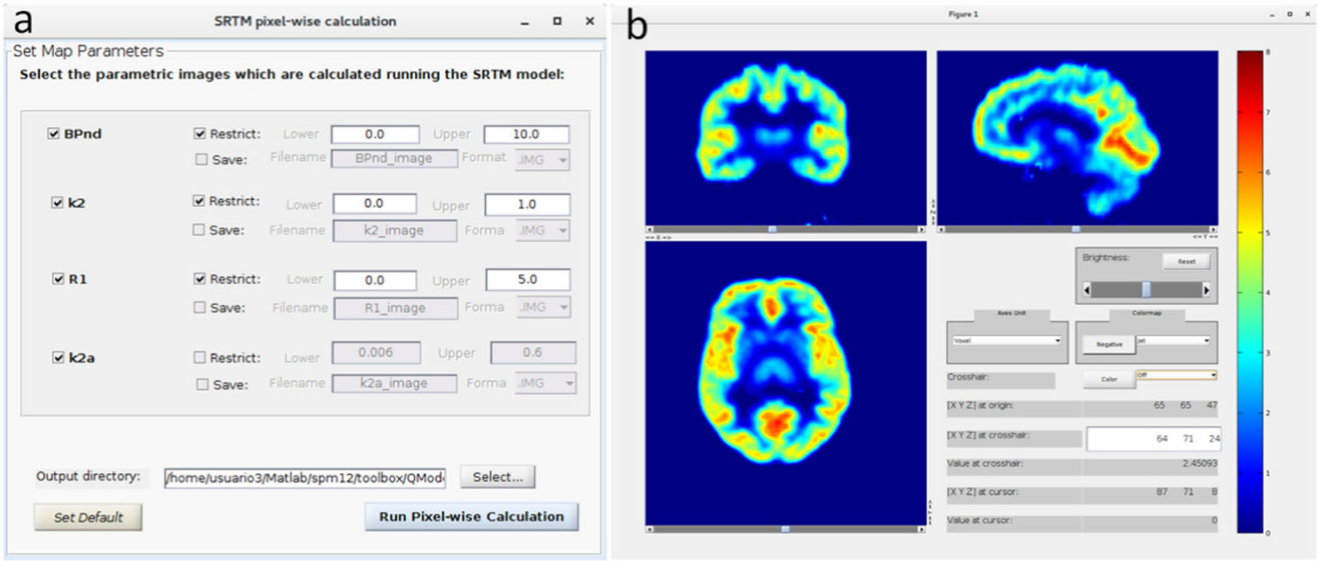

Fig. 4 Options selection window for image generation (**a**) and image viewer (**b**). After preprocessing, the options window is opened, allowing the user to restrict parameter values and to select a name and an image saving format. Then the image viewer can be used to display the images with different viewing and saving options

find detailed information on the way to use this option in a demo file included in the package.

In this way, several studies can be processed at a time, without the need of manual intervention. This option can be of use in studies including many subjects, and also to work with simulated data, in order, for instance, to evaluate quantification methods, or the program itself.

The command line functionality also provides advanced users with the possibility of adding new kinetic models in a relatively easy way. There is a specific section in the online manual ("Include your own model") indicating how to do it. If users are interested in running their new method also from the GUI, they can contact the authors for further advice.

Preliminary Validation and Performance Analysis

Validation

In Table 1 we can see the parameters obtained from the preprocessing for each model. The differences from PMOD are also shown. The relative differences obtained are in the orders of 10^{-12} to 10^{-16}, i.e. very small, in SRTM, SRTM2 with fixed k_2', Patlak and Logan. Bigger differences are found for SRTM2 when k_2' is left free. When k_2' is fixed in SRTM2, the fitting process is similar to SRTM, whereas if k_2' needs to be estimated, QModeling uses *bf* following the procedure described by the authors of the method (Wu and Carson 2002), whereas PMOD generates it by applying the Levenberg-Marquardt algorithm to the SRTM model. This explains the relative differences found in this case.

Table 2 shows the differences obtained with respect to PMOD in the calculation of the parametric images. The relative and absolute errors between the maximum values of the parametric images are shown for the four models, as well as between the mean values of these images. Once more, SRTM2 has been evaluated twice: setting and not setting k_2'. In all cases except SRTM2 with unfixed k_2', relative errors are in the range of 10^{-8} to 10^{-11}. SRTM2 with estimated k_2' shows larger differences due to the use of Levenberg-Marquardt algorithm by PMOD in the preprocessing step. Unfortunately, in SRTM2, the $k_2$parameter image could not be compared because of a bug in the image generation for $k_2$in PMOD v.3.310, that has been corrected in later versions (PMOD Technologies 2016). This bug only applies to the $k_2$image generation, not to the model estimation step, so the rest of the parametric images are comparable.

The estimated parameters were almost identical to those of PMOD, both in the preprocessing step and in parametric images. Small differences can be due to differences of precision between programming languages or to the fact that some algorithms used in QModeling are different from those of PMOD. For instance, in all three models, it is necessary to use a multiple regression technique, and the technique used in QModeling is QR decomposition, while PMOD uses Singular Value Decomposition (SVD) (PMOD Technologies 2012).

Performance Analysis

The results of the performance evaluation of QModeling are summarized in Table 3. No distinctions are made between

Table 1 QModeling results and differences from PMOD for the preprocessing of the three models

Model	Parameters	QModeling values	Differences from PMOD	
			Absolute error	Relative error
SRTM	$k_2(min^{-1})$	0. 42± 0.09	$(2.8 \pm 0.8) \cdot 10^{-15}$	$(7 \pm 3) \cdot 10^{-15}$
	BP(1/1)	3.6 ± 0.4	$(3 \pm 3) \cdot 10^{-14}$	$(9 \pm 7) \cdot 10^{-15}$
	$R_1(1/1)$	1.25 ± 0.04	$(9.8 \pm 0.3) \cdot 10^{-15}$	$(7.9 \pm 0.3) \cdot 10^{-15}$
	$k_{2a}(min^{-1})$	0.09± 0.02	–	–
SRTM2 (k_2'fixed)	k_2'fixed(min^{-1})	0.1282	–	–
	BP(1/1)	4.3 ±0.6	$(2 \pm 2) \cdot 10^{-14}$	$(6 \pm 5) \cdot 10^{-15}$
	$R_1(1/1)$	1.7 ± 0.1	$(1 \pm 1) \cdot 10^{-14}$	$(6 \pm 5) \cdot 10^{-15}$
	$k_{2a}(min^{-1})$	0.041 ± 0.006	$(2 \pm 2) \cdot 10^{-16}$	$(4 \pm 4) \cdot 10^{-15}$
SRTM2	$k_2'(min^{-1})$	0.33 ± 0.07	$(3 \pm 2) \cdot 10^{-3}$	$(9 \pm 7) \cdot 10^{-3}$
	BP(1/1)	3.6 ± 0.4	$(7 \pm 6) \cdot 10^{-3}$	$(2 \pm 2) \cdot 10^{-3}$
	$R_1(1/1)$	1.25 ± 0.04	$(4 \pm 3) \cdot 10^{-3}$	$(3 \pm 2) \cdot 10^{-3}$
	$k_{2a}(min^{-1})$	0.09± 0.02	$(8 \pm 6) \cdot 10^{-4}$	$(8 \pm 6) \cdot 10^{-3}$
Patlak	t*(min)	9.4± 0.4	–	–
	$K(min^{-1})$	0.050± 0.008	$(2 \pm 3) \cdot 10^{-16}$	$(3 \pm 5) \cdot 10^{-15}$
	Intercept(1/1)	0.4± 0.4	$(5 \pm 4) \cdot 10^{-15}$	$(3 \pm 3) \cdot 10^{-14}$
Logan	BP(1/1)	3.8 ± 0.5	$(2 \pm 1) \cdot 10^{-13}$	$(4 \pm 3) \cdot 10^{-14}$
	Intercept(1/1)	−990± 200	$(5 \pm 5) \cdot 10^{-12}$	$(-5 \pm 5) \cdot 10^{-15}$

output parameters, since all parameters are obtained simultaneously in the matrix calculations performed by QModeling.

The execution times we present were obtained with the default values proposed in PMOD. Even though TAC generation with QModeling appears to be somewhat longer than with PMOD, this is mainly due to PMOD loading the studies in a previous step to TAC generation, whereas QModeling does both functions in this single step.

In the preprocessing step, no matter the model, QModeling is always faster than PMOD by a factor of

Table 2 Differences from PMOD obtained in the generation of parametric images of the three models

Model	Parameters	Maximum values		Averages	
		Absolute error	Relative error	Absolute error	Relative error
SRTM	k_2	$(7 \pm 3) \cdot 10^{-8}$	$(3 \pm 2) \cdot 10^{-8}$	$(1 \pm 2) \cdot 10^{-10}$	$(5 \pm 7) \cdot 10^{-10}$
	BP	$(2 \pm 2) \cdot 10^{-7}$	$(1 \pm 1) \cdot 10^{-8}$	$(3 \pm 1) \cdot 10^{-10}$	$(1 \pm 4) \cdot 10^{-11}$
	R_1	$(1 \pm 0.2) \cdot 10^{-6}$	$(5 \pm 1) \cdot 10^{-8}$	$(4 \pm 7) \cdot 10^{-10}$	$(4 \pm 7) \cdot 10^{-10}$
	k_{2a}	$2.4 \cdot 10^{-8}$	$4 \cdot 10^{-8}$	$(8 \pm 3) \cdot 10^{-10}$	$(6.8 \pm 0.6) \cdot 10^{-9}$
SRTM2 (k_2'fixed)	k_2	–	–	–	–
	BP	$(3 \pm 2) \cdot 10^{-7}$	$(3 \pm 1) \cdot 10^{-e8}$	$(3 \pm 4) \cdot 10^{-10}$	$(1 \pm 1) \cdot 10^{-10}$
	R_1	$(5 \pm 3) \cdot 10^{-7}$	$(4 \pm 2) \cdot 10^{-8}$	$(7 \pm 7) \cdot 10^{-11}$	$(5 \pm 5) \cdot 10^{-11}$
	k_{2a}	$2.4 \cdot 10^{-8}$	$4 \cdot 10^{-8}$	$(4 \pm 1) \cdot 10^{-10}$	$(9 \pm 3) \cdot 10^{-9}$
SRTM2	k_2	–	–	–	–
	BP	$(7 \pm 9) \cdot 10^{-3}$	$(7 \pm 9) \cdot 10^{-4}$	$(3 \pm 3) \cdot 10^{-3}$	$(1 \pm 1) \cdot 10^{-3}$
	R_1	$(7 \pm 5) \cdot 10^{-2}$	$(9 \pm 7) \cdot 10^{-3}$	$(4 \pm 3) \cdot 10^{-3}$	$(3 \pm 2) \cdot 10^{-3}$
	k_{2a}	$2.4 \cdot 10^{-8}$	$4 \cdot 10^{-8}$	$(8 \pm 6) \cdot 10^{-4}$	$(7 \pm 5) \cdot 10^{-3}$
Patlak	K	$(4 \pm 2) \cdot 10^{-9}$	$(5 \pm 2) \cdot 10^{-8}$	$(4 \pm 2) \cdot 10^{-11}$	$(8 \pm 5) \cdot 10^{-9}$
	Intercept	$(4 \pm 3) \cdot 10^{-8}$	$(3 \pm 3) \cdot 10^{-8}$	$(2 \pm 2) \cdot 10^{-9}$	$(9 \pm 8) \cdot 10^{-9}$
Logan	BP	$(1.6 \pm 0.7) \cdot 10^{-7}$	$(2 \pm 1) \cdot 10^{-8}$	$(2.1 \pm 0.8) \cdot 10^{-10}$	$(7 \pm 3) \cdot 10^{-11}$
	Intercept	$(2 \pm 1) \cdot 10^{-5}$	$(2 \pm 1) \cdot 10^{-8}$	$(6 \pm 2) \cdot 10^{-8}$	$(-6.8 \pm 0.5) \cdot 10^{-11}$

The absolute and relative errors between maximum values of parametric images are shown, as well as the difference in the mean values of these images

Table 3 Comparative analysis of execution time of each module with default settings (values shown in seconds)

Model	PC	Software	TACs	Preprocessing	Gen. Images
SRTM	1	QModeling	0.37±0.07	0.31±0.00	6.11±0.14
	2		0.45±0.09	0.26±0.02	7.90±0.19
	3		0.68±0.05	0.51±0.03	14.14±0.36
	4		2.48±0.39	0.71±0.02	70.36±1.31
	1	PMOD	0.23±0.05	1.03±0.05	6.57±0.23
	2		0.35±0.05	1.08±0.07	6.90±0.13
	3		0.34±0.05	0.97±0.14	9.10±0.31
	4		0.57±0.27	3.95±0.39	45.53±1.55
SRTM2	1	QModeling	0.37±0.07	0.32±0.02	5.80±0.07
	2		0.44±0.08	0.28±0.02	7.15±0.24
	3		0.69±0.05	0.53±0.01	12.08±0.16
	4		2.45±0.41	0.82±0.01	55.65±0.41
	1	PMOD	0.22±0.05	1.09±0.09	6.66±0.49
	2		0.36±0.07	1.15±0.15	6.98±0.08
	3		0.35±0.06	1.05±0.16	7.14±0.18
	4		0.59±0.29	4.29±0.70	11.68±6.15
Patlak	1	QModeling	0.78±0.00	0.20±0.01	0.47±0.02
	2		0.91±0.00	0.15±0.01	0.62±0.02
	3		1.42±0.00	0.21±0.01	0.79±0.03
	4		4.87±0.03	0.25±0.00	3.21±0.03
	1	PMOD	0.18±0.01	1.00±0.08	1.91±0.01
	2		0.24±0.05	1.12±0.09	1.72±0.01
	3		0.19±0.02	0.96±0.03	1.89±0.05
	4		0.65±0.15	6.26±0.15	5.14±0.04
Logan	1	QModeling	0.37±0.07	0.13±0.01	7.24±0.17
	2		0.44±0.08	0.11±0.01	7.03±0.28
	3		0.68±0.04	0.11±0.00	11.40±0.14
	4		2.46±0.40	0.15±0.01	56.20±0.27
	1	PMOD	0.21±0.04	0.98±0.11	6.37±0.25
	2		0.36±0.08	1.01±0.14	6.49±0.09
	3		0.35±0.07	0.96±0.20	6.73±0.16
	4		0.59±0.29	4.06±0.53	4.72±0.57

The analysis was separately performed for the generation of TACs, the preprocessing and the calculation of the parametric images

1.9 to 27.5. Times for QModeling are always below 0.82 s, whereas PMOD needs between 0.96 s and 6.26 s, depending on the method and the computer. The difference is highest for the Logan model.

For image generation, we obtain a wide range of execution times, depending on the kinetic model and the computer used. For SRTM the execution time is slightly better for QModeling (7%) in the fastest computer (PC 1), although PMOD is somewhat faster (13%) in the other Ubuntu, PC 2, and 43% faster in the slowest PCs (3 and 4). Total times range from 6 s to 70s. The results for SRTM2 are similar, with QModeling yielding slightly better results (14%) for the fastest PC, very similar for the other Ubuntu, and PMOD being faster in the other two PCs, particularly PC4 (56 s QModeling vs. 11 s PMOD).QModeling is always faster than PMOD for the Patlak model, by 46 to 120%, in execution times ranging from 0.47 s (QModeling, PC 1) to 5.14 s (PMOD, PC 4).

Overall, the computation times in QModeling allow using the program in reasonable times even for slow computers, and its performance equals or even surpasses that of PMOD in more up-to-date systems.

The differences can be due to a variety of factors, such as programming language (QModeling is developed in MATLAB and PMOD in Java), differences in algorithms, and different adaptation of each environment to the available hardware.

Table 4 shows the preprocessing execution times when varying the input parameters. The number of *bf* and the resampling interval affect the computation time in QModeling, since they increase or decrease the number of calculations. PMOD does not show any changes when the Resampling values are varied and is little affected by the number of *bf*. Even though the computation times have greater variations for QModeling, the preprocessing times obtained with our software continue to be better than those of PMOD. Since we obtained only irrelevant changes in execution time whenk_{2a}-

Table 4 Comparative analysis of preprocessing execution time and its dependence on the input parameters (values shown in seconds)

Preprocessing							
Model	PC	Software	10 *bf*	400 *bf*	Resmp. 1	Resmp. 2	Default values
SRTM	1	QModeling	0.24±0.02	0.46±0.03	0.38±0.00	0.33±0.02	0.31±0.00
	2		0.17±0.01	0.42±0.01	0.35±0.02	0.28±0.01	0.26±0.02
	3		0.20±0.00	1.18±0.00	0.59±0.00	0.52±0.00	0.51±0.03
	4		0.35±0.01	1.88±0.03	1.77±0.05	0.94±0.02	0.71±0.02
	1	PMOD	1.00±0.12	1.10±0.09	1.05±0.11	1.02±0.11	1.03±0.05
	2		1.01±0.11	1.15±0.13	1.05±0.12	1.07±0.10	1.08±0.07
	3		0.96±0.13	1.00±0.13	0.95±0.15	0.94±0.13	0.97±0.14
	4		3.87±0.43	4.18±0.43	3.90±0.46	3.95±0.46	3.95±0.39
SRTM2	1	QModeling	0.25±0.01	0.50±0.01	0.44±0.01	0.35±0.01	0.32±0.02
	2		0.18±0.01	0.49±0.02	0.44±0.02	0.31±0.01	0.28±0.02
	3		0.20±0.00	1.29±0.00	0.72±0.00	0.58±0.00	0.53±0.01
	4		0.34±0.00	2.41±0.02	2.83±0.01	1.29±0.01	0.82±0.01
	1	PMOD	1.11±0.07	1.26±0.09	1.22±0.11	1.16±0.15	1.09±0.09
	2		1.10±0.13	1.27±0.14	1.23±0.17	1.15±0.16	1.15±0.15
	3		1.02±0.21	1.06±0.16	1.22±0.24	1.08±0.20	1.05±0.16
	4		4.01±0.53	4.42±0.53	4.84±0.62	4.53±0.51	4.29±0.70
Model	PC	Software	t* = 5	t* = 10	t* = 20	Default values	
Patlak	1	QModeling	0.22±0.02	0.22±0.00	0.22±0.01	0.20±0.01	
	2		0.18±0.01	0.17±0.01	0.16±0.01	0.15±0.01	
	3		0.25±0.00	0.25±0.01	0.25±0.01	0.21±0.01	
	4		0.22±0.01	0.20±0.01	0.22±0.01	0.25±0.00	
	1	PMOD	0.94±0.06	0.91±0.04	0.90±0.03	1.00±0.08	
	2		1.05±0.10	1.07±0.06	1.05±0.10	1.12±0.09	
	3		0.96±0.10	0.96±0.07	0.96±0.04	0.96±0.03	
	4		6.17±0.06	6.21±0.15	6.18±0.15	6.26±0.15	
Logan	1	QModeling	0.13±0.01	0.12±0.01	0.13±0.01	0.13±0.01	
	2		0.09±0.03	0.08±0.02	0.08±0.02	0.11±0.01	
	3		0.10±0.01	0.10±0.00	0.10±0.00	0.11±0.00	
	4		0.12±0.01	0.12±0.01	0.12±0.00	0.15±0.01	
	1	PMOD	0.99±0.10	0.96±0.10	0.96±0.10	0.98±0.11	
	2		0.99±0.11	1.00±0.06	0.99±0.06	1.01±0.14	
	3		0.95±0.16	0.87±0.07	0.86±0.10	0.96±0.20	
	4		3.96±0.58	3.98±0.59	4.01±0.57	4.06±0.53	

was changed (3% or smaller), the results were not included in the table. For SRTM2, working with a fixed or unfixed value of k_2' had no impact on these results, so the particular obtained values are not shown either.

In the case of Patlak and Logan on QModeling, fixing t^* implies a very slight decrease on computation time, which does not change much the differences between the two softwares. Fixing t^* at different points did not affect the results with either one.

Discussion

QModeling is a new software for the kinetic analysis of PET studies and the generation of parametric images. It is free, with the only limitation of working on MATLAB. It is easy to use, open-source and multiplatform. As an SPM toolbox, it facilitates the incorporation of kinetic modeling into broader neuroimaging protocols. This also allows the user to adapt this preprocessing to the available data in a flexible way. Moreover, QModeling is also applicable to non-neurological studies, including human and small-animal imaging.

The clarity and simplicity of the GUI make our software a friendly program, allowing a quick and easy navigation. A fully detailed manual is also provided to resolve any difficulties or doubts. The multiplatform character of this toolbox provides versatility, and, as an open source program, it can be modified by users to adapt it to their needs. Furthermore, good modularization makes it easily extensible with new pharmacokinetic models. It also offers the possibility to work from the MATLAB command line, which simplifies processing studies with many subjects and working with simulated data.

The preliminary validation and performance tests have allowed us to see that QModeling offers reliable results. Both at the preprocessing and at the voxel levels, the obtained parameters are very close to those of PMOD, with relative

differences in the order of 10^{-12} or lower in preprocessing results, for almost all parameters. The only exception is SRTM2 with unfixed k_2', where different numeric methods to estimate of this parameter lead to relative differences of the order of 10^{-3} or 10^{-4} in the fitted parameters. Similar results were obtained when evaluating the entire parametric images. Simulation studies could help determine the most numerically precise method for this case. However, differences of this order should not lead to relevant changes in the conclusions in kinetic modeling studies performed with either software. If parametric images are to be visually evaluated, the differences will be imperceptible, and if group statistics are calculated, physiological variability and PET image degrading phenomena will strongly prevail over this minimal effect. The obtained results, therefore, support the correctness of all results, and indicate that it is useful for biomedical research.

The obtained runtime performance shows that the user can do all kinetic analyses efficiently, as execution times are, overall, of the same magnitude as those of PMOD. Results for SRTM and SRTM2 showed similar values for the fastest computer. Even though PMOD showed a somewhat quicker performance in the slower workstations, QModeling provided a smooth processing pipeline as well. On the other hand, QModeling had better execution times for the Patlak model. Changes in the input parameters caused only small runtime variations, both in PMOD and in QModeling.

QModeling is an open project that can be improved in the future by adding new models, including databases, optimizing algorithms, etc., in order to offer an increasingly complete service to the user.

Conclusion

QModeling is a new application for PET pharmacokinetic analysis. It is free and easy to use, in order to facilitate research in the clinical context. It allows applying SRTM, SRTM2 Patlak Reference and Logan Reference models to dynamic PET studies with reliable results in efficient computation times. As a toolbox for SPM, it facilitates an easy inclusion of kinetic modeling into flexible neuroimaging pipelines, but it is also applicable to research in non-neurological applications. Besides, QModeling is a free, multiplatform, easy-to-use and open-source tool which offers the possibility of being expanded and enhanced with new models.

Information Sharing Statement

QModeling (RRID:SCR_016358) is available to download from: http://www.uimcimes.es/contenidos/golink?p=1. The toolbox is free but copyright software, distributed under the terms of the GNU General Public Licence as published by the Free Software Foundation. QModeling is written for MATLAB R2012a and onwards, and needs an installed version of SPM (version 8 or above) to work. To install our toolbox, the user has to copy QModeling folder into the SPM toolbox directory. To run the software, the user can do it from SPM GUI or from the MATLAB command line. For further information on how to use the software, please see the QModeling online manual: http://www.uimcimes.es/contenidos/golink?p=2.

Acknowledgements Francisco J. López-González is funded by a grant (PTA2014-09677-I) from the Spanish Ministry of Economy, Industry and Competitiveness under the Technical Support Staff (PTA) program.

Karl Thurnhofer-Hemsi and José Paredes-Pacheco are funded by PhD scholarships (FPU15/06512 and FPU16/05108, respectively) from the Spanish Ministry of Education, Culture and Sport under the FPU program.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

References

Bartmann, H., Fuest, C., La Fougere, C., Xiong, G., Just, T., Schlichtiger, J., Winter, P., Böning, G., Wängler, B., Pekcec, A., Soerensen, J., Bartenstein, P., Cumming, P., & Potschka, H. (2010). Imaging of P-glycoprotein–mediated pharmacoresistance in the hippocampus: Proof-of-concept in a chronic rat model of temporal lobe epilepsy. *Epilepsia, 51*, 1780–1790.

Bullich, S., Cot, A., Gallego, J., Gunn, R. N., Suárez, M., Pavía, J., Ros, D., Laruelle, M., & Catafau, A. M. (2010). Impact of scatter correction on D2 receptor occupancy measurements using 123I-IBZM SPECT: Comparison to 11C-Raclopride PET. *NeuroImage, 50*(4), 1511–1518.

Burger, C., & Buck, A. (1997). Requirements and implementation of a flexible kinetic modeling tool. *J Nucl Med, 38*(11), 1818–1823.

Coello, C., Fisk, M., Mohan, D., Wilson, F. J., Brown, A. P., Polkey, M. I., Wilkinson, I., Tal-Singer, R., Murphy, P. S., Cheriyan, J., & Gunn, R. N. (2017). Quantitative analysis of dynamic 18F-FDG PET/CT for measurement of lung inflammation. *EJNMMI Res, 7*(1), 47.

Fang, Y.-H. D., Asthana, P., Salinas, C., Huang, H.-M., & Muzic, R. F. (2010). Integrated software environment based on COMKAT for analyzing tracer pharmacokinetics with molecular imaging. *J Nucl Med, 51*(1), 77–84.

Frankle, W. G., Slifstein, M., Gunn, R. N., Huang, Y., Hwang, D.-R., Darr, E. A., Narendan, R., Abi-Dargham, A., & Laruelle, M. (2006). Estimation of serotonin transporter parameters with 11C-DASB in healthy humans: Reproducibility and comparison of methods. *J Nucl Med, 47*(5), 815–826.

Grotegerd, D., Redlich, R., Almeida, J. R. C., Riemenschneider, M., Kugel, H., Arolt, V., & Dannlowski, U. (2014). MANIA—A pattern classification toolbox for neuroimaging data. *Neuroinformatics, 12*(3), 471–486.

Gunn, R. N., Lammertsma, A. A., Hume, S. P., & Cunningham, V. J. (1997). Parametric imaging of ligand-receptor binding in PET using a simplified reference region model. *NeuroImage, 6*(4), 279–287.

Gunn, R., Coello, C., & Searle, G. (2016). Molecular imaging and kinetic analysis toolbox (MIAKAT) - a quantitative software package for

the analysis of PET neuroimaging data. *J Nucl Med, 57*-(supplement_2), 1928.

Keator, D. B., Gadde, S., Grethe, J. S., Taylor, D. V., & Potkin, S. G. (2006). A general XML schema and SPM toolbox for storage of neuro-imaging results and anatomical labels. *Neuroinformatics, 4*(2), 199–212.

Keramida, G., Anagnostopoulos, C. D., & Peters, A. M. (2017). The extent to which standardized uptake values reflect FDG phosphorylation in the liver and spleen as functions of time after injection of 18F-fluorodeoxyglucose. *EJNMMI Res, 7*(1), 13.

Klumpers, U. M. H., Veltman, D. J., Boellaard, R., Comans, E. F., Zuketto, C., Yaqub, M., & Lammertsma, A. A. (2007). Comparison of plasma input and reference tissue models for Analysing [11C]flumazenil studies. *J Cereb Blood Flow Metab, 28*(3), 579–587.

Lammertsma, A. A., & Hume, S. P. (1996). Simplified reference tissue model for PET receptor studies. *NeuroImage, 4*(3), 153–158.

Lammertsma, A. A., Bench, C. J., Hume, S. P., Osman, S., Gunn, K., Brooks, D. J., & Frackowiak, R. S. (1996). Comparison of methods for analysis of clinical [11C]raclopride studies. *J Cereb Blood Flow Metab, 16*(1), 42–52.

Logan, J., Fowler, J. S., Volkow, N. D., Wang, G. J., Ding, Y. S., & Alexoff, D. L. (1996). Distribution volume ratios without blood sampling from graphical analysis of PET data. *J Cereb Blood Flow Metab, 16*(5), 834–840.

Lopes, I., Vállez, D., Parente, A., Doorduin, J., Dierckx, R., Marques da Silva, A., Koole, M., Willemsen, A., & Boellaard, R. (2017). Pharmacokinetic modeling of [^{11}C]flumazenil kinetics in the rat brain. *EJNMMI Res, 7*(17), 1–12.

Maldjian, J. A., Laurienti, P. J., Kraft, R. A., & Burdette, J. H. (2003). An automated method for neuroanatomic and cytoarchitectonic atlas-based interrogation of fMRI data sets. *NeuroImage, 19*(3), 1233–1239.

Muzic, R. F. J., & Cornelius, S. (2001). COMKAT: Compartment model kinetic analysis tool. *J Nucl Med, 42*(4), 636–645.

Oikonen, V., Johansson, J., Liukko, K., Merisaari, H., Alenius, S., Laakkonen, C., & Krais, R. (n.d.). TPCCLIB source code documentation. Turku PET Centre. http://www.turkupetcentre.net/petanalysis/tpcclib/tpcclib-doc/index.html. Accessed 10 July 2017.

Olsson, H., Halldin, C., & Farde, L. (2004). Differentiation of extrastriatal dopamine D2 receptor density and affinity in the human brain using PET. *NeuroImage, 22*(2), 794–803.

Patlak, C. S., & Blasberg, R. G. (1985). Graphical evaluation of blood-to-brain transfer constants from multiple-time uptake data. Generalizations. *J Cereb Blood Flow Metab, 5*(4), 584–590.

Patlak, C. S., Blasberg, R. G., & Fenstermacher, J. D. (1983). Graphical evaluation of blood-to-brain transfer constants from multiple-time uptake data. *J Cereb Blood Flow Metab, 3*(1), 1–7.

PMOD Technologies. (2012). PMOD Pixel-wise Modeling (PXMOD) Version 3.4.Resource document. PMOD Technologies. http://www.pmod.com/files/download/v34/doc/PDF/PXMOD.pdf. Accessed10 July 2017.

PMOD Technologies (2016). PMOD Software Release Notes Version 3.6. Maintenance Builds of Release 3.5, p. 11. Resource document. PMOD Technologies. http://www.doc36.pmod.com/PDF/ReleaseNotes.pdf. Accessed 10 July 2017.

Rodriguez-Vieitez, E., Leuzy, A., Chiotis, K., Saint-Aubert, L., Wall, A., & Nordberg, A. (2017). Comparability of [^{18}F]THK5317and [^{11}C]PIB blood flow proxy images with [^{18}F]FDG positron emission tomography in Alzheimer's disease. *J Cereb Blood Flow Metab, 37*(2), 740–749.

Salinas, C. A., Searle, G. E., & Gunn, R. N. (2015). The simplified reference tissue model: Model assumption violations and their impact on binding potential. *J Cereb Blood Flow Metab, 35*, 304–311.

Schain, M., Fazio, P., Mrzljak, L., Amini, N., Al-Tawil, N., Fitzer-Attas, C., & Varrone, A. (2017). Revisiting the Logan plot to account for non-negligible blood volume in brain tissue. *EJNMMI Res, 7*(1), 66.

Schrouff, J., Rosa, M. J., Rondina, J. M., Marquand, A. F., Chu, C., Ashburner, J., Philips, C., Richiardi, J., & Mourão-Miranda, J. (2013). PRoNTo: Pattern recognition for neuroimaging toolbox. *Neuroinformatics, 11*(3), 319–337.

Sérrière, S., Tauber, C., Vercouillie, J., Guilloteau, D., Deloye, J., Garreau, L., Galineau, L., & Chalon, S. (2014). In vivo PET quantification of the dopamine transporter in rat brain with [18F]LBT-999. *Nucl Med Biol, 41*(1), 106–113.

Shen, J. (2014). Tools for NIfTI and ANALYZE image. File Exchange MathWorks. https://es.mathworks.com/matlabcentral/fileexchange/8797-tools-for-nifti-and-analyze-image. Accessed 10 July 2017.

Tabelow, K., Mohammadi, S., Weiskopf, N., & Polzehl, J. (2015). POAS4SPM: A toolbox for SPM to Denoise diffusion MRI data. *Neuroinformatics, 13*(1), 19–29.

TKMF Program (n.d.). Department of Molecular & Medical Pharmacology.UCLA School of Medicine. https://dragon.nuc.ucla.edu/modelfitting/modelfit.htmlAccessed 10 July 2017.

Wienhard, K. (2002). Measurement of glucose consumption using [18F]fluorodeoxyglucose. *Methods, 27*(3), 218–225.

Wu, Y., & Carson, R. E. (2002). Noise reduction in the simplified reference tissue model for neuroreceptor functional imaging. *J Cereb Blood Flow Metab, 22*(12), 1440–1452.

Yo-Han, J., Jeong-Hee, K., Young-Don, S., Hang-Keun, K., Yeon-Jeong, S., Sang-Yoon, L. & Jong-Hoon, K. (2017). The relationship between excitement symptom severity and extrastriatal dopamine $D_{2/3}$ receptor availability in patients with schizophrenia: a high-resolution PET study with [^{18}F] fallypride. *European Archives of Psychiatry and Clinical Neurosciences*, 1–12.