

Determination of the Blood Input Function for Dynamic Emission Tomography with Deep Learning

Dóra Varnyú and László Szirmay-Kalos

Department of Control Engineering and Information Technology,
Budapest University of Technology and Economics (e-mail: szirmay@iit.bme.hu)

Abstract

Dynamic tomography reconstructs a Time Activity Curve (TAC) for every voxel assuming that the algebraic form of the function is known a-priori. Practical kinetic models include the blood input function that may be scaled, integrated, derivated or convolved during the calculation of kinetic parameters. In direct dynamic tomography, the kinetic parameter calculation and reconstruction happen simultaneously, which either requires the blood input function from the beginning or obtains the kinetic parameters and the blood input function simultaneously as a joint optimization. In the latter case, we still need the approximation of the blood input function to serve as a starting point for the iterative optimization. This paper proposes a deep-learning-based approach to find a good guess for the non-linear blood input parameters from which the iterative process can start its refinement.

1. Introduction

With dynamic Positron Emission Tomography, we can determine the dynamics of radiotracer spreading and accumulation in tissues. The determined TAC function is expected to be consistent with biology, therefore it is searched in a pre-defined algebraic form that reflects the biological model.

The most popular approach is the *compartmental analysis*^{8, 19, 20, 18}, in which the algebraic form is based on the solution of differential equations expressing the tracer exchange between n compartments⁶. The solution has an algebraic form that is a non-negative linear combination of convolutions of the *blood input function* $C_p(t)$ and the *impulse response* $w(t)$ of the tissue. The impulse response is the non-negatively weighted sum of exponentials of unknown, non-negative *rate constants* α_i and constrained only for positive time values t by the Heaviside step function $\varepsilon(t)$:

$$w(t) = \varepsilon(t) \sum_{i=1}^n c_i e^{-\alpha_i t},$$

where c_i are the non-negative weights. Taking into account that a voxel is a mixture of tissue and blood, we obtain

$$\mathcal{K}(\mathbf{p}, t) = f_v C_W(t) + (1 - f_v) w(t) * C_p(t), \quad (1)$$

where $f_v \in [0, 1]$ is the unknown *fraction of blood* and C_W is the *total blood concentration function* taking into account

also the portion of the radiotracer that cannot diffuse into the tissues. The blood input function for tissues, $C_p(t)$, may be different from the radioactivity of the whole blood, $C_W(t)$, because metabolism is limited. The two functions may be fitted separately, or we can assume that one of them can be obtained from the other with a tracer-dependent metabolic correction. In this paper, we assume that a known tracer-dependent factor γ can be used to express total blood concentration $C_W(t)$ from the blood input $C_p(t)$ as $C_W(t) = \gamma C_p(t)$.

In PET, we use radiotracer isotopes decaying with positron emission. The positron emitted at the decay of an isotope wanders a few millimeters in the tissues and then collides with an electron. As a result, both particles annihilate, and two oppositely directed gamma-photons are born, which might be detected by the tomograph. A PET/CT system collects the *events* of simultaneous photon incidents in detector pairs. An event is a composition of the identification of the detector pair, also called *Line Of Response* or *LOR*, and its time of occurrence. The raw input data of the reconstruction is the list of events, which is processed directly in list mode, or binned in *frames* (t_F, t_{F+1}) .

Using the concentration function $\mathcal{K}(\mathbf{p}_V, t)$ of voxel V , the expected number of radioactive decays, i.e. number of

positrons generated in voxel V in frame $[t_F, t_{F+1}]$ is

$$\tilde{x}_F(\mathbf{p}_V) = \int_{t_F}^{t_{F+1}} \mathcal{K}(\mathbf{p}_V, t) \exp(-\lambda t) dt, \quad (2)$$

where λ is the decay rate of the radiotracer.

The correspondence between positron generation and gamma-photon detection is established by *system matrix* $\mathbf{A}$ that expresses the probability of generating an event in LOR L given that a positron is emitted in voxel V . The expected number of events $\tilde{y}_{L,F}$ in LOR L during frame F , i.e. in time interval (t_F, t_{F+1}) , is the sum of the contributions of all voxels in the volume during this time:

$$\tilde{y}_{L,F} = \sum_{V=1}^{N_V} \mathbf{A}_{L,V} \tilde{x}_F(\mathbf{p}_V). \quad (3)$$

As a result, the measured values in LOR L and frame F is of Poisson distribution with mean $\tilde{y}_{L,F}$.

Note that the blood input function is scaled, convolved and integrated until it conforms to the expected value of the measured data, which is just a realization of a random variable, or from another point of view, it is also contaminated by noise. From the values of LORs and frames, we need to determine the global blood input parameters as well as the compartment model parameters of every voxel.

In direct dynamic tomography, kinetic parameter calculation and reconstruction happen simultaneously, thus these functions or their approximations are needed from the start of the process. Possible methods to obtain the blood input function include the measurement of the real data, the application of the average of similar measurements, and reconstructing the blood input function also from the PET data.

This paper investigates the problem of determining the blood input function, aiming at making estimations more reliable without taking blood samples. Our approach is in between image-derived and model-derived approaches. Similarly to image-derived methods, we do not require physically measured samples, but unlike image-derived techniques, our method works robustly without precise segmentation of the blood cells.

Reconstruction means the identification of parameters $\mathbf{p} = (f_v, c_1, \alpha_1, c_2, \alpha_2, \dots)$ for every voxel. However, the algebraic form of Eq. (1) is not optimal for fitting since the TAC depends on these parameters non-linearly. Thus, we use an equivalent algebraic form, where $f_v^*, a_1, a_2, \dots$ are linear parameters if the rate constants are fixed¹⁵:

$$\mathcal{K}(\mathbf{p}, t) = f_v^* C_p(t) + \left(\varepsilon(t) \sum_{i=1}^n a_i e^{-\alpha_i t} \right) * C_p(t), \quad (4)$$

where the correspondence between the new and the original parameters is $a_i = (1 - f_v) c_i$ and $f_v^* = \gamma f_v$.

The blood input function determination from the reconstructed data is an inverse problem related to this equation.

We have $\mathcal{K}(t)$ or its integrals in various intervals, and from these function $C_p(t)$ needs to be recovered. However, this inverse problem is ill-posed and ambiguous, since the same function can be delivered with many different parameters and blood input functions. For example, the contribution of the fraction of blood parameter f_v^* can also be provided with making $\alpha_i \rightarrow \infty$ and $a_i e^{-\alpha_i t} \rightarrow f_v^* \delta(t)$. Thus, we need more a priori information, e.g. that the dynamics of the blood input function is much faster than the tissue response, or we can use a simplifying assumption, such as $f_v^* = 0$.

Note also that the automatic determination of the blood input function is a “chicken and egg” type problem. In order to find the TAC of voxels from which the blood input function can be extracted, we need a tomographic reconstruction. During direct tomographic reconstruction, we fit the TAC in every iteration, which requires the blood input function. We can break this evil loop by indirect reconstruction, which reconstructs frames independently, thus requiring no parameter fitting. However, with this we lose the advantages of direct reconstruction, namely that for the reconstruction of a frame, complete information of all frames can be exploited.

2. Previous work

There are different possibilities for the determination of blood parameters^{22, 21, 4}. We can take blood samples from the artery. Artery samples represent the total blood function, from which the blood input function is calculated with tracer-dependent metabolic correction. Alternatively, venous samples can be taken during transient equilibrium.

Blood sampling is a difficult and error-prone process, especially in pre-clinical testing where small animals are investigated. Thus, there is a significant interest in methods requiring no, or just minimal number of such samples.

2.1. Population-Based Input Function

This approach takes the average of previously taken blood input functions and scales it based on just a few current samples. Therefore, this method does not eliminate the need of sampling, only drastically reduces the number of required samples. On the other hand, the shape of the blood input function is fixed, which may not prove to be flexible enough to describe the current case.

2.2. Image-Derived Input Function (IDIF)

The *Image Derived Input Function* (IDIF) method uses the reconstructed time-activity values of blood voxels, thus it requires the segmentation of the artery or the needle region. This method can determine the total blood function from which the blood input is calculated with metabolic correction. If the segmented region is thin or small, partial volume effect and spill-in may compromise the estimation, which is

described by the following model ¹:

$$C_m(t) = C_W(t)R + C_t(t)S, \quad (5)$$

where $C_m(t)$ is the activity in the artery that results from the reconstruction, $C_W(t)$ is the real total blood concentration, $C_t(t)$ is the obtained activity in surrounding tissues, R is the recovery coefficient, and S is the spill-in coefficient.

The task is to obtain $C_W(t)$ from $C_m(t)$ and $C_t(t)$ when the recovery coefficient and the spill-in factor are also unknown. This problem is ill-posed in this form. For example, Eq. 5 contains the product of the recovery coefficient and the total blood activity, which is invariant to the simultaneous upscaling of the recovery coefficient and downscaling of the blood activity. To eliminate these ambiguities, additional information is needed.

The recovery coefficient and the spill-in factor may be estimated from the point spread function and the precise segmentation of the arteries, but this approach is complicated and not very accurate. Another option is to first extrapolate $C_W(t)$ from late venous samples or by selecting the maximum intensity voxels, then obtain the recovery coefficient and the spill-in factor with least squares estimation. Having them, the total blood input function is expressed from Eq. 5.

The method of Image-Derived Input Function requires the segmentation of blood cells and possibly the compensation of partial volume effects. Note that in case of small animals, the number of voxels that can be clearly identified as full blood is limited.

A sophisticated approach to separate real blood input and the contribution from the neighboring tissues is the application of *Independent Component Analysis* ¹¹. This is done by assuming that two non-Gaussian, statistically independent components are added. Recalling the Central Limit Theorem, when two non-Gaussian, independent signals are added, the result will be closer to the Gaussian distribution than either of the two signals. Additionally, the components are expected to be of lower complexity than their sum.

Independent Component Analysis also suffers from indeterminacy of the scaling factors, thus real blood samples are needed to scale the reconstructed functions.

2.3. Model-Based Input Function (MBIF)

The *Model-Based Input Function* method does not require anatomic information or the segmentation of the arteries. Instead, it estimates the blood input function and the kinetic parameters simultaneously. The MBIF method searches not only the impulse response, but also the blood input function in a pre-defined algebraic form.

A proven choice for the algebraic form of $C_p(t)$ is the

Feng model ⁵:

$$C_p(t) = A_1(t - \tau)e^{-\beta_1(t-\tau)} + \sum_{j=2}^4 A_j \left(e^{-\beta_j(t-\tau)} - e^{-\beta_1(t-\tau)} \right) \quad (6)$$

if $t > \tau$, the time of injection, and zero otherwise.

Note that model-based methods also have ambiguity. The same Time Activity Curve can be obtained by scaling down the tissue model parameters a_i and f_v^* , and simultaneously scaling up the linear blood parameters $A_1, \dots, A_4$. This can be resolved by taking a few blood samples.

3. Proposed method

Our proposed algorithm belongs to the category of both Image-Derived Input Functions, as it needs the identification of “blood voxels” that at least partially contain artery regions, and Model-Based Input Functions, since we find the parameters of the Feng model. Neither full nor total segmentation is needed, but the accuracy of the method increase if the selection includes more blood voxels. Selected blood voxels are assumed to have similar activity functions.

To address the “chicken and egg” problem, the TAC and the blood input function are estimated in parallel during the iterative reconstruction. The reconstruction should simultaneously determine the parameters of the Feng model and the per-voxel kinetic parameters. In order to guarantee fast and certain convergence, a crucial factor is the initialization of these unknown parameters. The measured data is a weighted sum of the TAC of different voxels, which are in turn a weighted sum of the convolution of the impulse responses and the blood input function. It means that the blood input function is involved in every voxel TAC and therefore in every LOR after convolutions with voxel-dependent impulse response functions. Our goal is to exploit this information and to obtain the non-linear $\beta_1, \dots, \beta_4$ parameters directly, which can be used to initialize the iterative reconstruction process.

4. Analysis of the dependence of the TAC and the sinograms on the blood input parameters

Calculating voxel activity requires the integration of the convolutions of exponential functions.

According to Eq. 2, the activity of a voxel in a time frame is expressed by a time integral. The antiderivative of the integrand is

$$I(t) = \int \left(\left(\varepsilon(t) \sum_{i=1}^n a_i e^{-\alpha_i t} \right) * C_p(t) \right) e^{-\lambda t} dt = \sum_{i=1}^n a_i \left(A_1 H_i(t) + \sum_{j=2}^4 A_j (G_{ij}(t) - G_{i1}(t)) \right), \quad (7)$$

in which there are two types of convolutions from the blood

input function and the impulse response. Using the $\alpha_i^* = \alpha_i + \lambda$ and the $\beta_j^* = \beta_j + \lambda$ notations, the first type is

$$G_{ij}(t) = \int \left(\varepsilon(t) e^{-\alpha_i t} * \varepsilon(t) e^{-\beta_j t} \right) e^{-\lambda t} dt = \frac{e^{-\alpha_i^* t} / \alpha_i^* - e^{-\beta_j^* t} / \beta_j^*}{(\alpha_i^* - \beta_j^*)}, \quad (8)$$

while the second one is

$$H_i(t) = \int \left(\varepsilon(t) e^{-\alpha_i t} * \varepsilon(t) t e^{-\beta_1 t} \right) e^{-\lambda t} dt = \frac{e^{-\beta_1^* t} / \beta_1^* - e^{-\alpha_i^* t} / \alpha_i^* - (\alpha_i^* - \beta_1^*)(t + 1 / \beta_1^*) e^{-\beta_1^* t} / \beta_1^*}{(\alpha_i^* - \beta_1^*)^2}. \quad (9)$$

Finally, voxel activity in a time frame is

$$\tilde{x}_F(\mathbf{p}) = f_v \int_{t_F}^{t_{F+1}} C_W(t) dt + I(t_{F+1}) - I(t_F). \quad (10)$$

We can observe that every TAC is the weighted sum of functions $t e^{-\beta_1^* t}, e^{-\beta_1^* t}, \dots, e^{-\beta_4^* t}$, and also contains exponentials $e^{-\alpha_i^* t}$. The LOR data, which are the primary measured data, are the weighted sums of the TACs of the voxels. Thus, each data element is also a weighted sum of functions $t e^{-\beta_1^* t}, e^{-\beta_1^* t}, \dots, e^{-\beta_4^* t}$, and also contains a large mixture of exponentials $e^{-\alpha_i^* t}$, which is different in every LOR. In addition, we can assume that the dynamics of the blood input is quicker than the diffusion into the tissues, i.e. $\beta_j \gg \alpha_i$. The task is then to retrieve the exponential functions that are present in every measured data element and initialize the exponents of the blood input function accordingly.

If we had sine functions instead of exponentials, the solution would be quite straightforward. We would compute the Fourier transform of the signal and identify the largest dominant frequencies as parameters $\beta_1, \dots, \beta_4$. Unfortunately, this concept is not directly applicable for exponentials since these functions are not orthogonal. Fitting with exponentials have been investigated^{9,3}, but these are complicated and cannot handle the $t e^{-\beta_1^* t}$ type terms. Consequently, we should attack this problem with more general approach, which is the application of neural network.

Note that we do not aim at precise identification of the exponents, only at a reasonable guess that can be used to initialize the iterative reconstruction.

5. Using a neural network to predict initial values for optimization

In our work, we have constructed and trained a CNN-LSTM network to predict the exponents of the blood input function from series of time-activity data. The network was implemented using the Keras deep learning API² and hyperpa-

rameter optimization was performed with the Ray Tune library¹².

5.1. Dataset

The model was trained with a series of simulated time-activity data obtained by evaluating Eq. 4 with randomized parameters, assuming $n = 2$ compartments. Exponents of the blood input function were selected uniformly in range $[2, 20]$, requiring the largest of the four to be at least 5 to obtain TAC curves that subside within the measurement time, and assumed that $\beta_1 > \beta_2 > \beta_3 > \beta_4$ (Note that this corresponds to the operation of real biochemical processes, as the rate of absorption into tissues is greater than that of back-diffusion and $\beta_1 > 5$ guarantees that the shape of the corresponding exponential is captured during the considered time interval). The A_1, A_2, A_3, A_4 linear parameters of the blood input function were sampled with standard uniform distribution. For each octet of $(A_1, \beta_1, A_2, \beta_2, A_3, \beta_3, A_4, \beta_4)$ parameters, Eq. 4 was evaluated with 10 different $(f_v, a_1, \alpha_1, a_2, \alpha_2)$ values, mimicking voxels with different impulse responses sharing the same blood input function in their TAC. The α_1, α_2 exponents of the impulse response were sampled from interval $[0, \beta_1]$, while the linear parameters were chosen with standard uniform distribution. The total simulation time for each sample was 1 second, during which the TAC was evaluated in 100 points, resulting in 100 positive real activity values in order of magnitude 10^{-2} . Our dataset consisted of 30 000 samples of $(A_1, \beta_1, A_2, \beta_2, A_3, \beta_3, A_4, \beta_4)$ parameter octets, for each octet having 10 samples of TAC data, making a total of 300 000 samples. Each sample consisted of the 100 evaluated activity values, i.e. there were 100 features. We used 70% of the data for training, 15% for validating and the remaining 15% for evaluating our model. As a preprocessing step, the activity values were standardized (they were transformed so that their mean was 0 and the standard deviation was 1).

5.2. Network Architecture

Our model had a CNN-LSTM architecture, using Convolutional Neural Network (CNN) layers for feature extraction and a Long Short-Term Memory (LSTM) layer for interpreting the features through time.

Two 1D convolutional layers were stacked on top of the input, both followed by a maximum pooling layer to reduce feature map size. The convolutional layers learned 50 filters using kernels of size 15. Their output was flattened and fed into an LSTM layer with 100 units. Finally, a fully connected (FC) layer with 4 neurons predicted the four exponents.

All hidden layers used Swish activation¹⁴

$$f(x) = x \cdot \text{sigmoid}(x) = \frac{x}{1 + \exp(-x)}$$

and He uniform weight initialization⁷.



Figure 1: Training loss

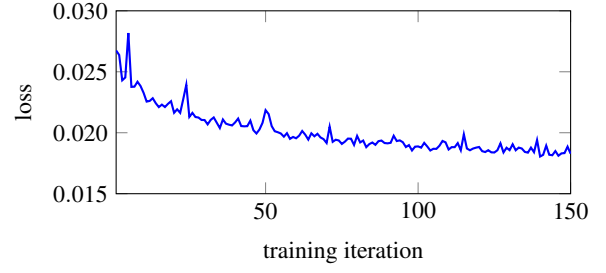


Figure 2: Validation loss

We trained our model using a custom loss function instead of a standard regression loss function like the mean squared error. A standard regression loss function would compare the true and the predicted exponents and calculate the performance of the model based on their numeric difference. However, in our case, numerically close exponents can result in blood input functions that look very different, and vice versa: numerically not too close exponents may define blood input functions that look quite similar. For this reason, we used a loss function that compares the blood input function curves. Eq. 6 was evaluated for both the true and the predicted exponents in 25 points with the A_1, A_2, A_3, A_4 linear parameters set to 1, and the mean of the squared differences of the two calculated curves were used as the loss. Note that only 25 points were used for the evaluation due to computational time constraints, however, these sample points were not evenly distributed over the measurement time $[0, 1]$. Instead, we took into account that the radiotracer spreads abruptly at the beginning of the examination and then slowly subsides, so more samples need to be taken at the beginning in order to lose as little information as possible due to downsampling.

5.3. Network Training

The presented model was trained for 300 epochs using the Adam optimizer¹⁰ with $\alpha = 0.00075$ learning rate and batch size of 128. Figure 1 displays the training and Figure 2 the validation loss during the process. The final training loss was 0.01619 and the validation loss was 0.01785.

6. Results in reconstruction

The proposed methods are tested with a 2D phantom where the number of LORs is $N_L = 10k$ and the number of voxels is $N_V = 1024$ (Figure 4). The 2D phantom has four anatomical parts, namely air, gray matter, white matter, and blood. Parameters of the two-tissue compartment model $[f_v, a_1, a_2, \alpha_1, \alpha_2]$ used in the test are listed in Table 1.

Figure 5 demonstrates the difficulties of the blood input parameter estimation. This figure is the statistical analysis

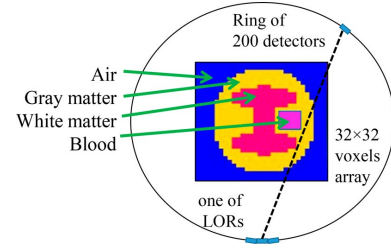


Figure 4: 2D phantom and tomograph model

Table 1: Tissue parameters

Region	a_1	a_2	α_1	α_2	f_v
Background	0	0	0	0	0
Gray matter	0.14	0.86	1.46	0.34	0
White matter	0.05	0.45	1.79	0.01	0
Blood	0	0	0	0	1

of the sinogram, in which the dynamics of the blood input function is embedded. However, due to the convolutions and averaging, the result has no resemblance to the blood input function, which is also shown by this figure. So, a simple extraction of the exponents from the sinogram would significantly underestimate the blood exponents parameters.

We compare four blood input acquisition strategies:

- *Ground truth*: In this case, we assume that the parameters of the Feng model are known a priori. In case of simulation, the same parameters are used to generate the measured data as the setting of the blood input function before reconstruction.
- *Model-based*: This is the classical approach when the statistical analysis of the sinogram data provides an initial estimation, which is refined in parallel with the iterative ML-EM estimation of the tissue parameters.
- *Deep learning*: This approach uses the trained neural net to provide $\beta_1, \dots, \beta_4$ blood parameters in response of the sinogram data. These parameters are set and only the

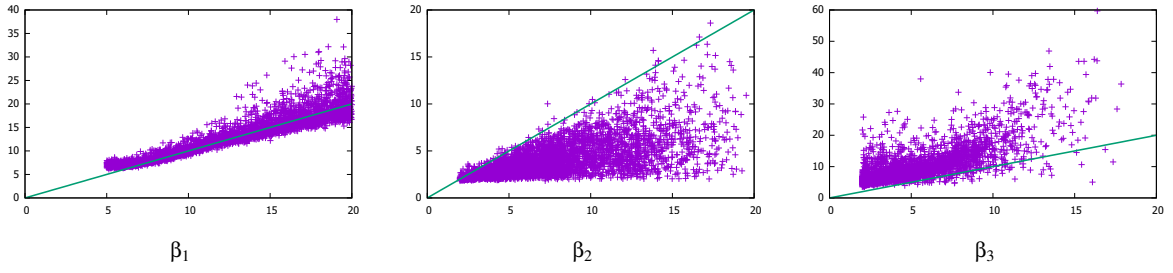


Figure 3: Reconstructions of non-linear parameters $\beta_1, \dots, \beta_3$. The estimation of β_4 is unstable, so it is not listed here. The horizontal coordinate is the ground truth, the vertical is the reconstructed value. Perfect reconstructions would be on the diagonal line.

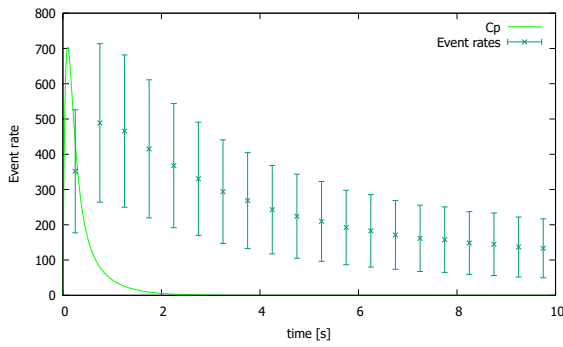


Figure 5: Number of hits per LOR per unit time with error bars showing the standard deviation divided by four. The ground truth of the blood input function is also shown scaled. Note that the dynamics of the sinogram does not reveal the dynamics of the included blood input function.

$A_1, \dots, A_4$ linear parameters are subject to iterative optimization together with the ML-EM iteration.

- *Deep learning + Model-based:* The initial estimation of the $\beta_1, \dots, \beta_4$ parameters are taken from the trained neural net, and these are kept fixed during the first 10 iterations. Then the algorithm becomes a model-based method and further refines the parameters.

Figure 8 depicts the reference blood input function and the TAC in different tissues, as well as the TAC and the macroparameter reconstructions of the examined methods. The total number of coincidences during the 10-second measurement is 420k. The measurement time is decomposed to 30 frames because the Model-based solution does not converge with fewer frames, but the proposed methods do.

Figures 6 and 7 show the convergence speed of the tissue and blood input reconstruction, respectively, for the compared methods. The classical approach fails with fewer frames. The deep learning solutions are quite close to the

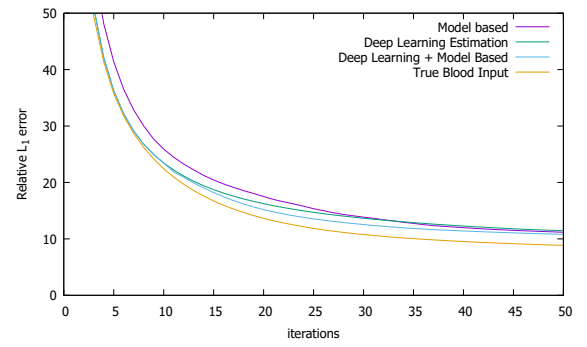


Figure 6: Relative error of the TACs with different blood input function estimation methods

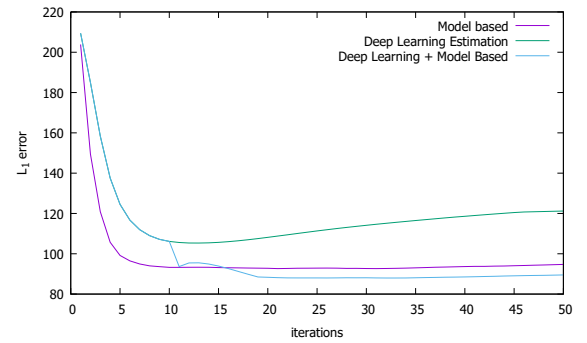


Figure 7: Error of the $C_p(t)$ blood input function reconstruction

optimal case when the blood input function is known a priori. The same conclusions can be drawn from looking at the reconstructed TACs and macroparameter maps (Figure 8).

7. Conclusions

This paper proposed a deep-learning-based method to estimate the exponents of the Feng-type blood input function from sinogram data. The estimated parameters are used to initialize an ML-EM reconstruction scheme that simultaneously determines both the blood input parameters and the tissue parameters.

Acknowledgement

This research has been financed by OTKA K-124124.

References

1. Kewei Chen, Daniel Bandy, Eric Reiman, Sung-Cheng Huang, Michael Lawson, Dagan Feng, Lang sheng Yun, and Anita Palant. Noninvasive quantification of the cerebral metabolic rate for glucose using positron emission tomography, 18f-fluoro-2-deoxyglucose, the patlak method, and an image-derived input function. *Journal of Cerebral Blood Flow & Metabolism*, 18(7):716–723, 1998. PMID: 9663501.
2. François Chollet et al. Keras. <https://github.com/fchollet/keras>, 2015.
3. P. De Groen and B. De Moor. The fit of a sum of exponentials to noisy data. *Journal of Computational and Applied Mathematics*, 20:175 – 187, 1987.
4. D. D. Feng, K. Chen, and L. Wen. Noninvasive input function acquisition and simultaneous estimations with physiological parameters for pet quantification: A brief review. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 4(6):676–683, 2020.
5. Dagan Feng, Sung-Cheng Huang, and Xinmin Wang. Models for computer simulation studies of input functions for tracer kinetic modeling with positron emission tomography. *International Journal of Bio-Medical Computing*, 32(2):95 – 110, 1993.
6. Roger N. Gunn, Steve R. Gunn, and Vincent J. Cunningham. Positron emission tomography compartmental models. *Journal of Cerebral Blood Flow & Metabolism*, 21(6):635–652, 2001. PMID: 11488533.
7. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, december 2015.
8. M.E. Kamasak, C.A. Bouman, E.D. Morris, and K. Sauer. Direct reconstruction of kinetic parameter images from dynamic pet data. *IEEE Trans Med Imaging*, 24(5):636–650, 2005.
9. Bernhard Kaufmann. Fitting a sum of exponentials to numerical data. 06 2003. arXiv:physics/0305019 [physics.data-an].
10. Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
11. Jih-Shian Lee, Kuan-Hao Su, Wen-Yuan Chang, and Jyh-Cheng Chen. Extraction of an input function from dynamic micro-pet images using wavelet packet based sub-band decomposition independent component analysis. *NeuroImage*, 63(3):1273–1284, 2012.
12. Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*, 2018.
13. Á. Szlavetz G. Hesz B. Benyó Á. Cserkaszy J. Lantos D. Légrády Sz. Czifrus A. Wirth B. Kári G. Patay D. Völgyes T. Bükki P. Major G. Németh B. Domonkos M. Magdics, L. Szirmay-Kalos. TeraTomo project: a fully 3D GPU based reconstruction code for exploiting the imaging capability of the NanoPET/CT system. In *World Molecular Imaging Congress*, 2010.
14. Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions, 2017.
15. L. Szirmay-Kalos, Á. Kacsó, M. Magdics, and B. Tóth. Robust compartmental model fitting in direct emission tomography reconstruction. *The Visual Computer*, 2020.
16. L. Szirmay-Kalos, M. Magdics, and B. Tóth. Multiple importance sampling for PET. *IEEE Trans Med Imaging*, 33(4):970–978, 2014.
17. L. Szirmay-Kalos, M. Magdics, B. Tóth, and T. Bükki. Averaging and Metropolis iterations for positron emission tomography. *IEEE Trans Med Imaging*, 32(3):589–600, 2013.
18. Guobao Wang and Jinyi Qi. An optimization transfer algorithm for nonlinear parametric image reconstruction from dynamic pet data. *IEEE Trans Med Imaging*, 31(10):1977–1988, 2012.
19. Hiroshi Watabe, Yoko Ikoma, Yuichi Kimura, Mika Naganawa, and Miho Shidahara. Pet kinetic analysis—compartmental model. *Annals of Nuclear Medicine*, 20(9):583, Nov 2006.
20. Jianhua Yan, Beata Planeta-Wilson, and R.E. Carson. Direct 4d list mode parametric reconstruction for pet with a novel em algorithm. In *Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE*, pages 3625–3628, Oct 2008.
21. Paolo Zanotti-Fregonara, Kewei Chen, Jieh-San Liow, Masahiro Fujita, and Robert B Innis. Image-derived input function for brain pet studies: Many challenges and few opportunities. *Journal of Cerebral Blood Flow & Metabolism*, 31(10):1986–1998, 2011. PMID: 21811289.
22. Paolo Zanotti-Fregonara, El Mostafa Fadaili, Renaud Maroy, Claude Comtat, Antoine Souloumiac, Sebastien Jan, Maria-Joao Ribeiro, Véronique Gaura, Avner Bar-Hen, and Régine Trébossen. Comparison of eight methods for the estimation of the image-derived input function in dynamic [18f]-fdg pet human brain studies. *Journal of Cerebral Blood Flow & Metabolism*, 29(11):1825–1835, 2009. PMID: 19584890.

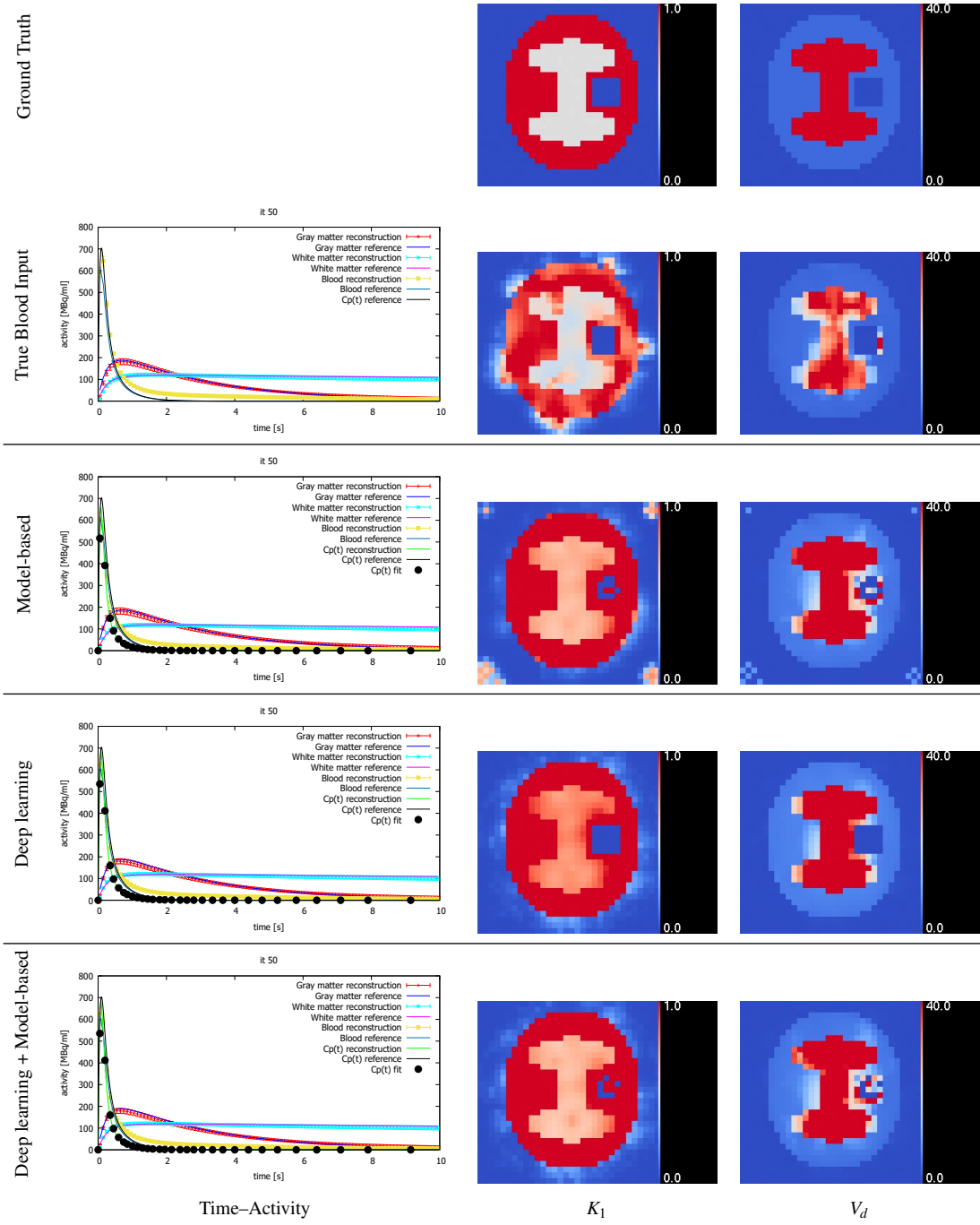


Figure 8: The reconstructed TACs and kinetic macro parameters using the examined blood input estimation methods