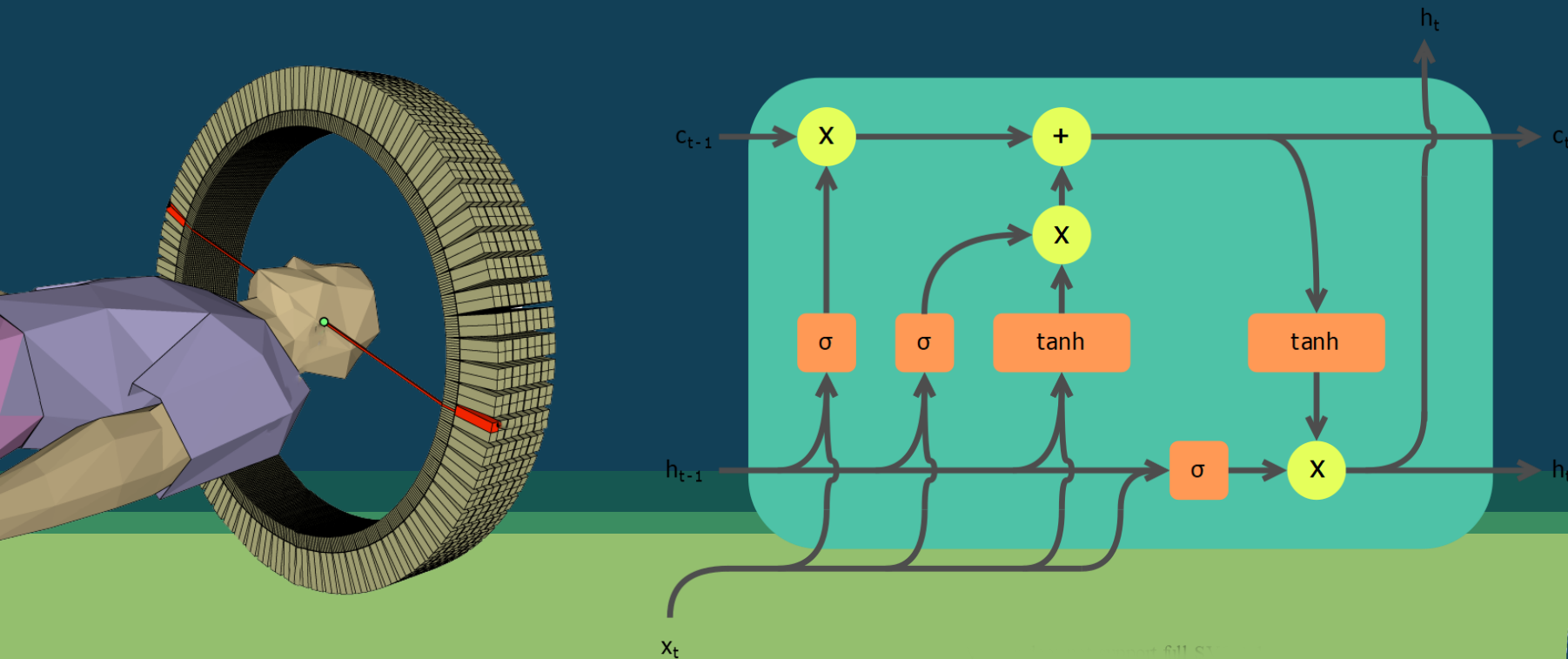


Dóra Varnyú, László Szirmay-Kalos
Budapest University of Technology and Economics, Hungary

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



SZÉCHENYI 2020


HUNGARIAN
GOVERNMENT

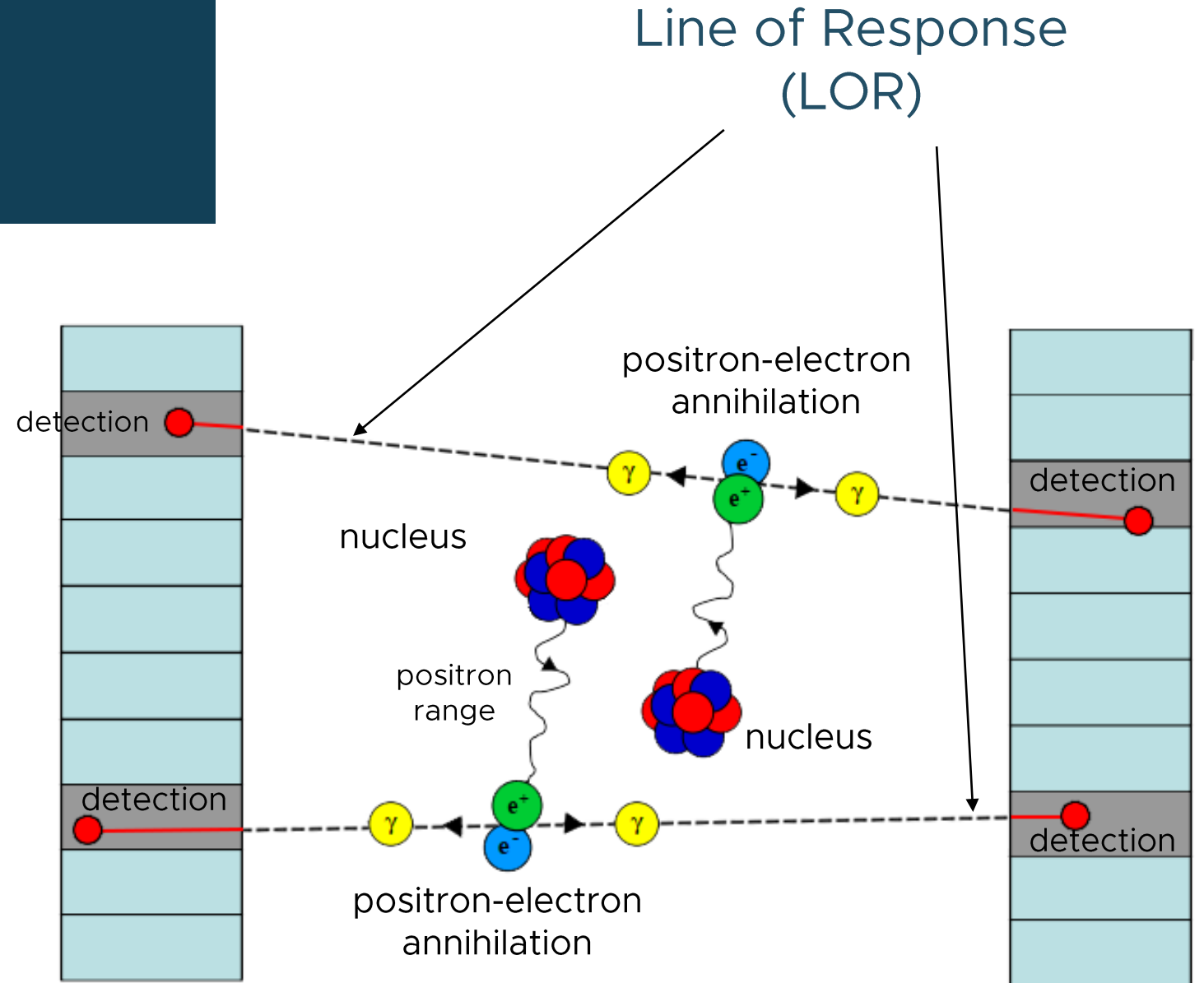
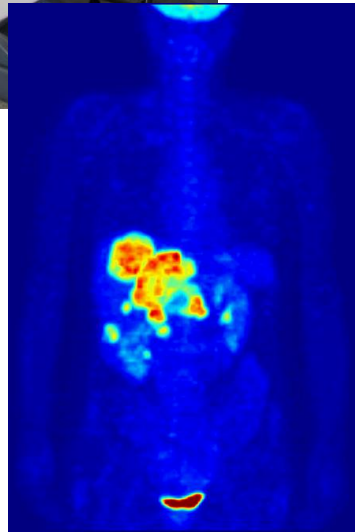
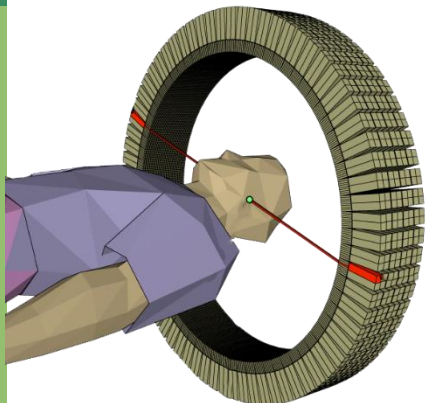
European Union
European Social
Fund



INVESTING IN YOUR FUTURE

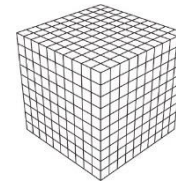


Positron Emission Tomography (PET)

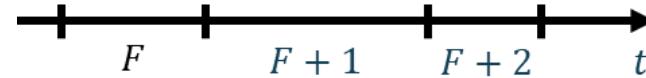




Dynamic PET reconstruction

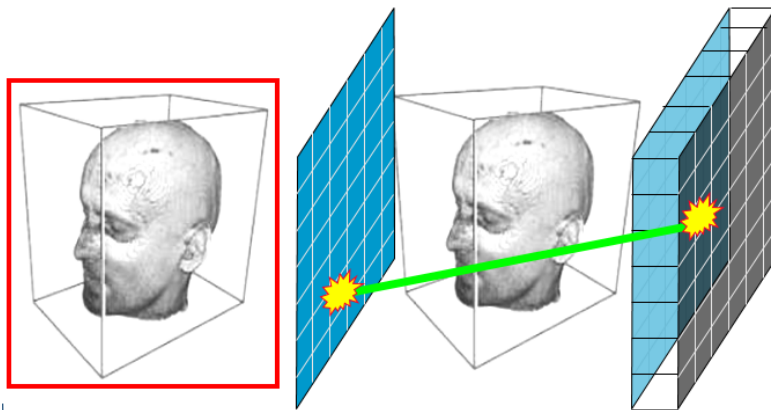


3D voxel grid

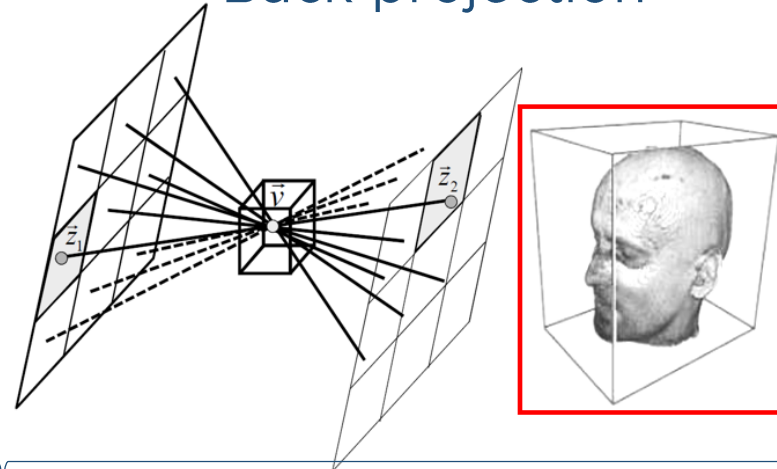


frames

Distribution estimation $\rightarrow$ Expected hit count
Forward projection



Measured hit count $\rightarrow$ Corrected distribution estimation
Back projection

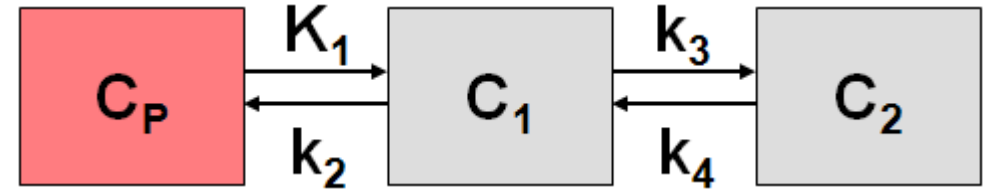


+ fitting the kinetic parametric curve

for all time frame

10-100 iterations

Two-tissue compartment model



Two-tissue compartment model
<https://www.pmod.com/files/download/v31/doc/pkin/2264.htm>

$\mathbf{p}_V = (f_V, c_1, \alpha_1, c_2, \alpha_2)$
 parameter vector

fraction of blood

blood input function

$$K(\mathbf{p}_V, t) = f_V C_W(t) + (1 - f_V) w(t) * C_p(t)$$

total blood
 concentration function

impulse response

$$w(t) = \varepsilon(t) \sum_{i=1}^2 c_i e^{-\alpha_i t}$$

$C_W(t) = \gamma C_p(t)$,
 where γ is a tracer-dependent factor

Reconstruction of the two-tissue compartment model

$\mathbf{p}_V = (f_V, c_1, \alpha_1, c_2, \alpha_2)$
the Time-Activity Curve
depends on it non-linearly!

$$K(\mathbf{p}_V, t) = \overset{f_V^* = \gamma f_V}{f_V^*} C_p(t) + \left(\varepsilon(t) \sum_{i=1}^2 \overset{a_i = (1 - f_V) c_i}{a_i} e^{-\alpha_i t} \right) * C_p(t)$$

f_V^*, a_1, a_2 are linear parameters
if rate constants α_1, α_2 are fixed

↓
direct optimization

How to get $C_p(t)$?

- inverse problem related to this equation
- ill-posed and ambiguous ☹
⇒ more a priori information needed!
- “chicken and egg” type problem

Methods to determine the blood input function

- by taking blood samples, $C_W(t)$ can be calculated, from which $C_p(t) = \frac{1}{\gamma} \cdot C_W(t)$
 - however, taking blood samples is a difficult and error-prone process
 - 3 classical methods to tackle this:
1. Population-Based Input Function
 - takes the average of previous samples and scales it based on a few current samples
 - does not zeroize, but drastically reduces the number of required samples
 - fixed shape → not flexible
 2. Image-Derived Input Function (IDIF)
 3. Model-Based Input Function (MBIF)

Image-Derived Input Function

requires the segmentation of the artery

$$C_m(t) = C_W(t) \cdot R + C_t(t) \cdot S$$

Diagram illustrating the equation for the reconstructed activity in the artery ($C_m(t)$), which is the sum of the total blood concentration function ($C_W(t)$) multiplied by the recovery coefficient (R), and the reconstructed activity in the surrounding tissues ($C_t(t)$) multiplied by the spill-in coefficient (S).

Labels and arrows pointing to the equation components:

- reconstructed activity in the artery (points to $C_m(t)$)
- total blood concentration function (points to $C_W(t)$)
- recovery coefficient (points to R)
- spill-in coefficient (points to S)
- reconstructed activity in the surrounding tissues (points to $C_t(t)$)

ill-posed problem → additional information needed, e.g. R and S

- estimate R and S from the PSF → complicated, not accurate
- extrapolate $C_W(t)$ from late venous samples or by selecting maximum intensity voxels, then calculate R and S with least squares optimization
- *independent component analysis (ICA)*: we assume that two non-Gaussian, statistically independent components are added; also suffers from the indeterminacy of the scaling factors
→ real blood samples are needed to scale the reconstructed functions

Model-Based Input Function

simultaneous estimation of the blood input function and the kinetic parameters

Feng model:

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

if $t > 0$ (the time of injection) and zero otherwise

- also has ambiguity, e.g. when scaling down the tissue model parameters f_V^*, a_1, a_2 , and simultaneously scaling up the linear blood input parameters $A_1, \dots, A_4$
- this can be resolved by taking a few blood samples



Proposed method

- finds the parameters of the Feng model
 - model-based input function
- needs identification of “blood voxels” that at least partially contain artery region
 - image-derived input function
- to guarantee fast and certain convergence, good initialization is crucial
- we aim to obtain the non-linear parameters $\beta_1, \dots, \beta_4$ directly, which can be used to initialize the iterative reconstruction process

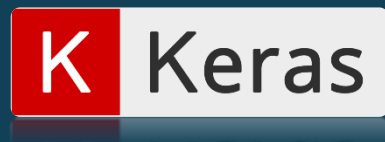
Dependence on the blood input parameters

- every voxel TAC is a weighted sum of $e^{-\beta_1^* t}, \dots, e^{-\beta_4^* t}, te^{-\beta_1^* t}$
- the LOR data are the weighted sums of the TACs of the voxels
→ each measured data element is also a weighted sum of $e^{-\beta_1^* t}, \dots, e^{-\beta_4^* t}, te^{-\beta_1^* t}$
- task: retrieve the exponential functions from the measured data
- with sine functions, Fourier transform would work
- unfortunately, fitting with exponentials is more complicated (e.g. no orthogonality); furthermore, it cannot handle the $te^{-\beta_1^* t}$ type terms
- we chose a more general, robust method



neural network

Neural network





Dataset

- series of time–activity data obtained by evaluating the concentration function $K(\mathbf{p}_V, t)$ with randomized parameters $(f_V, a_1, \alpha_1, a_2, \alpha_2, A_1, \beta_1, A_2, \beta_2, A_3, \beta_3, A_4, \beta_4)$
- for each octet of $(A_1, \beta_1, A_2, \beta_2, A_3, \beta_3, A_4, \beta_4)$, the concentration function 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
- 300 000 samples
- 70% training, 15% validating, 15% testing
- the total simulation time for each sample was 1 second, during which the TAC was evaluated in 100 points → 100 features
- preprocessing: all activity values were standardized

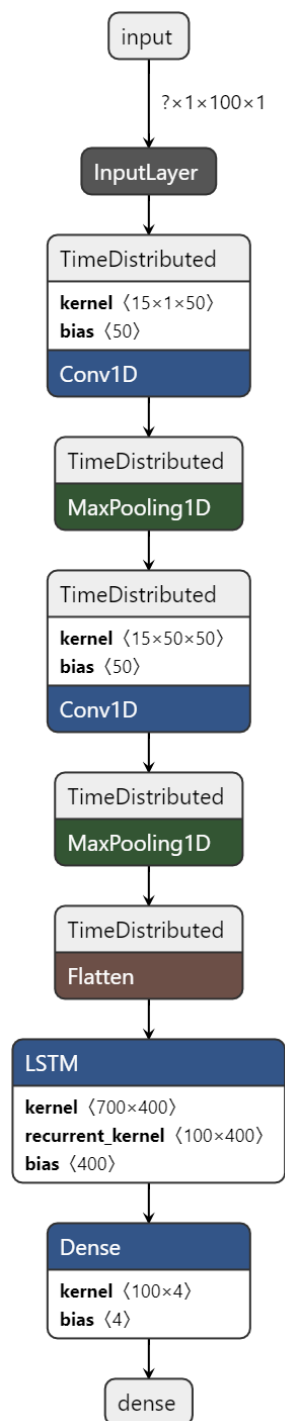


Architecture

- **CNN-LSTM architecture:** convolutional layers extract features and a Long Short-Term Memory layer interprets the features through time
- all hidden layers used Swish activation:

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

- custom loss function
 - motive: numerically close exponents can result in blood input functions that look very different, and vice versa
 - solution: $C_p(t)$ 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 (time samples were not evenly distributed, more was taken at the beginning)

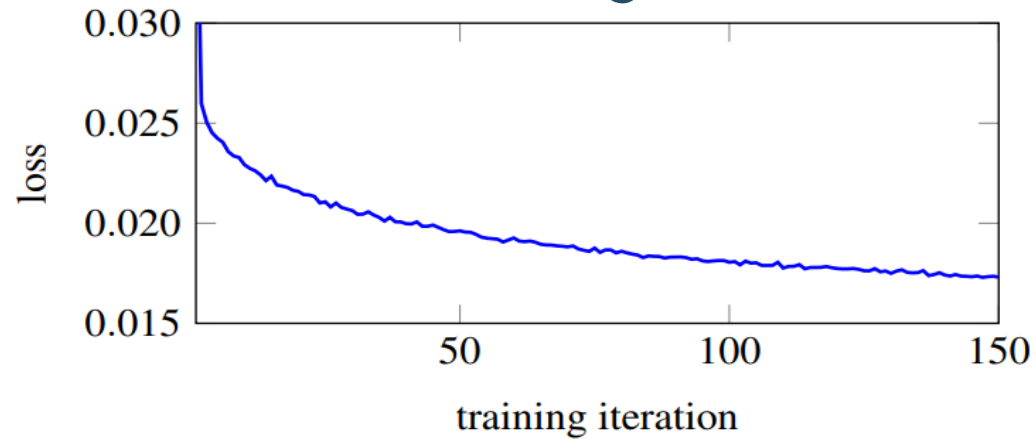




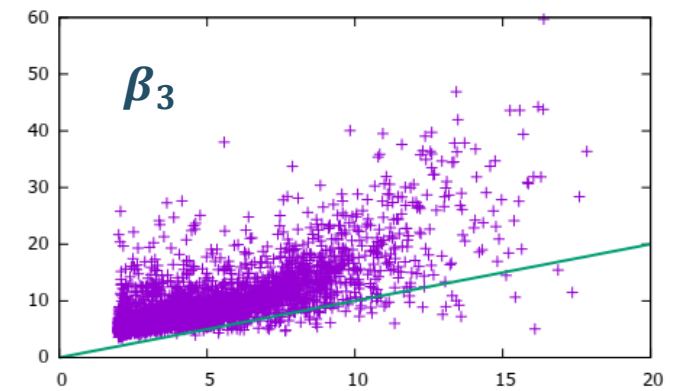
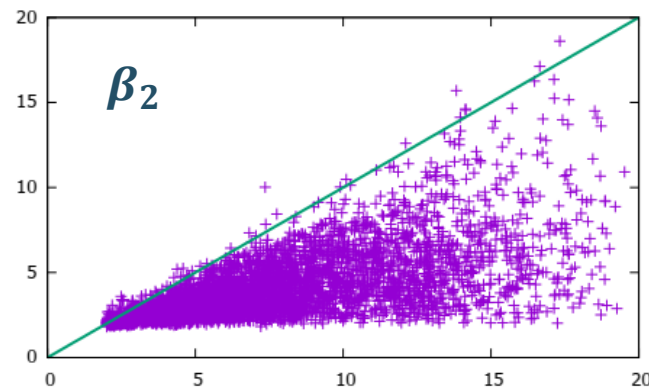
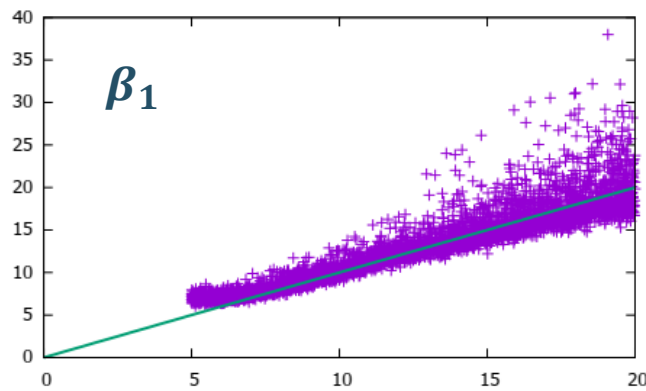
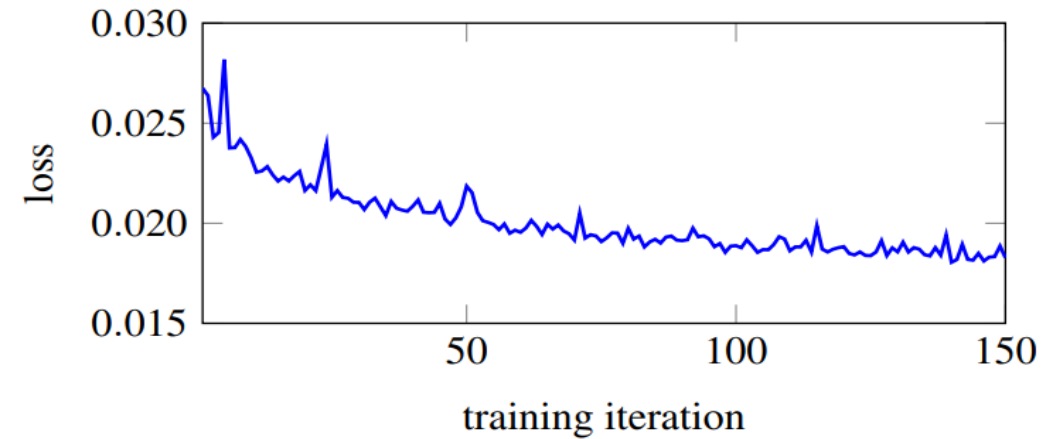
Training

- 300 epochs
- $\alpha = 0.00075$ learning rate
- Adam optimizer
- batch size of 128

Training loss

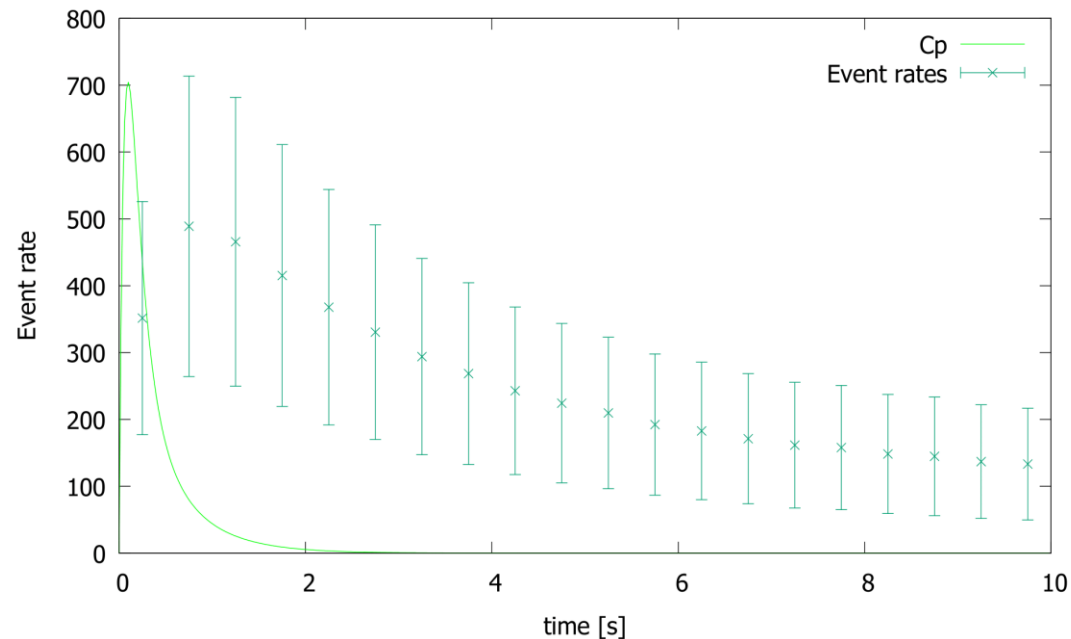
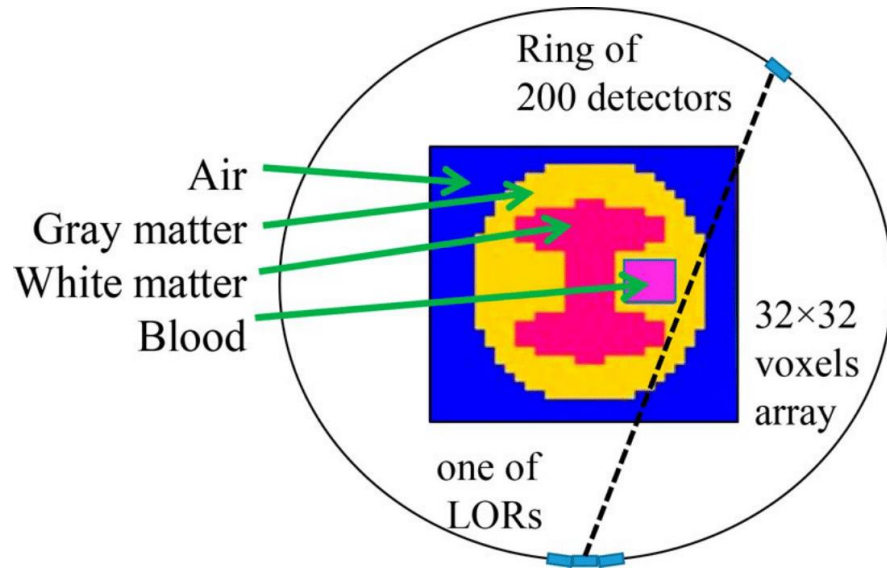


Validation loss



Results in reconstruction

2D phantom



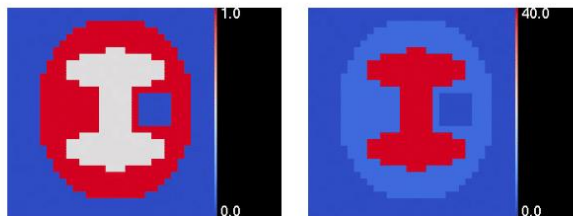
Tissue parameters

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

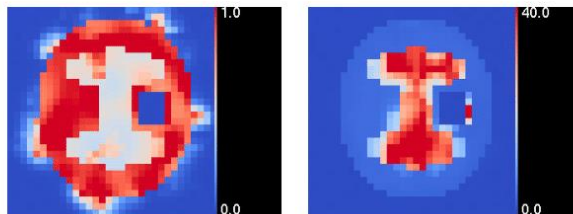
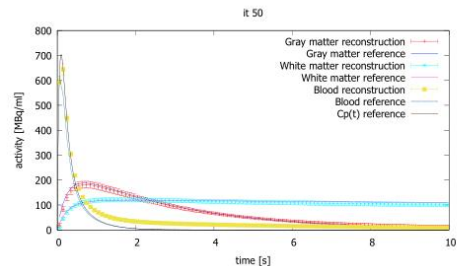
Compared blood input acquisition strategies:

1. ground truth
2. model-based
3. deep learning
4. deep learning + model-based

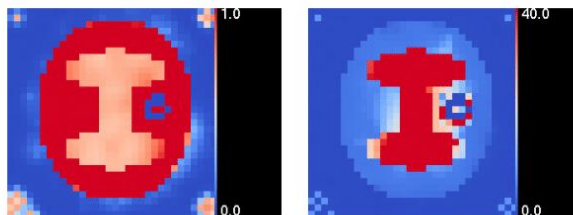
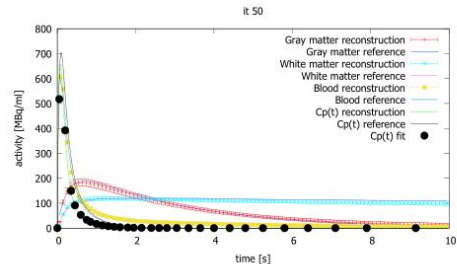
Ground Truth



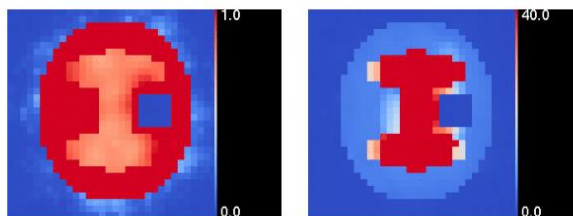
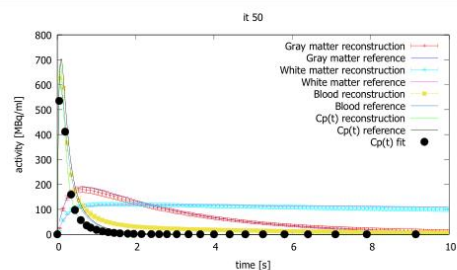
True Blood Input



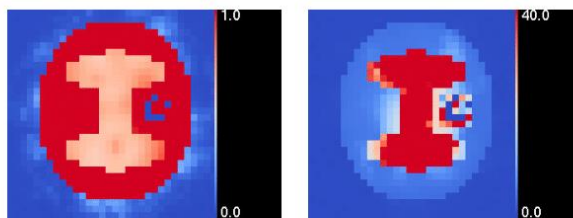
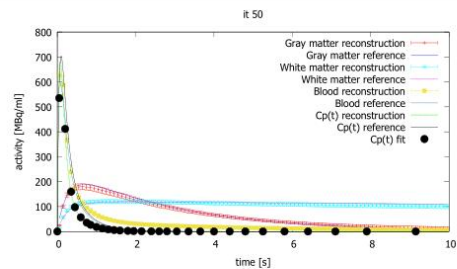
Model-based



Deep learning



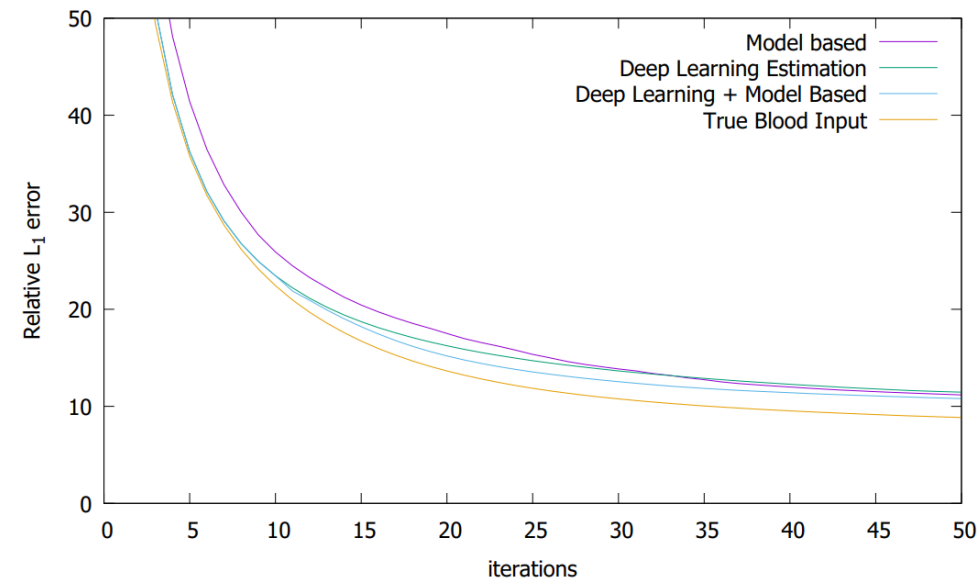
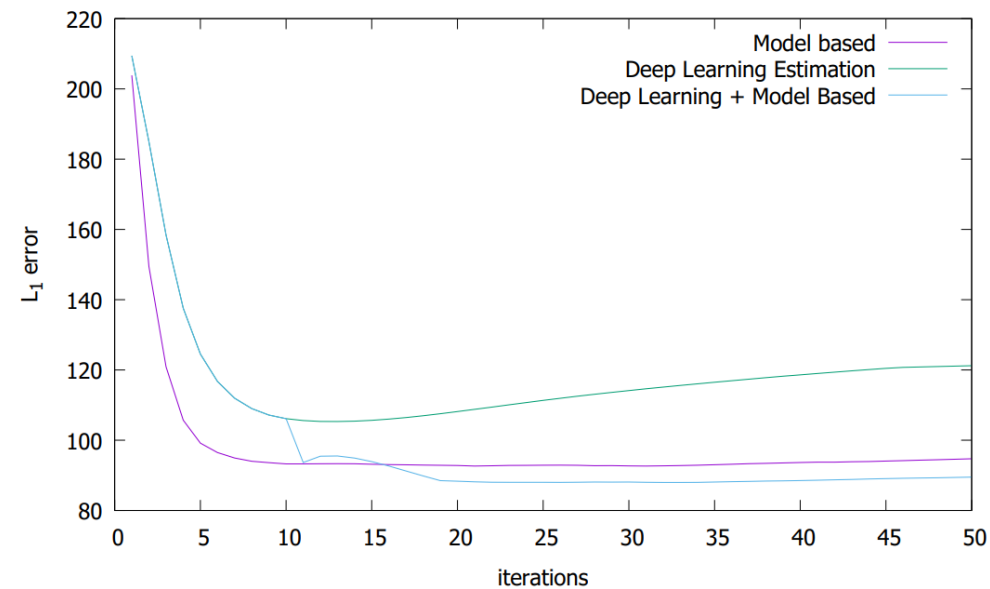
Deep learning + Model-based

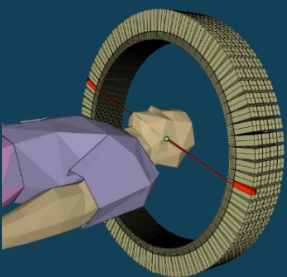


Time-Activity

 K_1 V_d

Relative error of the TACs

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



Thank you for your attention!

This research has been financed by OTKA K-124124 and by the European Union, co-financed by the European Social Fund (EFOP 3.6.1-16-2016-00014, Research, Development, and Educational Integration of Disruptive Technologies in the Field of E-mobility; EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).

SZÉCHENYI 2020



HUNGARIAN
GOVERNMENT

European Union
European Social
Fund



INVESTING IN YOUR FUTURE