Deep Learning for Positioning of Gamma Interactions in Monolithic PET Detectors

Milan Decuyper, Mariele Stockhoff, and Roel Van Holen

I. Introduction

MONOLITHIC scintillation detectors are investigated as an alternative to pixelated detectors in PET systems due to their good timing and energy resolution, high sensitivity and potentially lower cost. However, monolithic crystals are not yet implemented in any commercial clinical scanner as accurate and efficient algorithms are required to position scintillation events in the crystal. To this end, a wide range of statistical positioning algorithms, like maximum likelihood estimation [1], k-nearest neighbor (kNN) [2], gradient tree boosting [3] or neural networks [4] have been proposed. These methods require calibration where often a pencil beam is traversed over the crystal surface with a certain step size. They need to achieve a high spatial resolution and should be computationally efficient to process all events from a large number of detector blocks with a sufficient rate. The kNN algorithm for example attains good spatial resolutions but a distance metric needs to be calculated with all or a subset of training events which is computationally intensive.

In this work, we use a calibration dataset obtained from optical simulations (Stockhoff et al [5]) to investigate the use of deep neural networks for high resolution 2D positioning of scintillation events in a monolithic crystal. Neural networks have the advantage that they are able to learn complex non-linear relationships, can outperform more traditional machine learning algorithms and, once trained, events can be positioned with one forward propagation through the network which is fast and parallelizable on a GPU.

II. Materials and Methods

A. Data

In this study we use simulation data of a monolithic 50x50x16 mm^3 L(Y)SO scintillation crystal with a back side SiPM readout, 75% PDE and without LYSO intrinsic radioactivity. Arrays of 3 mm SiPM pixels are used with combined channel readout (summing the rows and columns) resulting in 32 channels. The calibration data is obtained using a perfect monoenergetic 511 keV source in 1 mm steps. Hence events are acquired for 49x49 positions. After energy filtering, the dataset consists of 10,000 calibration events per position from which 2,000/pos. are used for validation and an additional test set of 2,000 events/pos. for final evaluation. Each event is standardized to zero mean and unit variance. More details regarding the optical simulation setup can be found in the paper by Stockhoff et al [5].

Submitted for review on May 8, 2019

The authors are with the department of Electronics and Information Systems, Medical Image and Signal Processing, Ghent University, 9000 Ghent, Belgium (e-mail: milan.decuyper@ugent.be).

B. Neural Network Positioning

The positioning algorithm used is a regression neural network with 32 inputs and 2 outputs (x and y coordinate) as illustrated in Figure 1. Every hidden layer is followed by a softsign activation function. The network weights are optimized through backpropagation using stochastic gradient descent with Nesterov momentum ($momentum = 0.9$), mini-batch size of 256 and initial learning rate of 10^{-3}. The mean squared error between the predicted and ground truth calibration position was used as the loss function. Training is stopped when the validation loss did not improve for 30 epochs. After training, the network performance is evaluated on the independent test set. The networks are implemented in PyTorch and trained on an 11 GB NVIDIA GTX 1080 Ti GPU.

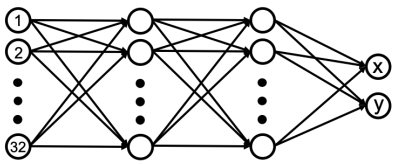

Figure 1: Neural Network architecture with 32 inputs, two hidden layers and two outputs x and y

Different networks are evaluated in terms of spatial resolution and computation time required to position 10,000,000 events on the GPU with varying number of hidden layers (two, three or four), number neurons (from 64 to 1024) in each layer and number of training events per position (from 1000 to 8000).

C. Nearest Neighbor Positioning

The neural network performance is compared with a nearest neighbor positioning algorithm. Here, the mean of all (energy filtered) events for each position is calculated and stored as a calibration map and all maps are interpolated to a calibration grid step size of 0.25 mm. For new events, the square difference to every calibration map is calculated. The event is then classified to the position with the nearest calibration map.

D. Performance Measures

We used the following performance parameters to evaluate the positioning models:

- Bias: Defined as the Euclidean distance between the actual and predicted x,y-position. The bias is calculated for every test event and the median bias is reported.
- FWHM: For each beam position a 2D histogram of the predicted positions was created from which the full width at half maximum (Gaussian fit) along the x- and y-

directions was calculated as a measure of the detector's intrinsic spatial resolution.

III. Results and Discussion

A. Neural Network Performance

The median bias performance measures for different network architectures is shown in Figure 2. We can observe that for 1,000 training events/pos., the bias saturates or even increases for more complex networks as overfitting starts to occur beyond 256 neurons. By increasing the number of training events, we can further improve the performance. The best performance is obtained with a neural network with four hidden layers, 512 neurons in each layer and 8000 training events per position for which the median bias and average FWHM values are shown in Table 1. A 2D histogram of all test predictions across the whole detector is plotted in Figure 4. These results show that a very high resolution can be obtained with enough training data and a complex network.

When reducing the complexity and/or the number of training events, the performance decreases. The resolution versus training set size for a three hidden layer (with each 256 neurons) network can be observed in Figure 3. With 1,000 events/pos. a FWHM of 0.52 mm is achieved. However, the resolution can be improved to a FWHM of 0.43 mm through the use of data augmentations (DA) like transposing and flipping. Moreover, with this network, 10 million events can be positioned per second. Hence the most favorable model can be selected based on the desired minimum resolution, amount of available calibration data and required event rate.

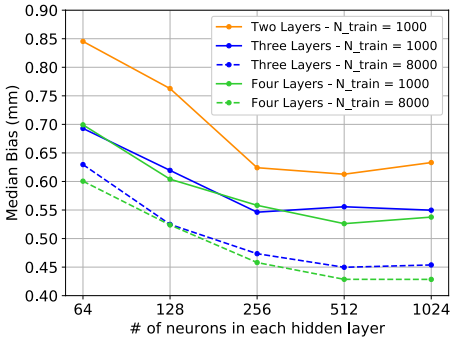

Figure 2: Median Bias for different neural network architectures. N_train indicates the number of training events per position.

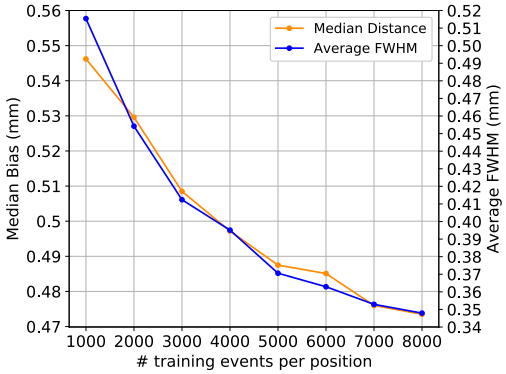

Figure 3: Median Bias and FWHM of a neural network with 3 hidden layers and 256 neurons in each layer for different numbers of training events/pos.

Table 1: Median Bias and mean FWHM (mm) across the whole detector and center 11x11 positions. N_{train} indicates the number of training events/pos. Time signifies the compute time required to position 10 million events on the GPU. DA: Data Augmentation

Network	N_{train}	Whole Detector		Detector Center		Time (s)
		Bias	FWHM	Bias	FWHM	
3L-256	1000	0.55	0.52	0.52	0.41	1.04
	1000+DA	0.51	0.43	0.46	0.33	
	8000	0.47	0.35	0.44	0.28	
4L-512	8000	0.43	0.23	0.39	0.20	3.53

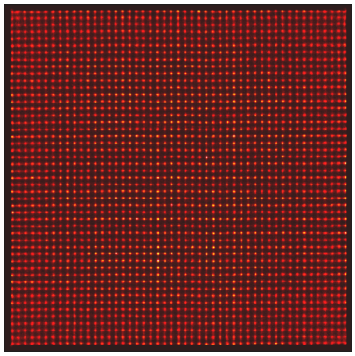

Figure 4: 2D histogram of predicted positions on the test set for the best performing neural network with 4 hidden layers and 512 neurons in each layer.

B. Nearest Neighbor Performance

With the nearest neighbor positioning algorithm, a median bias of 0.56 mm and FWHM of 0.3 mm is achieved in the center (11x11 positions) of the detector with 10,000 calibration events/pos. Hence, with deep neural networks we can achieve a better spatial resolution of 0.20 mm and lower bias of 0.39 mm with less calibration data.

IV. Conclusion

In this work we presented a neural network positioning algorithm trained on data from optical simulations. Different architectures were compared in terms of spatial resolution, event processing rate and required amount of training data. We showed that, based on the same dataset, deep neural networks can achieve a better spatial resolution compared to a nearest neighbor positioning method, even with lower number of training events. Furthermore, these results are better than, to the authors' best knowledge, any result currently reported in literature. Future work includes 3D positioning and training on real datasets.

References

[1] S. España et al., "DigiPET: sub-millimeter spatial resolution small-animal PET imaging using thin monolithic scintillators," *Phys. Med. Biol.*, vol. 59, no. 13, pp. 3405–3420, Jul. 2014.

[2] H. T. van Dam *et al.*, "Improved Nearest Neighbor Methods for Gamma Photon Interaction Position Determination in Monolithic Scintillator PET Detectors," *IEEE Trans. Nucl. Sci.*, vol. 58, no. 5, pp. 2139–2147, Oct. 2011.

[3] F. Muller et al., "Gradient Tree Boosting-Based Positioning Method for Monolithic Scintillator Crystals in Positron Emission Tomography," *IEEE Trans. Radiat. Plasma Med. Sci.*, vol. 2, no. 5, pp. 411–421, Sep. 2018.

[4] P. Conde *et al.*, "Determination of the Interaction Position of Gamma Photons in Monolithic Scintillators Using Neural Network Fitting," *IEEE Trans. Nucl. Sci.*, vol. 63, pp. 30–36, Feb. 2016.

[5] M. Stockhoff et al., "Advanced optical simulation study on the spatial resolution of a thick monolithic PET detector," Submitted to *Phys. Med. Biol. (PMB-108741)*