Ensemble of Neural Networks for 3D Position Estimation in Monolithic PET Detectors

A. Iborra[1], **A. J. González**[2], **A. González-Montoro**[2], **A. Bousse**[1] **and D. Visvikis**[1]

[1] Laboratory of Medical Information Processing (LaTIM), INSERM UMR 1101 – Université de Bretagne Occidentale, Brest, France.

[2] Instituto de Instrumentación para Imagen Molecular (I3M), Centro Mixto CSIC – Universitat Politècnica de València, Valencia, Spain.

E-mail: amadeo.iborra.carreres@gmail.com

Abstract. We propose an ensemble of multilayer feedforward neural networks to estimate the 3D position of photon interactions in monolithic detectors. The ensemble is trained with data generated from optical Monte Carlo simulations only. The originality of our approach is to exploit simulations to obtain reference data, in combination with a variability reduction that the network ensembles offer, thus, removing the need of extensive per-detector calibration measurements. This procedure delivers an ensemble valid for any detector of the same design. We show the capability of the ensemble to solve the 3D positioning problem through testing four different detector designs with Monte Carlo data, measurements from physical detectors and reconstructed images from the MindView scanner. Network ensembles allow the detector to achieve a 2-2.4 mm FWHM, depending on its design, and the associated reconstructed images present improved SNR, CNR and SSIM when compared to those based on the MindView built-in positioning algorithm.

1. Introduction

There has recently been an increasing interest in the use of monolithic scintilation detectors for positron emission tomography (PET) applications (Gonzalez et al. 2019). Such detectors have several advantages including easier detector assembly and reduced costs, without any associated reduction in the overall system performance in terms of spatial resolution or sensitivity (Gonzalez-Montoro et al. 2017).

In pixelated detectors, the determination of the interaction position of a gamma photon is usually performed by pixel identification. However, this procedure is not applicable to monolithic detectors. In this case, position determination is often obtained by a centrality estimate of the distribution of light collected by all photosensors, such as for example the center of gravity of the light distribution. Similarly, depth of interaction (DOI) estimation may be obtained by a dispersion estimate of this distribution. Nevertheless, the sensor with the largest signal does not necessarily correspond to the XY coordinates of the interaction, since a large fraction of the

scintillation photons may be reflected within the crystal before detection. Likewise, the dispersion of the measured light distribution may vary for each XY position at the same DOI, due to different inner reflections. A reference measurement (with known positions of irradiation) is usually used to correct these estimates, mitigating the aforementioned issues. There are different approaches to the interaction positioning estimate depending on the finish of the crystal (González et al. 2015, Llosá et al. 2013, Schaart et al. 2009, Pani et al. 2009). However, most of them do not account for the presence of inner reflections that, in turn, leads to a degradation in the system's energy resolution.

New methods have been proposed in order to improve the interaction positioning estimates, while trying to maximize the advantages and usability of monolithic detectors. These methods include gradient tree boosting (Muller et al. 2018, Muller et al. 2019), k-nearest neighbours approaches (Marcinkowski et al. 2016, Borghi et al. 2015, van Dam et al. 2011, Maas et al. 2009) or neural networks (Wang et al. 2013, Bruyndonckx et al. 2008, Bruyndonckx et al. 2004).

On the other hand, all of these approaches require the measurement of reference events which tend to be extensive and time consuming (Schaart et al. 2009). In this work, we present a method, based on neural network ensembles, that does not require any reference measurement, but instead, uses data from Monte Carlo simulations. Moreover, our proposed approach estimates the interaction point for any detector of a particular design (as opposed to a per-detector calibration), with a single ensemble being able to process an entire scanner. The benefits of the Monte Carlo simulation approach include parallel computing and automation, noiseless and scatter-free reference events, and arbitrary size of the training dataset. Although the proposed algorithm is applicable to different monolithic detector designs, its performance in this work has been tested using elementary detector data from the MindView scanner (Gonzalez et al. 2019). The evaluation included overall system acquired phantom and patient datasets. The implementation of the Monte Carlo simulations and neural network ensembles will be made publicly available upon the publishing of this work‡.

2. Materials and Methods

The light intensity distribution output in a monolithic detector is the outcome of some transformation of the measurements on its array of photosensors, usually, their sum by rows and columns into c channels. A common approach to obtain the interaction XY coordinates is to use a centrality function, such as for example the center of gravity (Anger 1958) (i.e. Anger logic). The DOI (Z) of an event is obtained in a similar manner, using a dispersion function to measure the width of the light distribution. In addition, corrections are applied to these function, obtained by irradiating the scintillator at known positions, thus, being able to translate the centrality and dispersion estimates into interaction coordinates. The specific details of the correction functions vary with the finish of the scintillator, due to differences in the reflections inside

‡ https://github.com/amibcar/eNN-PET

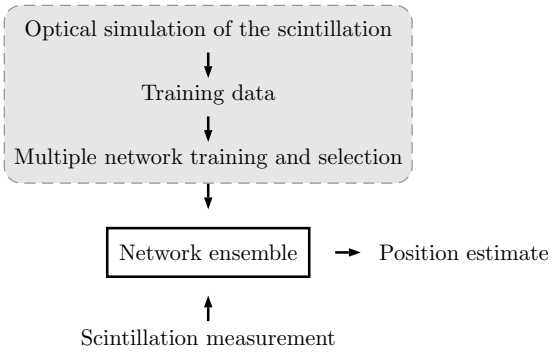


Figure 1: The overall methodology scheme is shown here. We start with the simulation of the output of the detector to obtain a training dataset. Training multiple networks, we select the best performing ones to form an ensemble. Finally, the position of a measured event is estimated through ensemble averaging.

the crystal. Teflon wrapping for example, increases the complexity of the correction functions due to diffuse reflections, while having a positive impact on other features, such as energy resolution (Lecoq 2016, Llosá et al. 2010). Moreover, the XY correction function requires an accurate estimate of the Z coordinate, due to differences in the required correction along different heights in the crystal, while the Z correction requires an XY estimate for similar reasons.

We assume that in general, there exists a function f that maps an intensity distribution $I = (I_1, I_2, \ldots, I_c)$ to the corresponding interaction position (x, y, z). Our goal in this work is to obtain $\hat{f}$, an approximation of f, for a given detector design, without requiring the irradiation of the scintillator at known positions. Different detectors (or sectors of a detector) following the same scintillator design might have differences in optical coupling, photosensor response, etc, affecting their output. However, $\hat{f}$ should be applicable to any detector based on the same design.

Multilayer feedforward neural networks are known to be universal approximators (Hornik et al. 1989, Hornik 1991). However, if we obtain $\hat{f}$ through a single network, the aforementioned variability in I may introduce inter- and intra-detector variability in the predictions. Hence, in this work we use a network ensemble (or committee) (Hansen & Salamon 1990), which are known to reduce variability in predictions. Regarding the training dataset required for the network, it is generated from Monte Carlo optical simulations. This allows a) avoiding the introduction into the training dataset of inter- and intra-detector variation, and b) avoiding to irradiate the scintillator to obtain reference events. The proposed method is outlined in figure 1 and described in the following sections. It should be noted that the elements contained in the gray box (see figure 1) are only computed once per detector design. Thus, in a scanner with N detector modules following the same design, only one ensemble is required, predicting interaction coordinates for all N modules.

2.1. Neural Networks

Multilayer feedforward neural networks are the building block of the proposed method. The goal of each network is to predict a single coordinate of the interaction point based on c features corresponding to a light intensity distribution $I = (I_1, I_2, \ldots, I_c)$.

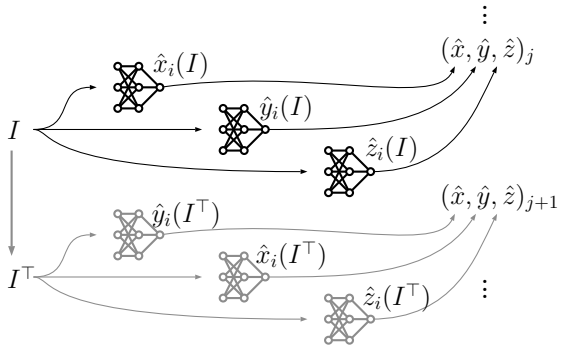


Figure 2: Outline of the i-th component of the network ensemble, generating the j-th prediction. In case that there is an applicable symmetry, the i-th component can be reused to generate the $(j+1)$-th prediction. Otherwise, the $(i+1)$-th component generates the $(j+1)$-th prediction. This outline describes the contents of the network ensemble box in figure 1.

We impose on each network the following fixed hyperparameters: ReLU (Nair & Hinton 2010) activation function, no dropout (Srivastava et al. 2014) and no batch normalization (Ioffe & Szegedy 2015). Additionally, all networks are trained with RMSE loss function and adagrad (Duchi et al. 2011) optimizer, which provides adaptive learning rates. These restrictions define a set of networks that differ only in the number of hidden layers h and the number of nodes in each layer $(n_1, n_2, \ldots, n_h) \in \mathbb{N}^h$. $\bigcup_h \mathbb{N}^h$ is the set of all possible configurations of the form $(n_1, n_2, \ldots, n_h)$ for any h. We consider a subset $\mathcal{A} \subset \bigcup_h \mathbb{N}^h$ of candidate configurations (*search space*), and for simplicity, we will use the variable i as the i-th element (candidate configuration) of $\mathcal{A}$.

For a given network configuration i, we separately train three versions of it, considering as labels the corresponding coordinate of the interaction point. Thus, we solve the problem of finding an approximate of

$$f(I) = (x, y, z) \ , \tag{1}$$

where (x, y, z) is the interaction position, by finding triples of networks that compute

$$\hat{f}_i(I) = (\hat{x}_i(I), \hat{y}_i(I), \hat{z}_i(I)) \ . \tag{2}$$

We assume that in general, the problems of predicting X, Y and Z coordinates are of similar complexity. Therefore, a successful configuration $i = (n_1, n_2, \ldots, n_h)$ for a given detector design, requires a good performance in the three predictions. Since there is no evident relationship between detector designs and successful configurations, we define $\mathcal{A}$ a search space of candidate configurations. As we train and test the candidates, we save the better performing (triples of) trained networks to form an ensemble. After testing all the candidates, we obtain the best networks for a given detector design in the defined search space.

In this work, test an $\mathcal{A}$ containing all configurations for $h \in \{2, 3, 4, 5, 6\}$ and $n_i \in \{100, 200, 400\}$ in any order, totalling 1089 candidate configurations for each coordinate. From these candidates, we select the 10 best performing $\hat{f}_i(I)$. Let $(I_1, I_2, \ldots, I_{c/2})$ be the light distribution collected along the rows of the photosensor, and correspondingly $(I_{c/2+1}, I_{c/2+2}, \ldots, I_c)$ for the columns. Then, due to symmetry in the studied detectors, from every

$$I = (I_1, \ldots, I_{c/2}, I_{c/2+1}, \ldots, I_c) \ ,$$

we generate a second input

$$I^{\top} = (I_{c/2+1}, \ldots, I_c, I_1, \ldots, I_{c/2}) \ , \tag{3}$$

that is the light distribution of the same interaction, after transposing the photosensor array. Thus, we assume that if $f(I) = (x, y, z)$, then $f(I^{\top}) = (y, x, z)$. So, for every selected $\hat{f}_i(I)$ we will add an additional

$$\hat{f}_i(I^{\top}) = \big(\hat{y}_i(I^{\top}), \hat{x}_i(I^{\top}), \hat{z}_i(I^{\top})\big) \tag{4}$$

to the ensemble, leaving an ensemble size of 20 point predictors for every measured I (see figure 2). In case that this symmetry cannot be exploited, we would simply select the 20 best performing $\hat{f}_i(I)$. The final prediction, which is an approximate of (1), is the ensemble average

$$\hat{f}(I) = \frac{1}{20}\left(\sum_{i=1}^{10} \hat{x}_i(I) + \hat{y}_i(I^{\top}), \sum_{i=1}^{10} \hat{y}_i(I) + \hat{x}_i(I^{\top}), \sum_{i=1}^{10} \hat{z}_i(I) + \hat{z}_i(I^{\top})\right) \qquad \text{or} \qquad (5a)$$

$$\hat{f}(I) = \frac{1}{20}\left(\sum_{i=1}^{20} \hat{x}_i(I), \sum_{i=1}^{20} \hat{y}_i(I), \sum_{i=1}^{20} \hat{z}_i(I)\right) \ . \tag{5b}$$

Recall that i refers to a given configuration and in this case it refers to the 10 (or 20) better performing network configurations, depending on whether the previously described symmetry can be exploited (5a) or not (5b). Let N be the number of networks in our ensemble. The result of (5a) for a given I converges as N increases. However, each network used in the ensemble increases the computational cost of the solution. We would like to find an N so that the inclusion of the $(N+1)$-th network in the ensemble changes its result by less than τ in most of the cases (see section 3.3).

Regarding the implementation details, I is normalized before the input layer, so that

$$\sum_{k=1}^{c} I_k = 1 \ . \tag{6}$$

This allows the networks to be insensitive to differences in inter-detector gain, although they are still sensitive to intra-detector gain variations. This normalization also ensures that there will be no large scale differences between features from different events. Thus, there is no reason to introduce any batch normalization. Regarding the dropout, the explicit search of different network configurations in $\mathcal{A}$ replaces the potential benefits of dropping out units. Even though it is an efficient model averaging technique, it is important for the heterogeneity of the ensemble to avoid the potential inclusion of duplicate $(n_1, n_2, \ldots, n_h)$ for the prediction of the same I. The explicit search over $\mathcal{A}$ without using dropout ensures total control over the included architectures on the ensemble.

In terms of training, validation and testing, we shuffle and split simulated events into three datasets: *train* (50%), *validation* (5%) and *test* (45%). Let an epoch be the number of steps necessary to train a network over all *train* dataset records. Networks

are trained from *train* and predict the *validation* data (which do not form part of the training) every 4 epochs. When the *validation* results stop progressing, or we reach 20 epochs, the training stops. The following step is to predict the *test* data, yielding the score of the network. In all our cases, training reached 20 epochs without degradation of the *validation* results (see section 3). The loss function and the selection criteria for the ensemble are discussed in the following section.

Lastly, the selected optimizer has several advantages in this framework. It avoids the need to introduce any control on sampling of training events over crystal dimensions. This is particularly useful over the Z dimension, due to the exponential decay of counts along the depth of the scintillator. Thus, there is no need to tune learning rates, since those are estimated by the optimizer to compensate for uncommon parameter updates. On the other hand, the adagrad optimizer reduces the magnitude of gradient over time, eventually preventing further learning. Other optimizers with different learning rates' estimation policies can be also considered within the proposed framework. The neural networks have been implemented using TensorFlow (Abadi et al. 2015) open source software library.

2.2. Figures of merit

In the j-th prediction, let $\hat{q}_j$ be the predicted value of the interaction coordinate q_j. After N predictions, we evaluate a network with the root mean square error, defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(\hat{q}_j - q_j)^2} \ . \tag{7}$$

RMSE is the loss function during training, since we seek to minimize large discrepancies between f and $\hat{f}_i$.

When considering predicted points $\hat{p}_j = (\hat{x}, \hat{y}, \hat{z})$ of an interaction position $p_j = (x, y, z)$, we evaluate the performance of an ensemble of networks based on their distance as the mean absolute error

$$\mathrm{MAE} = \frac{1}{N}\sum_{j=1}^{N}\|\hat{p}_j - p_j\| \ . \tag{8}$$

We use MAE to analyze the results of each ensemble through the scintillation block, since it provides a convenient magnitude of the expected error in each region of the crystal. Additionally, as it is computed from the test dataset, it shows a lower bound for such errors. Implemented and simulated detectors might differ, potentially decreasing the performance of the predictions. However, depending on the performance objectives of a detector design, other metrics might be also used. Neither training or ensemble selection are influenced by (8).

We also consider the *accuracy* within t of a coordinate prediction as

$$\mathrm{Acc} = \frac{|P|}{N} \quad \text{such that} \quad P = \{\hat{q}_j : |\hat{q}_j - q_j| < t\} \ , \tag{9}$$

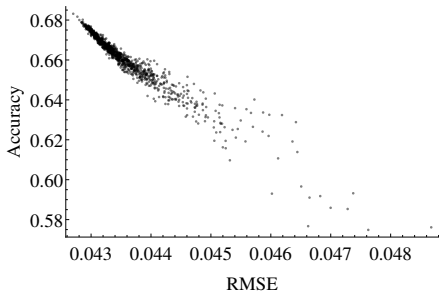


Figure 3: Accuracy (9) within 1 mm of the prediction of the X coordinate as a function of RMSE (7).

where $|P|$ denotes the cardinal of the set P. As figure 3 shows in our context, the Accuracy is strongly correlated with RMSE. On the other hand, Accuracy provides an intuitive quantification of the performance of a network in $[0, 1]$, and disregards the magnitude of errors lower than t, therefore not counting them towards the penalty of a network to be part of the final ensemble. Accuracy, as defined by (9), does not take into consideration the magnitude of errors larger than t, but instead accounts them as *misses*. However, since we use RMSE as a loss function, we expect that the optimization of each network prevents errors of large magnitude. The reason behind this definition of Accuracy is that, in our context, we are not especially interested in further minimization of the error in certain regions (e.g., center), while having large errors in others (e.g., edges). Instead, we are interested in having a response as uniform as possible through the scintillation volume, maximizing the expected events that will be considered reasonably close to the interaction point. As figure 3 shows, for a given RMSE, there could be several networks with different accuracies. Thus, we rank networks based on (9) for each coordinate, in order to maximize the number of events that will be considered sufficiently close to the interaction point.

Given a triple of networks according to (2), we consider its accuracy predicting an interaction point as

$$\mathrm{Acc}_{\mathrm{sel}} = \mathrm{Acc}_X \cdot \mathrm{Acc}_Y \cdot \mathrm{Acc}_Z \ , \tag{10}$$

where Acc_X, Acc_Y and Acc_Z represent the accuracy in predicting X, Y and Z coordinates separately. Therefore, (10) is used to decide which networks form the final ensemble. For a given network configuration $i = (n_1, n_2, \ldots, n_h)$, Acc_X, Acc_Y and Acc_Z are computed from the test dataset when the respective trainings are finished, avoiding the need to save the resulting predictions (which could have non-negligible size for large tests). On one hand, it is convenient that X, Y and Z trainings can be executed in no particular order. On the other hand, it is also convenient to know as soon as possible if i will perform better than one of the configurations in the current ensemble. Assume the worst configuration w of the provisional ensemble have an accuracy A_{low} according to (10). During the search of the best configurations:

- As soon as any Acc_X, Acc_Y or Acc_Z is lower than A_{low}, the current network can be discarded and its pending trainings canceled.
- Whenever all Acc_X, Acc_Y and Acc_Z are known, the new configuration either replaces w if $A_{\mathrm{low}} <$ (10), or it is discarded.

After the exploration of the search space it is relevant to determine the accuracy predicting an interaction point as an extension of (9), based on the distance between the two points

$$\text{Accuracy} = \frac{|P|}{N} \quad \text{such that} \quad P = \{\hat{p}_j : \|\hat{p}_j - p_j\| < t\} \ , \tag{11}$$

assessing the performance of an ensemble as the percentage of predictions $\hat{p}_j$ closer than t to their interaction point p_j.

Finally the obtained interaction position maps are compared according to the signal-to-noise and contrast-to-noise ratios as well as the structural similarity (Wang et al. 2004) defined as:

$$\text{SNR} = 10 \log_{10} \left(\frac{\mu_r}{\sigma_r} \right) \tag{12}$$

in dB, where r is an homogeneous volume far from ROI borders, and μ_r and σ_r are its mean and standard deviation.

$$\text{CNR} = \frac{|\mu_r - \mu_b|}{\sqrt{\sigma_r^2 + \sigma_b^2}} \ , \tag{13}$$

where b is a homogeneous volume in the background, far from ROI borders.

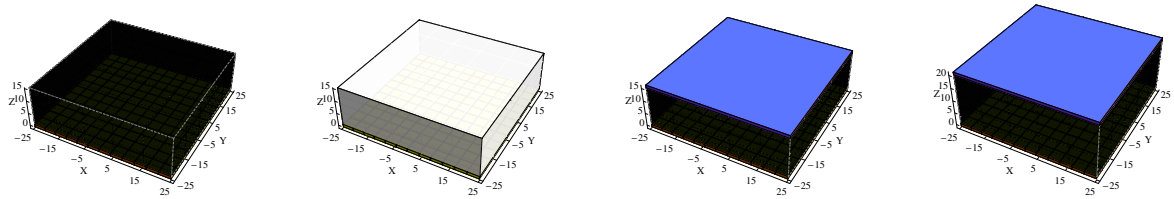
$$\text{SSIM}(I_1, I_2) = \frac{(2\mu_1\mu_2 + C_1)(2\sigma_{12} + C_2)}{(\mu_1^2 + \mu_2^2 + C_1)(\sigma_1^2 + \sigma_2^2 + C_2)} \ , \tag{14}$$

where μ_1 and μ_2 are local means of I_1 and I_2, σ_1 and σ_2 are the corresponding local standard deviations, and σ_{12} is the covariance between I_1 and I_2. Regularization constants C_1 and C_2 are set as suggested by the authors of the metric in (Wang et al. 2004).

We evaluate metrics given by (12), (13) and (14) (see section 3.6) on the reconstructed phantoms. We use a digital reference object (DRO) to define a region of interest (ROI) of constant activity. After erosion, we obtain r, a homogeneous volume of constant activity, far from ROI borders. Then, we compute the SNR (12) of the activity inside the phantom. Likewise, after inversion and erosion of the DRO, we obtain b, a background volume inside the phantom, far from ROI borders. We use b to compute the CNR (13) between activity and background regions inside the phantom. Finally, we apply a 2 mm Gaussian filter to the DRO and compute the SSIM (14) between this and both reconstructed images. Thus, we assess the preservation of DRO's structures (Jaouen et al. 2018) in the reconstructed images without taking into account the post-reconstruction filtering.

2.3. Studied detectors

The studied detectors have a 12×12 SiPM custom readout, coupled to the scintillation block with optical grease. The scintillation block is a LYSO crystal with a base of 50 mm in X and Y dimensions. All its faces are polished and the surface in contact with the SiPM array is not coated in any configuration. Regarding the output of the detector, it consist of 24 channels, each being a signal proportional to the sum of a row or a column.



(a) **(b)** **(c)** **(d)**

Figure 4: Considered scintillator designs. D^B: light absorbing coating on all faces (a). D^T: teflon wrapping on all faces (b). D^R: light absorbing coating on all faces except the entrance face, which has a retroreflective device installed (represented in blue) (c). D^{R20}: same design as (c) but with a crystal height of 20 mm (d).

We selected four different detector designs that cover a range of design options. Additionally, they pose different challenges for a generic position determination algorithm, due to the differences in their inner reflection patterns. The four designs are depicted in figure 4 and defined as follows:

D^B light absorbing coating in all faces: decrease inner reflections in order to increase the accuracy of centrality and dispersion functions to estimate interaction positions. This has a negative impact in energy resolution, since a considerable amount of scintillation light is absorbed rather than detected.

D^T teflon wrapping on all faces: increase inner reflections in order to increase the energy resolution through a larger collection of scintillation light. This has a negative impact in the position estimation through centrality and dispersion functions, due to inner reflections.

D^R light absorbing coating on all faces except the entrance face, which has a retroreflective device installed: reach a compromise regarding inner reflections in a way that mitigates the degradation of centrality and dispersion estimates (*e.g.* retroreflective devices). This has a positive impact in energy resolution compared to the minimization of inner reflections.

D^{R20} same as D^R with a crystal thickness of 20 mm: increase the crystal thickness in order to increase the interaction count. This is the detector design used in the MindView system (Gonzalez et al. 2019, Gonzalez-Montoro et al. 2017).

All detectors, except D^{R20}, are 15 mm in Z, while D^{R20} is 20 mm. D^B is coated with black paint with an estimated absorption of 90%. D^T is wrapped with Teflon tape with no coupling component (air) between the Teflon and the crystal. D^R and D^{R20} have a retroreflective device in their entrance face, instead of paint. The retroreflective device is coupled with optical grease to the crystal in both cases.

We measured coincidences of an 11×11 array of ^{22}Na collimated sources with the previously described detectors. Measured data consisted of $\sim 2 \times 10^6$ coincidences for the array.

2.4. Clinical Data

Additionally, figure 4 (d) allows to test out method with phantom and patient data from the MindView system. Phantom data have been acquired from a Hoffman phantom filled with ~ 15 MBq of a solution of FDG and Gadolinium, and scanned for 20 minutes. Patient data have been acquired from a 72 year old dementia patient, with an injected dose of ~ 200 MBq, and scanned for 15 minutes ($85 - 100$ minutes after injection). In section 3, we compare image reconstructions using the built-in position determination of the MindView system, and using our proposed method.

2.5. Training Data

One of the main features of our proposed approach is to generate the training data from Monte Carlo optical simulations, instead of irradiating the scintillator at known positions. We use the GATE/GEANT4 (Agostinelli et al. n.d.) optical model (van der Laan et al. 2010) to define and run simulations of the previously described detectors. This has several advantages:

- The training dataset can be arbitrarily large.
- The simulation of each event can be executed in parallel.
- The labels (positions) for training are noiseless.

Moreover, we avoid the inclusion of inter- and intra-detector variability, mechanical inaccuracies or collimator scatter in the training data. This, in turn, prevents trained models (networks or other machine learning algorithms) from learning these errors as if they were part of the underlying model.

The simulation concerns a single detector and a spherical gamma source of 0.1 mm in radius. The description of the detector consists of the scintillator crystal, its coating, the photosensor array, and its coupling (see figure 4). Specifically, the gamma source emits 511 keV photons perpendicular to the entrance face of the crystal ($-Z$ direction). It is placed at a given XY coordinate at $Z = 100$ mm. The base of the crystal is placed at $Z = 0$. It is defined as LYSO material with a scintillation yield of 30/keV, an attenuation length of 1.2 cm for 511 keV, and a refractive index of 1.82. The photosensor array is in contact with the base of the crystal. It is defined as the *G4_SILICON_DIOXIDE* built-in material, with 4.36 mm pixel pitch, and an active area of 3 mm^2 per pixel. This models a high-density custom designed array of SiPMs from SensL (MindView-Series type, similar to J-series) (Jackson et al. 2014).

Both coating and coupling are modeled through boundary behavior. The simulation uses the *UNIFIED* model (Nayar et al. 1991, Levin & Moisan 1996) in GEANT4 for the reflection of photons at surfaces. The photosensor is coupled with optical grease to the crystal, and this is modeled at its boundary. We consider that every photon that leaves the crystal at this boundary is detected, so it is defined as a dielectric-metal boundary, since, in this case, only reflection and absorption (no refraction) are possible under the

UNIFIED model. These boundaries override the refractive index of the pixel to that of the optical grease (1.4), in order to model the coupling.

The coating is modeled at the rest of the boundaries of the crystal, which are in contact with air. Since the crystal is polished, a photon might suffer specular reflection before crossing a boundary. In the case of black paint, the boundary overrides the refractive index of air to that estimated for the paint (1.45). In the case of teflon tape, the refractive index is set to 1, since we expect a thin layer of air between the crystal and tape. If a photon crosses the boundary, we assume that it will be reflected back into the crystal with some probability, following Lambertian reflection rules. We experimentally estimate these probabilities for paint (0.1) and teflon (0.7) coatings, and implemented this behavior using both *groundbackpainted* and *reflectivity* properties of the *UNIFIED* model.

The retroreflective device cannot be modeled using a single boundary, thus, we include a layer of epoxy at the entrance face of the crystal. This 1 mm layer generates two boundaries: crystal-epoxy and epoxy-air. The first models a polished interface that allows a photon to remain inside the crystal through specular reflection. The latter is used if a photon leaves the crystal and always reflects the photon in the direction that it came from. This behavior is achieved by using the *backscatterconstant* and both *dielectric_metal* and *reflectivity* properties of the *UNIFIED* model. Since the retroreflective device is coupled to the crystal with optical grease, we modified the refractive index of the epoxy to 1.4.

Considering all instances, the XY coordinates of the gamma source range from -24 to 24 in 1 mm increments, so that the entire crystal is covered. Each instance has an autogenerated seed for the random engine and simulates events at different depths inside the crystal, according to its attenuation length. Several runs through the XY range allow a proper crystal depth coverage. In this work, we simulated $\sim$ 8 M and $\sim$ 9 M events, yielding an average of 216 and 185 events/mm^3, for each of the 15 and the 20 mm thick detectors respectively.

3. Results

3.1. Network Training

All networks are trained for 20 epochs while their performance is monitored through the *validation* dataset. This performance is shown in figure 5 as the absolute difference in RMSE between validations. We set an early stopping condition checking for an increase in RMSE between the i-th and the $(i-2)$-th validations (totalling 8 epochs of difference). In our case, this condition was never triggered in the first 20 epochs of training for any network.

Variation of RMSE decreases faster predicting XY than Z coordinates, as seen in figures 5 (a) and (b) respectively. This shows potential improvement in extending our epoch limit for Z coordinate predictions. However, as we show later in section 3.4, there

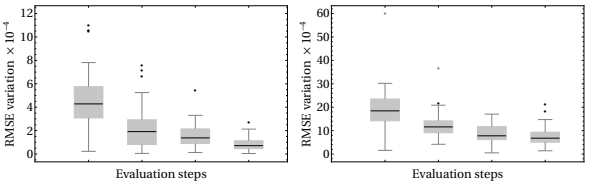


Figure 5: Training convergence expressed as the absolute difference in RMSE between consecutive validation steps. Results correspond to networks for all detector designs predicting XY (a) and Z (b) coordinates.

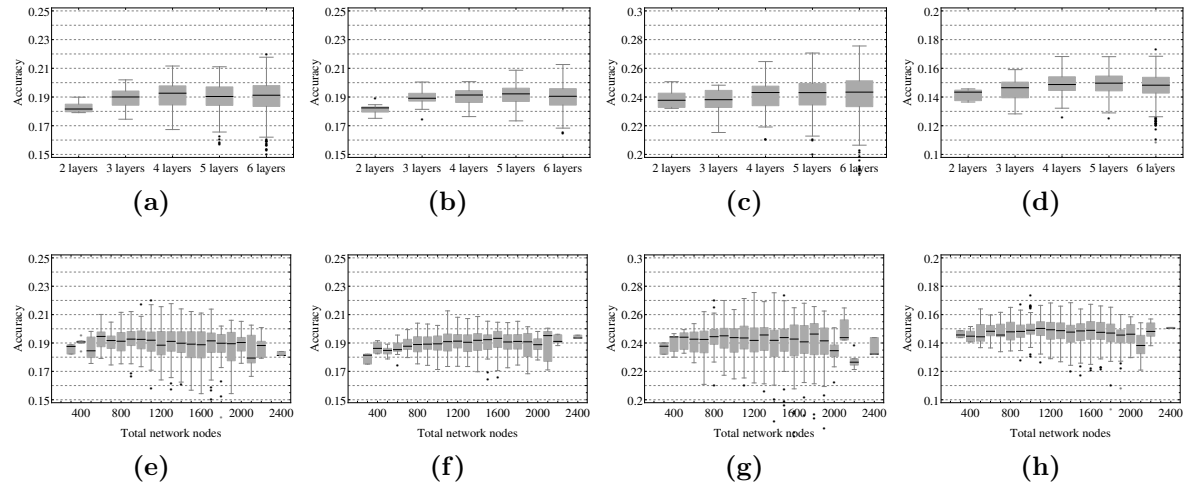


Figure 6: Acc_{sel} (10) within 1 mm as a function of hidden layers (left column) and total network nodes (right column). Each row corresponds to a different detector design: D^B (a) (e), D^T (b) (f), D^R (c) (g), and D^{R20} (d) (h). Results of (10) for every group of configurations are represented as box and whisker plots.

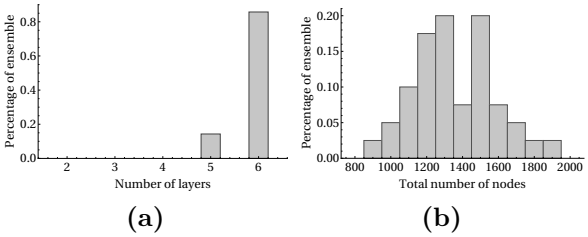
is an important region near the photosensors that cannot be properly determined due to scintillation light sampling.

3.2. Hidden Layers and Number of Nodes

Each configuration in the search space defines the hidden layers and nodes of a triple of networks (2) which are independently trained to predict each coordinate of the interaction point. We rank each triple of networks $\hat{f}_i$ to form the ensemble that will later be used for the interaction prediction (5*a*). Assuming that all $\hat{f}_i$ avoid large errors as much as possible due to their optimization during training to minimize RMSE, we are interested in including into the ensemble those $\hat{f}_i$ that generate as many predictions as possible *close* to the interaction point. Thus, the rank follows their Acc_{sel} (10) within 1 mm in the *test* dataset.

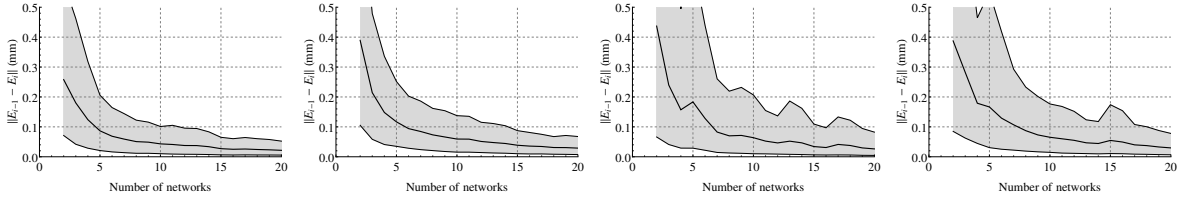
Results are summarized in figure 6 as a function of the number of layers and total number of nodes. The number of layers and nodes used in the selected ensembles of the four detector designs are shown in figure 7. The better-performing networks in our search space have around 1400 nodes distributed into 6 layers.

Regardless of the notable differences on the four detector designs, their best



(a) (b)

Figure 7: Number of layers (a) and total number of nodes (b) used in the networks of the selected ensembles.



(a) (b) (c) (d)

Figure 8: Ensemble convergence with D^B (a), D^T (b), D^R (c) and D^{R20} (d), represented as the variation of the ensemble's output as its number of networks increases. The variation shown at i corresponds to the distances between the predictions with the best $i-1$ networks and the predicted points including the next better performing network i. Grayed area includes the variation of 90% of the events (leaving out the 5% least and most varying predictions). Inner line corresponds to the average variation.

configurations do not differ much in terms of total number of nodes and layers. However, the Acc_{sel} (10) scores for each detector do vary between designs. On one hand, D^B and D^T yield similar results. On the other hand, D^R and D^{R20} show significant differences. While D^R is associated with the best results of all designs, D^{R20} shows a remarkable decrease in accuracy. Moreover, when grouped by layers, the Acc_{sel} (10) of D^{R20} converges after 4 layers, unlike the other detector designs. Figures 6 and 7 show potential improvement using more than 6 layers for all of the other designs.

3.3. Network Ensemble Size

Figure 8 shows the convergence of the ensemble average (5*a*) as N increases in the four studied detector designs. The convergence is represented by the distance between the predictions of ensembles with $i-1$ and i networks. These distances are obtained from the first 2×10^5 events in each detector measurement. In each case, below the upper bound are the lowest 95% differences predicting these events. For $N = 20$, 95% of predictions change less than $\tau = 0.1$ mm in all detectors.

3.4. Ensemble Testing

Once the ensemble is defined, its performance on predicting interaction coordinates is tested. The aim of these tests is to estimate the error associated with the approximate $\hat{f}$ at different regions of the crystal. There are two sources of error at this stage:

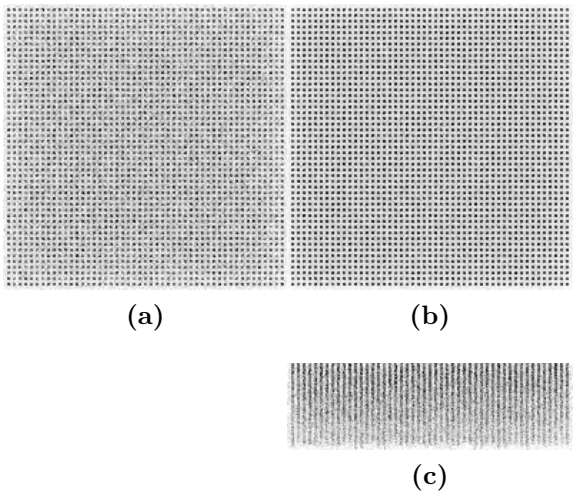


Figure 9: Interaction coordinates of the *test* dataset. Interactions are obtained from 49×49 sources located every 1 mm, from -24 to 24 in the XY directions, emitting gamma photons perpendicular to the entrance face of the crystal. This data correspond to D^B, but the other designs present similar results. Views correspond to an XY histogram including all $Z \in [0, 6]$ mm (a), an XY histogram including all $z > 6$ mm (b), and an XZ histogram including all $Y \in [-3, 0]$ mm (c). All histogram bins are 0.25 mm in width and height. Figures 10-13, and 17-20 follow the same representation.

a) from the networks that form the ensemble, *i.e.*, $\hat{f}(I)$ is not a good approximate of $f(I)$.

b) from the detector design, i.e., there is no detectable difference between I_1 and I_2 corresponding to two different interaction points $(x, y, z)_1$ and $(x, y, z)_2$.

In general, errors of the type a) are due to a wide variety of reasons. However, with this methodology, there could be two main causes: a.1) The training dataset does not include a proper sampling of f's domain or image, *i.e.*, does not include enough samples of possible Is or does not cover interactions in some regions of the crystal, respectively. In our case, this is prevented during the simulation of the training through the grid of sources, size of training, and Monte Carlo randomness. a.2) The network does not have sufficient nodes to approximate $f(I)$. This is prevented through the search of different configurations.

As an example of ground truth, figure 9 shows three views of the interaction coordinates in the *test* dataset (corresponding to the simulation of the D^B design, but nearly identical to the rest of the detector designs considered). Views consist on 2D histograms of the interaction coordinates through XY and XZ dimensions. A background of interactions is present due to the inclusion of Compton interactions in the simulation. In the ideal case, the output of the ensembles (or any other positioning algorithm) should be identical to figure 9.

Interactions predicted by the selected ensembles are shown in figures 10 for D^B, 11 for D^T, 12 for D^R, and 13 for D^{R20} designs. These figures include an additional histogram comparing the frequency of the simulated and predicted Z coordinates along the depth of the crystal. This frequency should decay exponentially along the crystal depth, reflecting the attenuation of the scintillator for 511 keV photons. Since in all of the studied designs use LYSO, the exponential decay in frequency along Z should maintain the same rate in all cases.

When comparing figure 9 (c) and figures 10-13 (d), we can observe a clustering of the predicted positions (around the centers of the 12 photosensors) in the lower regions of

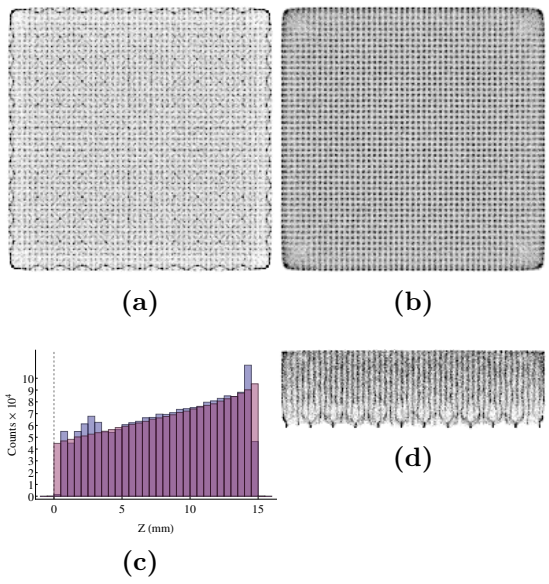


Figure 10: Prediction of *test* dataset for D^B. (a), (b) and (d) follow the same representation as figure 9. (c) shows an histogram of predicted Z coordinates (blue) and Z labels (red). Gray dashed line represents the photosensor plane.

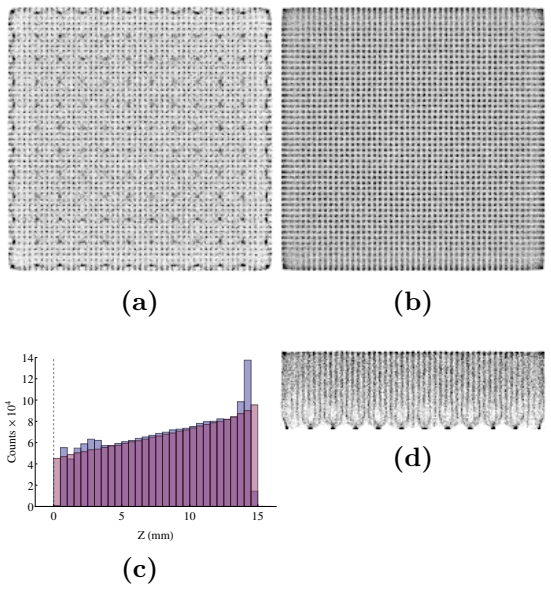


Figure 11: Prediction of *test* dataset for D^T. (a), (b) and (d) follow the same representation as figure 9. (c) shows an histogram of predicted Z coordinates (blue) and Z labels (red). Gray dashed line represents the photosensor plane.

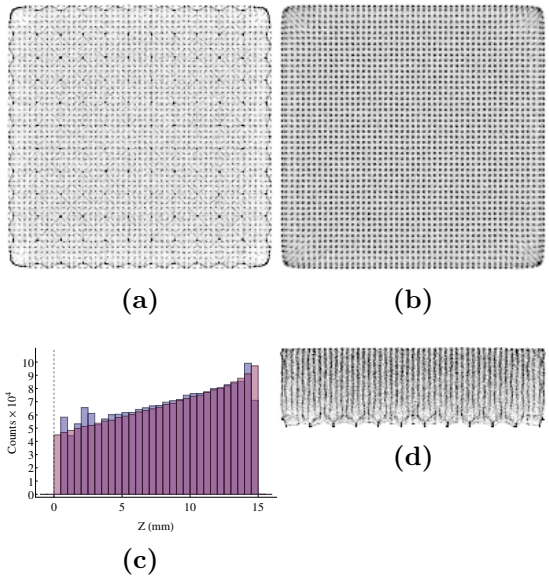


Figure 12: Prediction of *test* dataset for D^R. (a), (b) and (d) follow the same representation as figure 9. (c) shows an histogram of predicted Z coordinates (blue) and Z labels (red). Gray dashed line represents the photosensor plane.

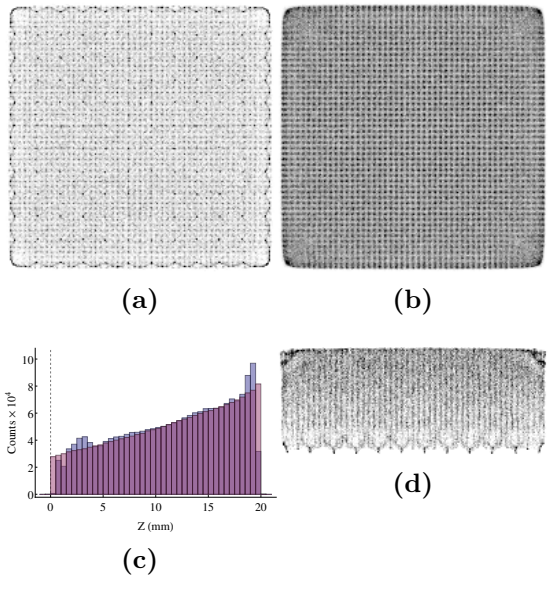


Figure 13: Prediction of *test* dataset for D^{R20}. (a), (b) and (d) follow the same representation as figure 9. (c) shows an histogram of predicted Z coordinates (blue) and Z labels (red). Gray dashed line represents the photosensor plane.

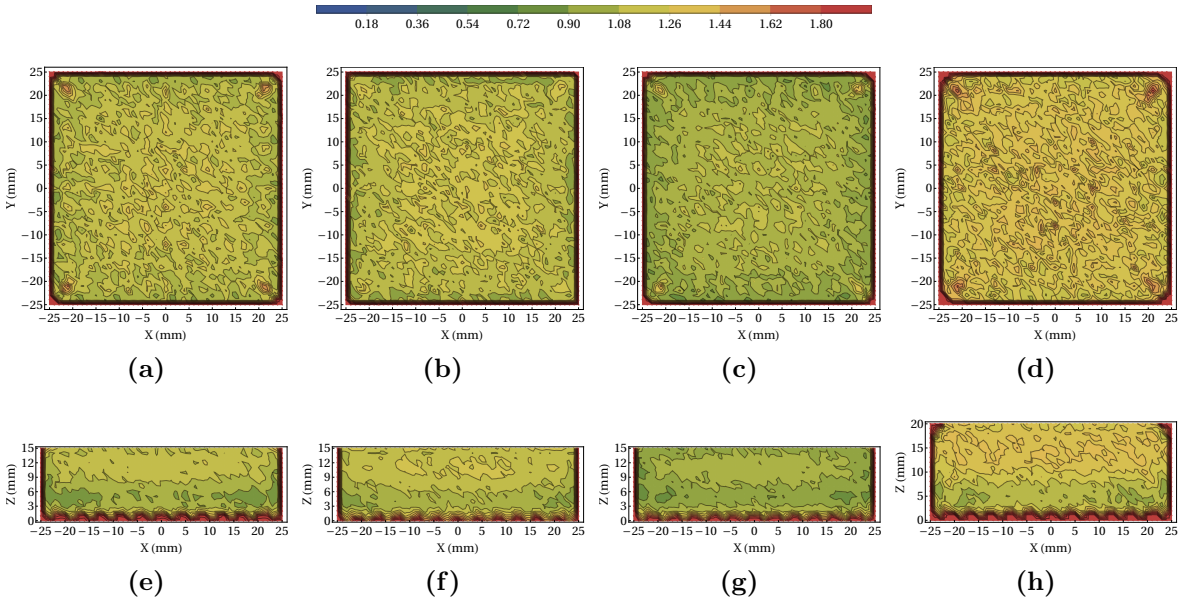


Figure 14: MAE of the *test* dataset for D^B (a)(e), D^T (b)(f), D^R (c)(g), and D^{R20} (d)(h). MAE is presented in ten contours corresponding to the legend on top of the figure.

figures 10-13 (d) (close to the photosensor plane). These errors are of type b), present in all designs and caused by the sampling of the scintillation light. For a given photosensor size, when an interaction occurs close enough to a photosensor, light measured by the rest of photosensors is marginal or zero. This difference in magnitude of the measurement of the photosensor below the interaction and the rest of the photosensors, makes nearby interactions (over the same photosensor) indistinguishable. The response of $\hat{f}$ is to output the center of the photosensor, minimizing the RMSE. Figures 10-13 (c) show the same error but in Z coordinate frequency near the photosensors in all designs. Figures 10-13 (c) show a systematic underestimation of the Z near the last millimeter before the entrance face of the crystal. While present in all designs, it is more significant in the D^T design (see figure 11 (c)).

There is also a systematic error in the XY edges of the crystal, noticeable when comparing figures 9 (a) and (b), and figures 10-13 (a) and (b). It can be seen how the first and last two rows and columns of sources are compressed towards the center of the crystal. Despite seeming a type a) compression artifact, figures 10-13 (b) and (d) images show that this compression stays constant along a wide range of Z. The reason behind this error is again the scintillation light sampling. Those two rows and columns occur past half of the first and last photosensor. Again, past that point, light measured by most of photosensors is marginal compared to light measured by the photosensor below the interaction. Thus, the response of $\hat{f}$ is driven by the loss function, providing the coordinate that minimizes the RMSE for all those indistinguishable inputs. An amplified version of this effect can be seen on the corners of the crystal, in which the difference between collected light by the different photosensors is even bigger. However, in this last case, the design of the detector is relevant, due to the proximity of crystal

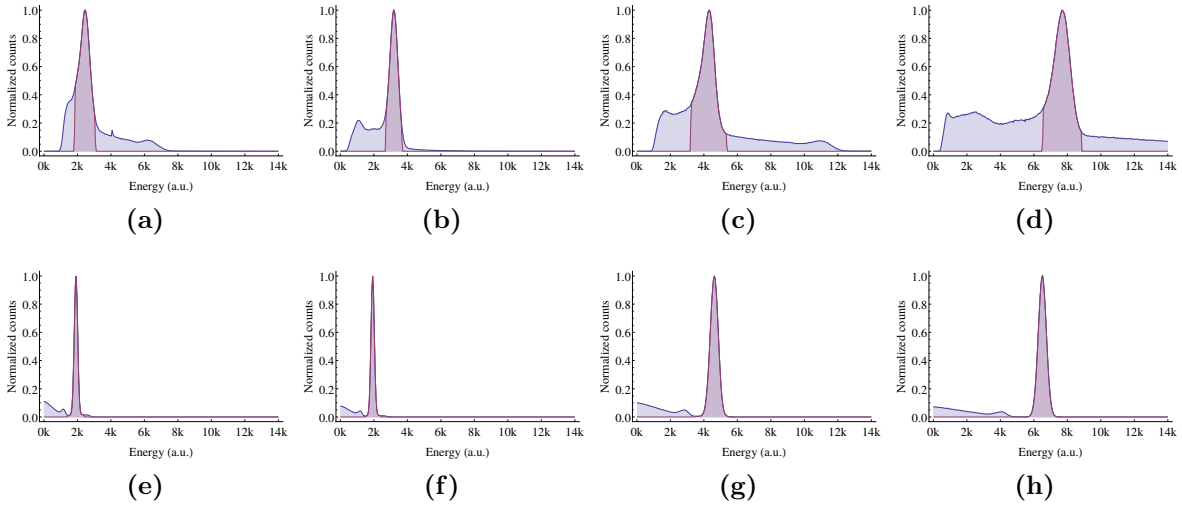


Figure 15: Energies of the four array measurements (top row) with energies of the simulated test dataset (bottom row). Each row correspond to a different design, namely, D^B (a)(e), D^{R20} (b)(f), D^R (c)(g) and D^T (d)(h). Red areas on left column correspond to the selected energy window for measurements. Red areas on right column correspond to the energy windows for measurements applied to the *test* dataset and are shown for reference.

faces that generate inner reflections. Specifically, D^T (amongst the studied designs) minimizes this *corner effect*.

Except for the previously described errors, the 1 mm spaced sources are distinguishable along the entire crystal, and the exponential decay in frequency of the Z coordinate matches the attenuation of the LYSO crystal and, thus, the simulation. Figure 14 presents a summary of the MAE for each detector design, where errors near photosensors, edges and corners are better seen. Additionally, figures 14 (e), (f), (g) and (h) show a noticeable increase in MAE for $z > 9$ mm in all cases, and a general increase in MAE when the geometry of the crystal is extended to 20 mm in depth. Considering the previous results, and specially figures 10-13 (b), the trained ensembles solve properly the positioning problem.

3.5. Ensemble Performance with Measured Data

Once the ensemble is tested, we determine its suitability for measured data, or equivalently, we test if measured interactions are similar to simulated populations used during training. For each design, we estimate the interaction positions of an array of 11×11 collimated sources. Figure 15 shows the energy spectrum of each measurement and *test* dataset (for reference). Each energy window has been obtained by fitting the energy peak to a normal distribution, and defining lower and upper limits around its mean μ as $[\mu - w\mu, \mu + w\mu]$. Specifically, for D^B and D^R, $w = 0.25$ and for D^T and D^{R20}, $w = 0.15$.

Interaction positions for the measured arrays are summarized in figures 17-20, following the representation of previous figures. We include the obtained FWHM after

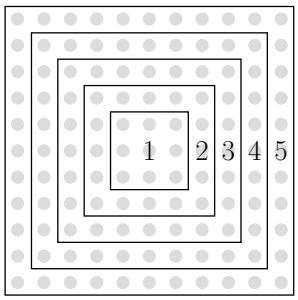


Figure 16: Sources included at each distance to the center regarding the computation of the FWHM in figures 17-20 (d).

fitting each source to a Gaussian distribution, grouped by the distance of each source to the center of the detector. The different groups of sources are illustrated in figure 16. Since we lack of a ground truth for the measurements, frequencies of Z coordinate in figures 17-20 (e), are compared to

$$Ae^{-0.082z}\left(\operatorname{erf}\left(\frac{b-z}{\sqrt{2}\sigma}\right)-\operatorname{erf}\left(\frac{a-z}{\sqrt{2}\sigma}\right)\right), \tag{15}$$

where 0.082 mm^{-1} is the attenuation coefficient of LYSO for 511 keV, a and b are the limits for DOI measurement, and σ is proportional to the DOI resolution. We fixed $a = 1$, $b = 14$ (19 in the case of D^{R20}) and $\sigma = 0.5$, while we obtained a least squares fit for A. Figures 17, 18 (e), corresponding to D^B and D^T, obtain similar results. Both follow (15) closely, which in addition to figures 17, 18 (c), indicates good DOI prediction capabilities.

In case of figures 19, 20 (e), corresponding to D^R and D^{R20}, there is a systematic overestimation of the Z coordinate. This is represented as an increase in frequency between $[10, 15]$ and $[15, 20]$ mm respectively. Considering that the conditions of the measurement do not change from D^B or D^T, and the simulation definition of the absorbent coating and photosensor coupling is identical to that on D^B, this overestimation of the Z coordinate must be due to differences between simulated and physical behaviors of the retroreflective device. There is also a systematic error in the Z coordinate shown in figure 19 (e) in frequencies between $[5, 9]$ mm. Even though, the photosensor coupling is identical in all four designs and simulation definition is almost identical between D^R and D^{R20} (only varying with the depth of the crystal), this defect is only present in D^R. Additionally, it was not detected in any of the 60 D^{R20} detectors on the MindView system either. Thus, we attribute it to an unexpected behavior in the particular detector used to implement the D^R design.

Regarding XY coordinates, figure 21 compares the centers of the sources predicted by the ensembles (obtained from fitting) and their real position. Unlike in the simulated *test* dataset, D^B and D^T show different performance, obtaining respectively 11% and 45% of sources at least 1 mm away of their real position. The lower half of the D^T prediction contains most of the positioning error. Let the rows and columns of the array be numbered from 1 to 11, top to bottom and left to right respectively. Rows 7, 8 and 9 suffer compression towards row 10 (which is closer to the edge) instead of towards the center of the crystal. The same behavior can be observed in columns 8 and

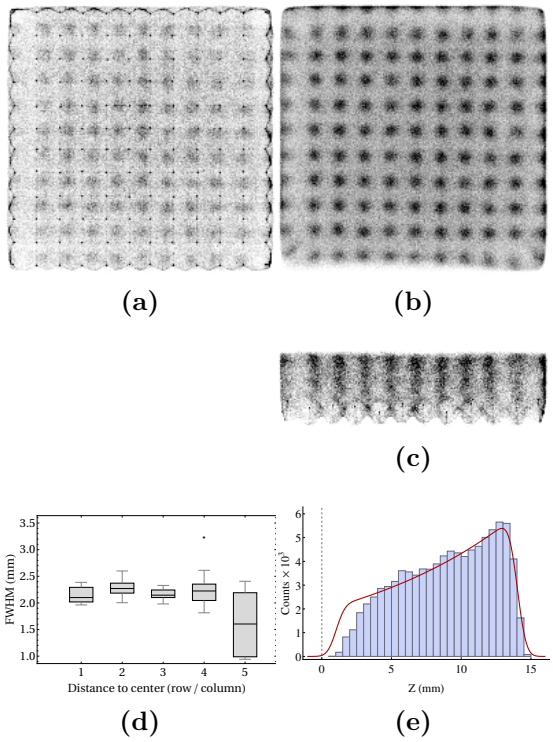


Figure 17: Prediction of an 11 × 11 array of sources with D^B (a) (b) (c), including FWHM (d) and DOI frequency (e).

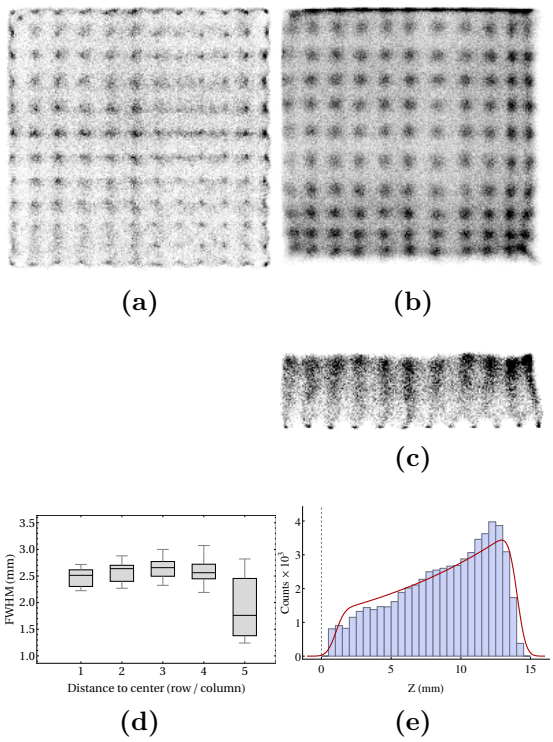


Figure 18: Prediction of an 11 × 11 array of sources with D^T (a) (b) (c), including FWHM (d) and DOI frequency (e).

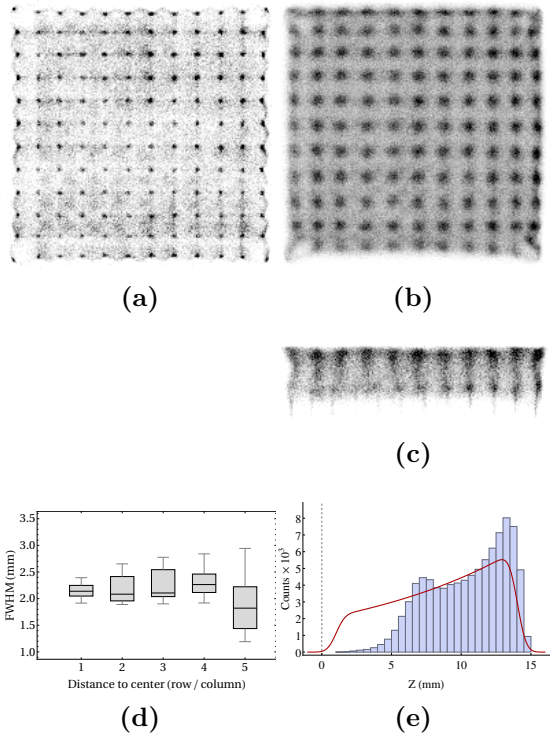


Figure 19: Prediction of an 11 × 11 array of sources with D^R (a) (b) (c), including FWHM (d) and DOI frequency (e).

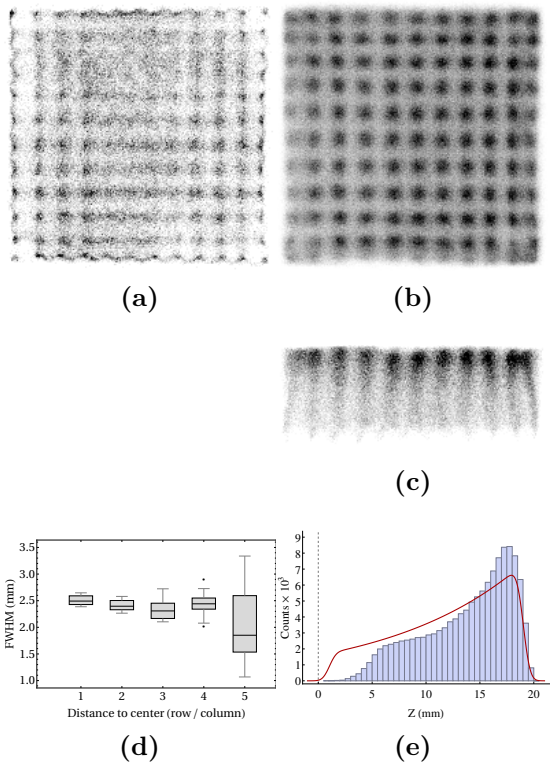
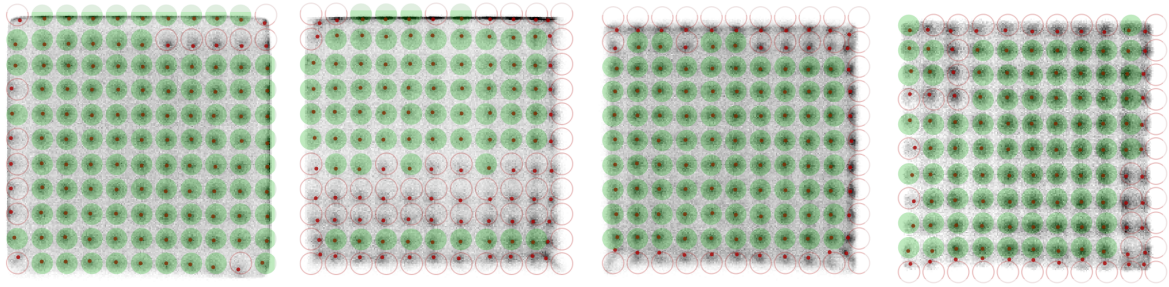
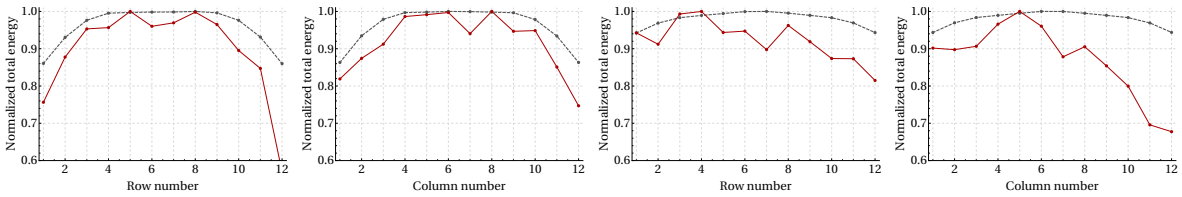


Figure 20: Prediction of an 11 × 11 array of sources with D^{R20} (a) (b) (c), including FWHM (d) and DOI frequency (e).



(a) **(b)** **(c)** **(d)**

Figure 21: Predicted arrays with D^B (a), D^T (b), D^R (c) and D^{R20} (d). Each array has an overlay with the fitted center of each predicted source (red dot) and a 2 mm radius circle centered at the real position of the source. Circles are presented as green disks for predicted sources closer than 1 mm to the real position, and as red circles otherwise.



(a) **(b)** **(c)** **(d)**

Figure 22: Normalized sum of the output of each row and column for D^B (a) (b), and D^T (c) (d). Gray dashed lines show the sum corresponding to both simulated *test* datasets, while red lines correspond to both array measurements.

9, without observable symmetry in second row and column. Moreover, this effect is not present on the other detector designs.

Figure 22 shows total collected energy by rows and columns regarding D^B and D^T. It compares the normalized sums of the outputs of simulation and measurement. Intra-detector variability is shown as a mismatch between the output of rows and columns, as well as between the relative values of their sums. For example, in figure 22 (a), rows 6 and 7 should present higher values. While D^B shows a close match to the simulation, D^T has important differences in rows $7-12$, but especially in columns $7-12$. These columns correspond to the Y coordinate in the lower part of figure 21 (b), which shows the misplacements of sources. This suggests that the cause of the different performance between D^B and D^T is related to an unexpected behavior in the D^T detector.

Table 1 summarizes the results obtained from measurements and *test* datasets for each detector design. Differences between physical and simulated results have a strong influence in the obtained FWHM, with respect to the expected Accuracy based on the test datasets. Results on interactions closer to the photosensors (collected from the *test* datasets) show that neural networks can take advantage of inner reflections to better estimate the interaction position in that range. Even though differences between simulated and measured interactions notably influenced the results, the obtained network ensembles show reasonable outputs. Specifically, in the case of D^T, systematic

Table 1: Ensemble performance for the scintillator designs. Accuracy (11) and MAE (8) correspond to the *test* dataset. Average FWHM are obtained from fitting all sources in the measured arrays. Results are divided in three blocks, the first three rows (bold) correspond to data where $6 < z$ while the two middle rows correspond to data where $z \leq 6$ mm. The last two rows account for all data.

Figure of merit		D^B	D^T	D^R	D^{R20}
Accuracy (11)	%	**68.7**	**67.5**	**72.1**	**63.6**
MAE (8)	mm	**1.11**	**1.15**	**1.00**	**1.28**
Average FWHM	mm	**2.02**	**2.37**	**2.11**	**2.29**
Accuracy (11)	%	56.7	59.2	59.6	53.5
MAE (8)	mm	1.22	1.16	1.13	1.36
Accuracy (11)	%	64.9	64.9	68.1	61.6
MAE (8)	mm	1.15	1.15	1.04	1.30

deviations between 10-30% in energy on some channels had a moderate impact (< 2 mm) in the prediction of the interaction positions.

3.6. Image Reconstruction using Network Ensembles Position Estimates

In this section we show that one trained ensemble can be used to estimate interaction coordinates on any detector that follows the same design. Using data from the MindView system, which uses 60 monolithic detectors with the D^{R20} design, we predict the interaction coordinates in all its detectors with the previously analyzed D^{R20} network ensemble. Predicted coordinates are serialized in a list mode file for reconstruction. We compare reconstructed images from the MindView's native positioning system (see section 2) and the D^{R20} network ensemble. All image reconstructions were performed using the open source CASToR (Merlin et al. 2018, *Customizable and Advanced Software for Tomographic Reconstruction (CASToR)* 2017) platform with OSEM (6 iterations, 12 subsets), 1 mm^3 voxels, with the Siddon projector and 2 mm Gaussian post-filtering.

Figure 23 shows reconstructed images from a Hoffman phantom. (a) and (d) columns correspond to the DRO after applying a 2 mm Gaussian filtering. (b) and (e) correspond to the ensemble and (c) and (f) columns to centrality based reconstructions. Figure 24 shows line profiles through these images. The line profiles include the filtered (black line) and original (gray area) DRO values. Included arrows point to cases in which D^{R20} network ensemble leads to better peak to valley ratios. These profiles, show that the improvements occur near fine structures, where the effect of parallax error is best appreciated. Table 2 shows the obtained results with the metrics introduced in section 2.2. While we observe a slightly better SSIM score, there are clear improvements in SNR and CNR when considering the ensemble network reconstruction over the centrality estimate.

We provide a similar example in figure 25, showing reconstructed images from a patient, in order to show that the position estimate quality is maintained under more

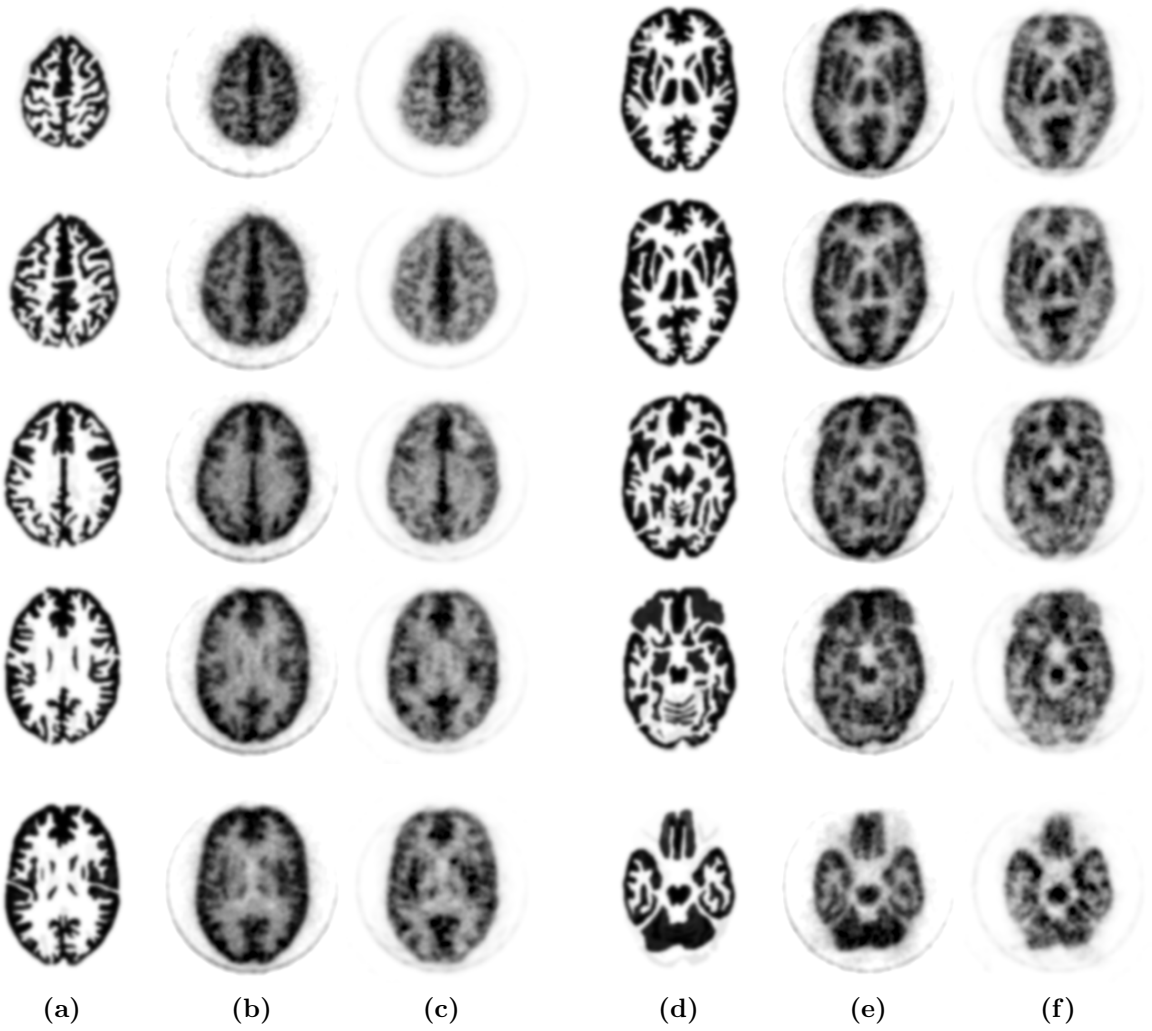


Figure 23: Different slices from reconstructed images of a Hoffman phantom. Columns (a), (d) correspond to the DRO, (b), (e) correspond to the D^{R20} ensemble, and (c), (f) to interaction position determination based on centrality estimates. All images have been normalized by linear histogram stretching with saturation of the 0.3% highest values.

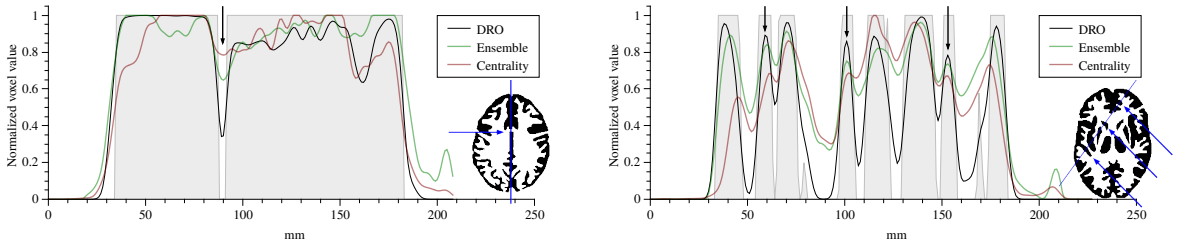


Figure 24: Line profiles of slices in figure 23, columns (a), (b) and (c), row 3 (left), and columns (d), (e) and (f), row 1 (right). Grayed area represents the value of the original DRO, while black curve is its profile after applying a 2 mm Gaussian filter.

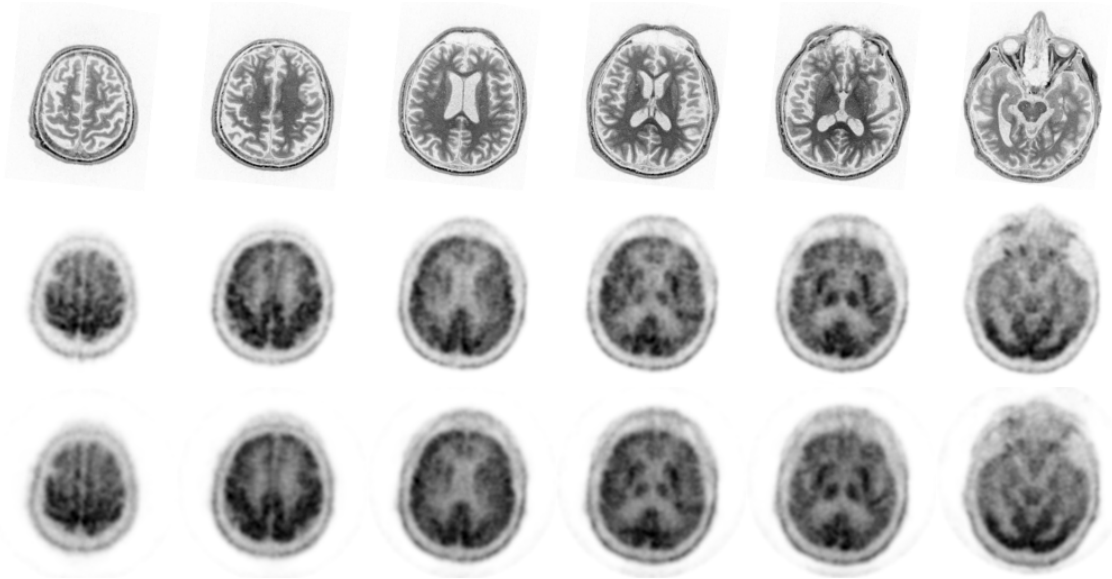


Figure 25: Different slices from reconstructed images of a patient. Top row corresponds to MP RAGE MRI, central row to the D^{R20} ensemble, and bottom row corresponds to interaction position determination based on centrality estimates.

Table 2: Quantitative metrics (12), (13) and (14) obtained from the DRO and phantom reconstructions.

Positioning algorithm	SNR (dB)	CNR	SSIM
Ensemble (D^{R20})	9.56	3.487	0.849
Centrality based	5.99	2.497	0.833

realistic conditions (patient involuntary movement, etc). Reconstruction parameters and normalization procedures remain the same over all images. Top row of figure 25 shows a simultaneously acquired rapid gradient-echo (MP RAGE) MRI. Central and bottom rows of figure 25 correspond to the ensemble network and centrality estimates respectively. Again, visual inspection reveals better contrast with the ensemble network in some fine details.

The MindView built-in positioning system calibrates each detector separately, correcting for inter- and intra-detector variability. The calibration is conducted on arrays of sources, which events are positioned by centrality estimates and corrected to match the real positions in the array. However, to be able to separate and locate the sources in the centrality estimates, these sources have to be sufficiently distant in the array. Additionally, the behavior of the detector is interpolated between each source, and at different DOIs. Specifically in this case, the behavior of the last 4 mm to the edge of the detector is defined from the correction of a single source. Although the network ensembles cannot correct for intra-detector variability, taking advantage of a Monte Carlo generated training provides a better sampling of the detector behavior. Moreover, Monte Carlo reference data offers simultaneous and accurate (x, y, z) information for

each interaction.

4. Conclusions

Since 2010, there has been active development in the optical simulation capabilities of the GATE platform, that led to the inclusion of measured surfaces obtained by atomic force microscopy, among other platform capabilities. The combination of these developments with current machine learning techniques leads to a robust and accurate approach to obtain the position of interaction in monolithic detectors, with minimum implementation effort and no required per-detector calibration.

In this work we propose a methodology to obtain such positioning estimator through an ensemble of multilayer feedforward neural networks, trained with data generated from an optical Monte Carlo simulation. Our proposed approach is designed to work in general with a cuboid-shaped monolithic scintillator on top of an array of photosensors. We provide simulation parameters for four different detector designs, varying in surface treatment and size. Additionally, we propose an event generation setup and training size. We describe an easy-to-implement, robust network architecture, which can be completely built using popular machine learning frameworks, such as TensorFlow, with minimum implementation effort. We propose an heuristic to find the set of best hyper-parameters for a given design and form the ensemble of networks that estimates the interaction position. We show the capability of the ensemble to solve the positioning problem considering the inherent sampling problems of monolithic detectors. We test the obtained ensembles with measurements from physical detectors to verify validity of the simulated training. Finally, we use one of the proposed ensembles to estimate the interaction positions in the MindView scanner, comparing the reconstructed images, to verify the applicability of a single ensemble to 60 detectors (not involved in any step of the training). Reconstructed images are compared to those obtained with the built-in positioning system of the MindView scanner.

Results show that Monte Carlo training is valid for an ensemble to estimate interaction positions, including DOI. Furthermore, despite deviations from the simulated behavior due to intra-detector variability (with systematic deviations between 10-30% in gain of several channels), the ensembles of networks provide a robust 3D positioning output (with errors of < 2 mm). With the studied detectors, network ensembles resolve arrays of sources with average FWHMs of 2-2.4 mm depending on the design. Regarding the DOI, the ensembles show a robust output, being remarkably accurate for some designs. Reconstructed images based on ensemble estimations show improved SNR, CNR and SSIM when compared to those based on the MindView built-in more typical positioning algorithm.

References

Abadi M, Agarwal A, Barham P et al. 2015 'TensorFlow: Large-scale machine learning on heterogeneous systems'. Software available from tensorflow.org.

Agostinelli S, Allison J, Amako K et al. n.d. GEANT4–a simulation toolkit *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* (3), 250–303.

Anger H O 1958 Scintillation Camera *Review of Scientific Instruments* **29**(1), 27–33.

Borghi G, Tabacchini V, Seifert S & Schaart D R 2015 Experimental Validation of an Efficient Fan-Beam Calibration Procedure for k-Nearest Neighbor Position Estimation in Monolithic Scintillator Detectors *IEEE Transactions on Nuclear Science* **62**(1), 57–67.

Bruyndonckx P, Lemaitre C, van der Laan D J et al. 2008 Evaluation of Machine Learning Algorithms for Localization of Photons in Undivided Scintillator Blocks for PET Detectors *IEEE Transactions on Nuclear Science* **55**(3), 918–924.

Bruyndonckx P, Leonard S, Tavernier S et al. 2004 Neural network-based position estimators for PET detectors using monolithic LSO blocks *IEEE Transactions on Nuclear Science* **51**(5), 2520–2525.

Customizable and Advanced Software for Tomographic Reconstruction (CASToR) 2017. www.castor-project.org.

Duchi J, Hazan E & Singer Y 2011 Adaptive Subgradient Methods for Online Learning and Stochastic Optimization *Journal of Machine Learning Research* **12**(Jul), 2121–2159.

González A, Conde P, Iborra A et al. 2015 Detector block based on arrays of 144 SiPMs and monolithic scintillators: A performance study *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **787**, 42–45.

Gonzalez A J, Gonzalez-Montoro A, Vidal L F et al. 2019 Initial Results of the MINDView PET Insert Inside the 3T mMR *IEEE Transactions on Radiation and Plasma Medical Sciences* **3**(3), 343–351.

Gonzalez-Montoro A, Aguilar A, Canizares G et al. 2017 Performance Study of a Large Monolithic LYSO PET Detector With Accurate Photon DOI Using Retroreflector Layers *IEEE Transactions on Radiation and Plasma Medical Sciences* **1**(3), 229–237.

Hansen L & Salamon P 1990 Neural network ensembles *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**(10), 993–1001.

Hornik K 1991 Approximation capabilities of multilayer feedforward networks *Neural Networks* **4**(2), 251–257.

Hornik K, Stinchcombe M & White H 1989 Multilayer feedforward networks are universal approximators *Neural Networks* **2**(5), 359–366.

Ioffe S & Szegedy C 2015 Batch normalization: Accelerating deep network training by reducing internal covariate shift *International Conference on Machine Learning* pp. 448–456.

Jackson C, O'Neill K, Wall L & McGarvey B 2014 High-volume silicon photomultiplier production, performance, and reliability *Optical Engineering* **53**(8), 081909.

Jaouen V, Bert J, Boussion N et al. 2018 Image enhancement with PDEs and nonconservative advection flow fields *IEEE Transactions on Image Processing* .

Lecoq P 2016 Development of new scintillators for medical applications *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **809**, 130–139.

Levin A & Moisan C 1996 A More Physical Approach to Model the Surface Treatment of Scintillation Counters and its Implementation into DETECT *1996 IEEE Nuclear Science Symposium. Conference Record* Vol. 2 IEEE pp. 702–706.

Llosá G, Barrillon P, Barrio J et al. 2013 High performance detector head for PET and PET/MR with continuous crystals and SiPMs *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **702**, 3–5.

Llosá G, Barrio J, Lacasta C et al. 2010 Characterization of a PET detector head based on continuous LYSO crystals and monolithic, 64-pixel silicon photomultiplier matrices *Physics in Medicine and Biology* **55**(23), 7299–7315.

Maas M C, Schaart D R, van der Laan D J J et al. 2009 Monolithic scintillator PET detectors with intrinsic depth-of-interaction correction *Physics in Medicine and Biology* **54**(7), 1893–1908.

Marcinkowski R, Mollet P, Van Holen R & Vandenberghe S 2016 Sub-millimetre DOI detector based on monolithic LYSO and digital SiPM for a dedicated small-animal PET system *Physics in Medicine and Biology* **61**(5), 2196–2212.

Merlin T, Stute S, Benoit D et al. 2018 CASToR: a generic data organization and processing code framework for multi-modal and multi-dimensional tomographic reconstruction *Physics in Medicine & Biology* **63**(18), 185005.

Muller F, Schug D, Hallen P, Grahe J & Schulz V 2018 Gradient Tree Boosting-Based Positioning Method for Monolithic Scintillator Crystals in Positron Emission Tomography *IEEE Transactions on Radiation and Plasma Medical Sciences* **2**(5), 411–421.

Muller F, Schug D, Hallen P, Grahe J & Schulz V 2019 A novel DOI Positioning Algorithm for Monolithic Scintillator Crystals in PET based on Gradient Tree Boosting *IEEE Transactions on Radiation and Plasma Medical Sciences* pp. 1–1.

Nair V & Hinton G E 2010 Rectified linear units improve restricted boltzmann machines *Proceedings of the 27th international conference on machine learning (ICML-10)* pp. 807–814.

Nayar S, Ikeuchi K & Kanade T 1991 Surface reflection: physical and geometrical perspectives *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**(7), 611–634.

Pani R, Others, Vittorini F et al. 2009 Revisited position arithmetics for LaBr3: Ce continuous crystals *Nuclear Physics B - Proceedings Supplements* Vol. 197 pp. 383–386.

Schaart D R, van Dam H T, Seifert S et al. 2009 A novel, SiPM-array-based, monolithic scintillator detector for PET *Physics in Medicine and Biology* **54**(11), 3501–3512.

Srivastava N, Hinton G, Krizhevsky A, Sutskever I & Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *The Journal of Machine Learning Research* **15**(1), 1929–1958.

van Dam H T, Seifert S, Vinke R et al. 2011 Improved Nearest Neighbor Methods for Gamma Photon Interaction Position Determination in Monolithic Scintillator PET Detectors *IEEE Transactions on Nuclear Science* **58**(5), 2139–2147.

van der Laan D J J, Schaart D R, Maas M C et al. 2010 Optical simulation of monolithic scintillator detectors using GATE/GEANT4 *Physics in Medicine and Biology* **55**(6), 1659–1675.

Wang Y, Zhu W, Cheng X & Li D 2013 3D position estimation using an artificial neural network for a continuous scintillator PET detector *Physics in Medicine and Biology* **58**(5), 1375–1390.

Wang Z, Bovik A, Sheikh H & Simoncelli E 2004 Image Quality Assessment: From Error Visibility to Structural Similarity *IEEE Transactions on Image Processing* **13**(4), 600–612.