Investigations of Scatter Correction Methods in Quantitative PET using Deep Learning

Konstantinos Drymas Vrakidis

MSc Student, Medical Imaging, Utrecht University
Research Intern, Department of Radiology and Nuclear Medicine, UMC Utrecht
E-mail: k.d.vrakidis@students.uu.nl, Student no.: 3187909

Supervisors: Woutjan Branderhorst dr., Hugo W.A.M. de Jong prof. dr. ir.

Abstract

Introduction: To render a reconstructed image acquired with PET as quantitatively accurate, corrections for the scattered coincidences have to be applied. Their contribution tends to introduce a low-frequency additive component to the acquired data, which ultimately leads to loss of contrast and erroneous SUV measurements in the reconstructed images. Despite Monte Carlo (MC)-based methods being considered the most accurate for scatter corrections, their computational demands prevent them from being clinically in use. In this work, the use of Deep Learning (DL) was investigated as a method to provide MC-grade scatter corrections within clinical timeframes.

Materials & Methods: MC simulations of two types of phantoms, 9 analytical and 24 voxelized-patients, were performed in GATE. With the resulting sinogram data of prompt coincidences and attenuation factors as input and the scattered coincidences as output, a 2D U-Net was trained. Independent network trainings were performed on 18 unique datasets. Each of them was constructed by using a different subset of simulated phantoms, a different input-output pre-processing, and a different 2D view over the same sinogram data. The performances of the trained networks were evaluated on a test dataset, which was consisted of 5 simulated phantoms excluded from the trainings, using Normalized Root Mean Squared Error (NRMSE) as a metric.

Results: The best-performing network achieved an NRMSE (mean $\pm$ standard deviation) of (4.94 ± 1.88) % overall, and (3.91 ± 1.21) % specifically for voxelized patient phantom cases. It was trained using the projection views of the sinogram data, with no input-output blurring. With input-output blurring applied, comparable results were obtained. Full scatter estimations of a single bed position were generated within 4.8 seconds. In 64% of the test cases, using the projection views of the sinogram data resulted in a lower mean NRMSE. The inclusion of analytical phantoms decreased the performance of the network on the voxelized-phantom tests by an NRMSE of 1% on average.

Conclusions: The feasibility of using DL for scatter estimation can be claimed. Improved accuracy is achieved by using the projection views instead of the sinogram views for trainings. Valid DL methods to generate a scatter estimation can be based on both unprocessed and blurred MC-generated training data. Which of the two constitutes the optimal strategy remains inconclusive, as their quantitative accuracy must be evaluated on the final reconstructed images.

Keywords: Positron Emission Tomography, Scatter Corrections, Monte Carlo Simulations, Deep Learning

1. Introduction

Over the last decades, Positron Emission Tomography (PET) has proven to be one of the most acclaimed tools for non-invasive functional imaging techniques through its successful implementation in applications of clinical diagnostics (oncology, cardiology, neurology, and psychiatry) and many preclinical studies. To compensate for physical phenomena that affect the acquisition in PET and to render a reconstructed image as quantitatively accurate, several corrections need to be applied to the acquired data. Among

these, the corrections for the falsely registered coincidence events due to scatterings have to be applied.

In principle, image reconstruction in PET is based on the assumption that from the detection of two photons within a short time frame, we can assign the location of a positron-electron annihilation between the two detectors. This assumption is violated when at least one of the two annihilation photons undergoes at least one Compton scattering process, changing their original trajectory. A successful detection of such coincidence will not be representative of the activity distribution between the detectors.

In clinical scans, the contribution of scattered events is estimated to be in the range of 30-40% of the prompt events and 20-200% of true events [1], depending on the bed position. Simulations show that 90% of them are caused by scatterings happening within patients, with the rest due to the hardware components of the scanner and the bed [2].

In the reconstructed images, this contribution tends to introduce a low-frequency additive component with little structural information [1], which is more prominent towards the center of the image [1]. This ultimately leads to lower image contrast and erroneous SUV measurements.

Techniques to estimate the scatter contribution and correct its effects have been developed. Generally, these are characterized by a trade-off between the accuracy and the computational cost of the estimation. The ones most commonly used are based on Single Scatter Simulation (SSS) algorithm [3, 4], and the Monte Carlo Scatter simulations (MCS) [5].

SSS is a model-based scatter correction algorithm that estimates the scatter distribution using analytical calculations. These involve Compton-scattering cross-sections and ray tracing through the reconstructed image volumes of the emission (PET) and the transmission (CT or pseudo-CT) scans. For subsequent scatter correction the estimated distribution of scatter events can be incorporated during (iterative) image reconstruction.

In 3D PET, these operations require several simplifications to achieve a scatter estimation within clinical timeframes. Among them are the omissions of the contributions due to multiple scatterings and due to from Out-of-Field of View (OoFoV) activities. To compensate for these, an error-prone scaling of the scatter contribution is applied to match the acquired data, which tends to overestimate the scatter contribution [6]. Inaccuracies in scatter correction using SSS are reportedly causing severe photopenic (halo-) artifacts to the reconstructed images. That is especially prominent around anatomies with typically high activity concentrations, such as the bladder and the kidneys [7].

Improvements to the SSS algorithms have been proposed such as the MC-SSS, which uses low-count Monte-Carlo simulations to estimate more accurate scaling factors [8], or the inclusion of double-scattered events in the calculations [9]. These however are sophisticated models that increase the computational cost.

Conversely, the MCS method can successfully address these weaknesses of SSS [7]. Due to its ability to include the physical and stochastic nature of the entire acquisition, it is considered the most accurate estimation method of the contribution of scatter. However, this comes with a high computational cost, which renders it unusable for clinical workflows. Acceleration methods by running the MC simulations on Graphics Processing Units (GPUs) can be found [10] but implementation on clinical scans with high enough counts has not been reported yet.

Recent advancements in GPUs have also enabled rapid progress in Machine Learning (ML) and Deep Learning (DL). Tasks in medical imaging, such as classification and regression, have become a source of inspiration for the development of various architectures of Neural Networks (NNs). Especially Convolutional NNs (CNNs), through their superior capability to recognize patterns in images, can be considered the primary type of NNs used in medical imaging. The most popular architecture of such CNNs is the U-Net [11], with huge success in image segmentation tasks. ML and DL methods for PET imaging have also been actively investigated, as a means to obtain faster and quantitatively more accurate PET images with lower doses. Applications vary from smaller PET instrumentation and image acquisition tasks to solving the image reconstruction inverse problem with a sinogram-to-image mapping [12].

Scatter correction using DL has also been a target for investigations. Joint attenuation and scatter correction by training with corrected and uncorrected reconstructed images has been proposed [13,14]. This could potentially compensate for the lack of transmission scanning in hybrid PET/MRI systems. Correction methods exclusively for scatter using CNNs can be found, such as the use of a 2D U-Net to directly generate projections of the scatter contribution from emission and transmission projections [15]. While the accuracy and the speed of the inferences were promising, it was trained only on SSS as the scatter ground truth. This leaves the evaluation of the performance of a CNN on MCS an open question.

This study aims to continue the investigations regarding the use of a U-Net for supervised learning, by using the MC-estimated scatter contributions as training targets. Such development could be a step towards a more accurate and applicable to clinical workflow scatter estimation method since it can potentially combine the accuracy of MCS with the reported inference speed of CNNs. For that purpose, analytical and patient-based phantoms were MC simulated for the generation of training datasets consisting of emission, transmission, and scatter events. Trainings using different views (sinograms and projections) of the same projection data were independently examined. Issues raised by the nature of

the MC-generated data are addressed, such as the noise & the sparsity. These have a big impact on the training process of such networks, and finding suitable preprocessing methods could be valuable for future studies using similar data

2. Methods

Monte Carlo simulations for various phantoms were performed to emulate PET acquisitions and generate their corresponding scatter contributions. For the preprocessing of the data, different strategies were followed, creating distinct datasets with which NNs were trained independently. The performances of the trained networks were evaluated on a set of data that has been excluded from the trainings. The following sections cover the steps we took to accommodate our investigations in detail.

2.1 Simulated Phantoms

In the MC simulations, two types of phantoms were used. Voxelized phantoms based on patient scans and analytically-modeled ones.

For the first, three bed positions were cropped from each of 8 different whole-body 18F-fluorodeoxyglucose (FDG) PET/CT patient scans, creating a set of 24 voxelized phantoms in total. To reduce the computation time, the axial size of the phantoms was limited to 40 cm, from which only the central 14 cm belong to the scanner's FoV (see 2.2).

Typically challenging body sites for SSS are located on the lower torso, due to the high activity concentrations in organs of the renal system, such as the bladder and the kidneys. Cases of the bladder being located outside the scanner's axial FoV but close to its boundaries can be considered as good cases of high scatter to prompts ratios. Additionally, conventional methods can only compensate for scatter originating outside of the FoV, and not directly estimate it. To investigate the response of a DL method to such cases, the selected bed positions were centered on sequential sections of the lower torso with 1.5 cm of overlap between them.

The three bed positions obtained from each patient were classified based on whether the bladder is included in the cropped simulated volume or not (No-Bladder), and whether an included bladder is located outside (OoFoV-Bladder) or inside (InFoV-Bladder) the scanner's axial FoV. Details of the patients and the simulated phantoms can be found in the appendices.

For the set of analytical phantoms, a total of 9 were modeled. These included models of a NEMA-Scatter, a NEMA-Uniformity, and a NEMA-IQ phantom used in various quality assessment protocols. The rest of them were modeled for this study. In general, they consisted of ellipsoidal water phantoms that contained simple structures of different materials, such as air and bone. As

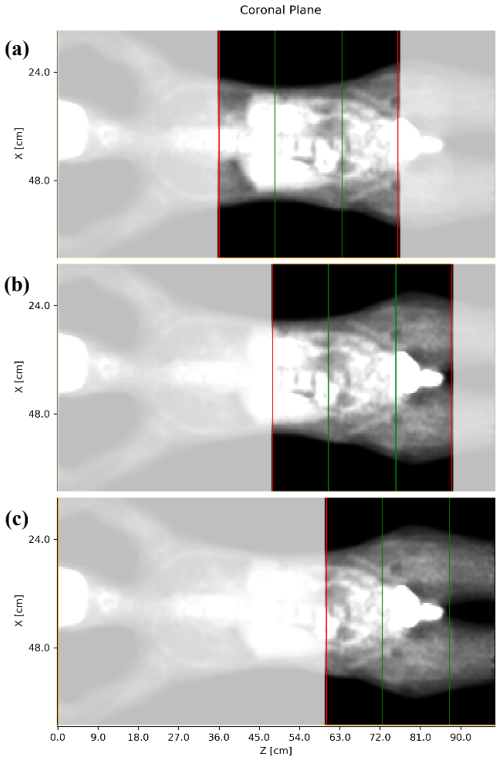


Figure 1. Example of the three bed positions cropped from the PET images of a single patient: No-, OoFoV- and InFoV- Bladder in (a), (b) and (c) respectively. The red lines define the total simulated volume, and the green lines define the part that is located inside the axial FoV of the scanner.

sources, geometric shapes of various activities were modeled inside these phantoms. Schematics and details for these phantoms can be found in Appendix I.

2.2 Monte Carlo Simulations

The phantoms were simulated in GATE [16], an emission tomography package for Geant4, a Monte Carlo physics simulation toolkit. The scanner used for the simulations was modeled based on PETMR-U, the PET/MRI scanner developed at UMC Utrecht, which is equipped with silicon photo-multiplier (SiPM) detectors. The simulation parameters and the digitizer settings of the scanner, such as the energy resolution of the detectors or the settings of the coincidence unit, were based on values found in the literature regarding digital photon counting simulations [17]. The diameter and the axial FoV of the scanner are 75 cm and 14 cm, respectively. That means that only 35% of simulated phantom volumes can be considered to be within the axial FoV (InFoV). The simulated acquisition times were set to 300s for the analytical phantoms and, in line with the clinical acquisition times per bed position, to 150s for the voxelized ones.

2.3 Data Format and Pre-Processing

From each MC-simulated phantom, full 3D PET histograms were acquired of prompts, random, scatter, and true coincidences, as well as attenuation coefficients for 511 keV photons. These histogram data were organized in stacks of 1296 direct- and oblique-plane sinograms. These were non-interpolated, in which each voxel uniquely corresponds to a Line-of-Response (LoR) of the scanner.

To reduce the dimensionality of the data, only direct-plane sinograms were used. To minimize the loss of information by discarding oblique sinograms, and to increase the statistics of the direct sinograms, Single-Slice Rebinning was used [18] to rebin up to 8 oblique sinograms to every direct one. Ultimately for each simulation, the resulting data were organized in 3D arrays formed by stacks of 64 direct-plane sinograms, with dimensions of 216 angles by 432 interleaved projection bins.

In line with conventional scatter correction methods, random coincidences were subtracted from the prompts before any further processing. The resulting random-corrected prompts (prompts(RC)), along with the attenuation (AC were further processed to be used as inputs of the NN. MC-generated scatter coincidences are processed similarly in the same 3D format. These will be used as training targets during the NN training. During the training and after feeding the inputs in the NN, the objective is to match its generated output to the MC Scatter, in what is described as "supervised learning". In these types of ML, the terms outputs and targets can be used interchangeably.

As the MC-generated data is by nature sparse and noisy, the noise level may have a large influence on the performance of the network. To investigate the input of noise practically, three combinations of data preprocessing of the inputs-label were investigated. Their differences lie in whether the downscaling and the blurring were applied to both prompts(RC) and scatter, to just the scatter, or to none of them, creating input-target pairs that can be described as

Pre-Processing Strategy	Data type	Downscaling Kernel	Gauss Sigma
RR	Prompts(RC)	-	-
	AC	-	[0.75x1.5x2]
	Scatter	-	-
RB	Prompts(RC)	-	-
	AC	-	[0.75x1.5x2]
	Scatter	[1x4x6]	[3 x 6 x 8]
BB	Prompts(RC)	[1x4x6]	[1.5 x 3 x 4]
	AC	[1x4x6]	[0.75x1.5x2]
	Scatter	[1x4x6]	[3 x 6 x 8]

Table 1. The downscaling and Gaussian blurring kernels are applied to different data in each pre-processing strategy.

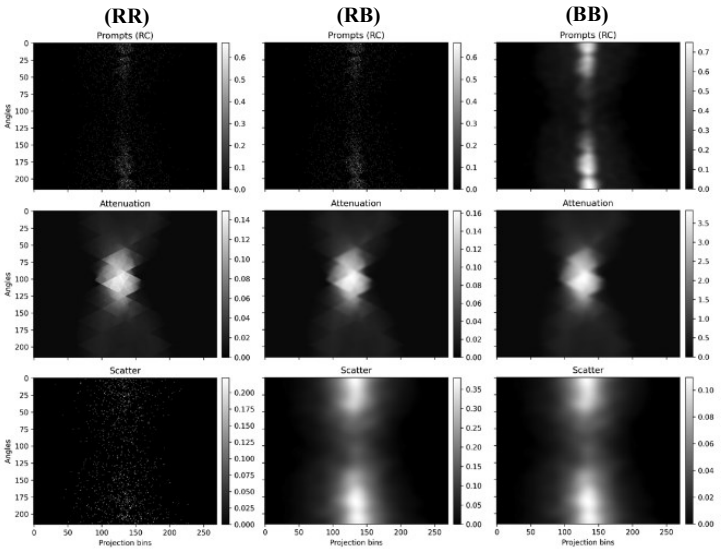


Figure 3. Sinogram examples of the inputs (prompts(RC) and attenuation factors) and their respective targets (scatter) in the three pre-processing strategies RR, RB and BB.

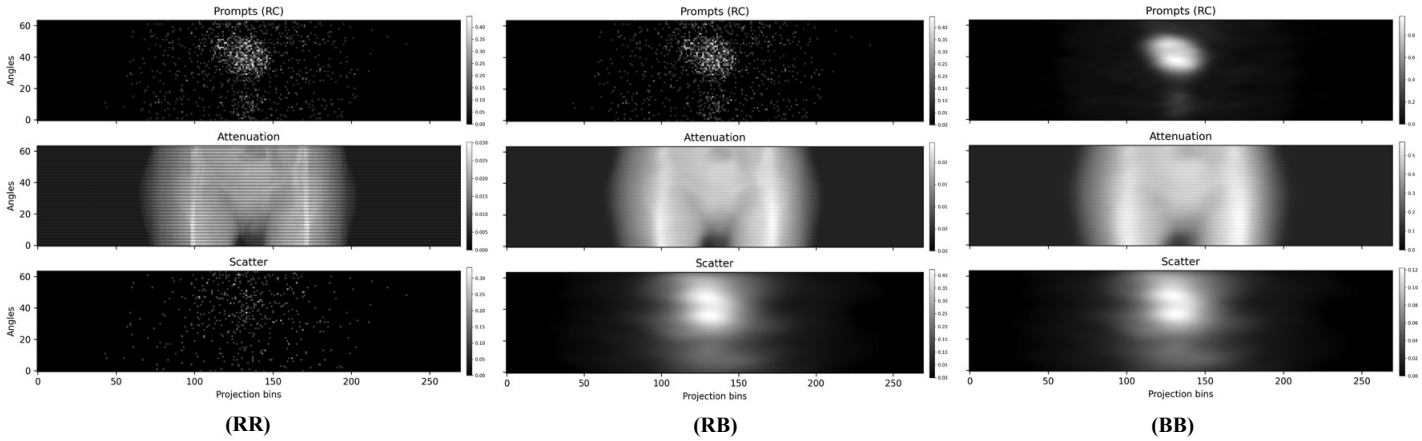


Figure 2. Projection examples of the inputs (prompts(RC) and attenuation coefficients) and their respective targets (scatter) in the three pre-processing strategies RR, RB and BB.

blurred-blurred (BB), raw-blurred (RB), and raw-raw (RR) respectively. The applied processing consisted of downscaling in the form of sum-pooling with a [1x4x6] kernel while maintaining the original dimensions of the arrays. This was followed by a gaussian blurring with different standard deviations. Specific processing details per data type for each combination are summarized in Table 1.

In all cases, prompts(RC) and scatter were normalized by the scan time, converting coincidence event counts into counting rates. Additional normalization was applied, by using the maximum value present in the 3D array of Prompts(RC). AC was normalized by a constant.

The aforementioned 3D format of the data allows for two distinct 2D visualizations of the same information. The common sinogram view (projection bins – angles), and the projection view (projection bins – axial projection bins). This enabled the investigation of trainings in both sinograms and projections independently.

2.4 Training Datasets

Using combinations of the two types of simulated phantoms, two types of data representations, and three strategies of input-label blurring strategies, 12 datasets were generated. Another set of 6 datasets was generated by merging the analytical and voxelized phantoms. A method of data enhancement specific to the sinogram datasets was applied by exploiting the 180° symmetry present in sinograms. As such, additional copies of the existing sinogram data were generated by introducing a random angular shift, which doubled the amount of training data for sinogram trainings. Ultimately, 18 distinct datasets were created, whose characteristics can be found in Table 2.

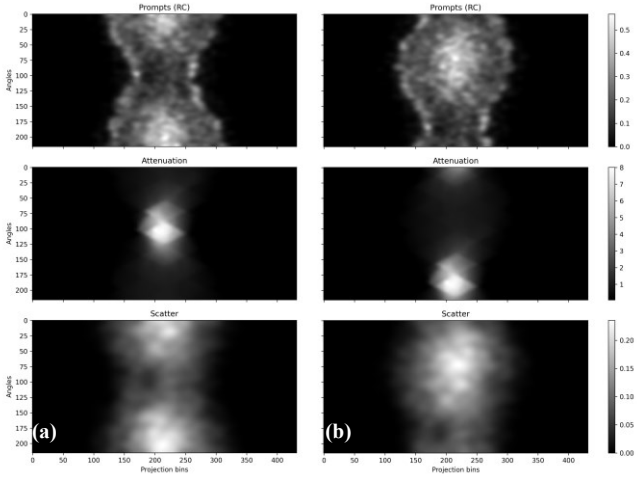


Figure 4. Introducing a random angular shift on sinograms, a 2-fold increase of training data was achieved in the training datasets of sinograms. Original and shifted sinograms in columns (a) and (b) respectively.

Training Phantoms	Projection Type	Preprocessing Strategy (Input-Label)	Dataset no.	Slice Dimensions	Slices per Simulation (+ sin. enh.)	Training Set		Test Set	
						Simulations	Total Slices	Simulations	Total Slices
Analytical	Sinogram	R-R	1	[216 x 432]	64 (+ 64)	7	896(enh.)	5	320
		R-B	2	[216 x 432]	64 (+ 64)	7	896(enh.)	5	320
		B-B	3	[216 x 432]	64 (+ 64)	7	896(enh.)	5	320
	Projections	R-R	4	[64 x 432]	216	7	1512	5	1080
		R-B	5	[64 x 432]	216	7	1512	5	1080
		B-B	6	[64 x 432]	216	7	1512	5	1080
Voxelized	Sinograms	R-R	7	[216 x 432]	64 (+ 64)	21	2688(enh.)	5	320
		R-B	8	[216 x 432]	64 (+ 64)	21	2688(enh.)	5	320
		B-B	9	[216 x 432]	64 (+ 64)	21	2688(enh.)	5	320
	Projections	R-R	10	[64 x 432]	216	21	4536	5	1080
		R-B	11	[64 x 432]	216	21	4536	5	1080
		B-B	12	[64 x 432]	216	21	4536	5	1080
Analytical + Voxelized	Sinograms	R-R	13	[216 x 432]	64 (+ 64)	28	3584(enh.)	5	320
		R-B	14	[216 x 432]	64 (+ 64)	28	3584(enh.)	5	320
		B-B	15	[216 x 432]	64 (+ 64)	28	3584(enh.)	5	320
	Projections	R-R	16	[64 x 432]	216	28	6048	5	1080
		R-B	17	[64 x 432]	216	28	6048	5	1080
		B-B	18	[64 x 432]	216	28	6048	5	1080

Table 2. List of created datasets and their characteristics.

2.5 Neural Network Trainings

The prompts(RC), along with the attenuation coefficients (AC) were used as inputs of a CNN for supervised learning, with scatter being the training target. The architecture of the CNN was based on a 2D U-NET [11] and was implemented in PyTorch. The encoder of the model consisted of a sequence of 3x3 kernel convolutional layers, element-wise rectified linear units (ReLu), batch normalization layers (BN), and 2x2 max-pooling operations reaching a latent space of dimensions 512x54x27 and 512x54x8, for sinograms and projections respectively. The decoder was composed of 2x2 kernel transposed convolutional layers with the usual concatenation of skipped connection volumes. The output layer consists of a convolutional layer coupled with a ReLu. The total number of trainable parameters in the model was 7.7 million.

The training process was performed using the Adam optimizer with learning rates Lr=1e-04 and weight decay 10^{-7}, as well as with Mean Square Error (MSE) as the loss function. These were chosen after performing a minimal grid search over other possible values. Early stopping was used with patience of 50 epochs, to terminate the training before it reaches overfitting, by monitoring the training and the validation losses.

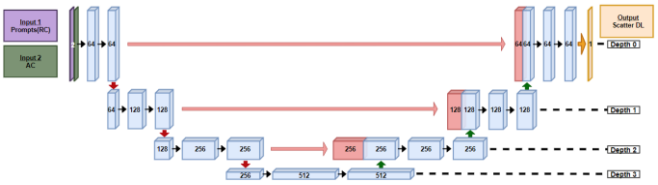
2.6 Evaluation

From all datasets, the data originating from the simulations of the analytical phantoms NEMA-IQ and NEMA-scatter, and the three simulated bed positions (No-, OoFoV- and InFoV-Bladder) of a single patient, were excluded from the trainings. Feeding their prompts(RC) and AC in the trained networks, inferences of the scatter contribution were made in the 2D form of sinograms or projections. These representations were concatenated back to the complete 3D format and then compared to the respective MC scatter contributions.

The evaluations were performed using the Normalized Root Mean Squared Error (NRMSE) as a metric. For S_{MC} and S_{DL} the MC- and the DL- generated scatter, NRMSE over all voxels n of the 3D array was calculated based on the formula:

$$NRMSE = 100\% \frac{\sqrt{MSE}}{S_{MC}^{max} - S_{MC}^{min}}, \quad with\ MSE = \frac{\sum_i \left(S_{MC}^i - S_{DL}^i\right)^2}{n}$$

In the cases of the RR strategies, both the target and the inferred scatter were downscaled and blurred similarly to the RB and BB ones, before the evaluations.



Operations	Tensor Size [Width x Height]		
		Sinograms	Projections
2D Convolution (3x3) + BN + ReLu			
2D Convolution (3x3) + ReLu	Depth 0	432x216	432x64
Concatenation	Depth 1	216x108	216x32
Max-Pooling (2x2)	Depth 2	108x54	108x16
2D Transposed Convolution (3x3)	Depth 3	54x27	54x8

Figure 5. Architecture of the CNN used

2.7 Analysis

The optimal training dataset will be chosen based on the mean performance of the network over the five cases of the test dataset. Grouping the training datasets based on their common characteristics and computing their average performance was also performed to allow for investigations of the overall impact that each characteristic had on the performance of the network.

In some cases, to visually inspect the MC -DL scatter mismatch in the entire 3D array format, we constructed 3 images. Each one of them was estimated by the maximum values of the Normalized Mean Absolute Error (NMAE) along a certain dimension, as estimated by:

$$NMAE^i = 100\% \frac{MAE^i}{S_{MC}^{max} - S_{MC}^{min}}, \quad with\ MAE^i = \frac{\left|S_{MC}^i - S_{DL}^i\right|}{n}$$

Essentially, one of the images contains the maximum NMAE over all sinogram views of the data, and the other the projection views. Such visualization can assist in locating cases in which a trained network underperforms.

3. Results

3.1 Simulations

The statistics of the data acquired in the Monte Carlo simulations varied per phantom. Specifically for the voxelized phantoms, on average 1.1 million prompts were acquired in 150s, composed of 0.5 million true, 0.4 million scattered, and 0.2 million random coincidences. Higher statistics were possible with the analytical phantoms in 300s, with 11.1, 7.1, 1.9, and 2.1 million coincidences for prompts, trues, scatter, and randoms respectively. These numbers regard the full 3D PET acquisition, which contained 121 million LoRs in a set

Phantom Type	Class	# of Sims	InFoV/ Simulated Activity[%]		Prompts		Prompts(RC)		Trues/ Prompts(RC) [%]		Scatter/ Prompts(RC) [%]		Randoms/ Prompts(RC) [%]	
			Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
Analytical	-	9	48.6	24.3	11126783	9395664	9003065	8439554	58.5	20.3	24.8	10.1	22.7	8.8
Voxelized	No Bladder	8	37.67	6.18	1043011	121736	843577	72484	55.94	6.66	41.82	3.16	23.51	6.16
Voxelized	OoFoV Bladder	8	25.93	3.75	912746	257513	683230	215619	52.96	2.05	47.04	2.05	33.46	9.76
Voxelized	InFoV Bladder	8	63.50	5.04	1336589	363021	1173061	287713	60.06	3.72	39.94	3.72	13.92	6.39

Table 3. Mean and standard deviations of activity and coincidence ratios, as obtained by the MC simulations.

of 1296 sinograms. From these, only 548 were used, after SSRB. On average, each simulation required a computation time that would be equivalent to 1600 hours in a single 4.5 GHz CPU core.

A summary of important ratios describing the statistics of the simulations can be found in Table 3, while the full results can be seen in Appendix III. A noteworthy observation could be considered the correlation of the InFoV/Simulated activity ratio with the number of trues, scatter, and randoms. As expected, the OoFoV-Bladder phantoms have the least amount of trues, and the maximum number of scatter and randoms coincidences.

3.2 Pre-Processing

The MC-generated sinograms were processed and visualized according to their type (prompts(RC), AC or scatter) and the preprocessing strategy that they belonged to, as seen in Table 1. The different preprocessing operations followed in different datasets, created different ranges of values between inputs and the target. These however were constant and consistent among the examples of the same dataset type. Examples of sinograms and projections under the different preprocessing strategies can be seen in Figures 3 and 4 respectively.

3.3 Trainings

The duration of the model trainings varied depending on the projection type, the amount of training data included in each dataset and the activation of the implemented early stopping. For datasets with the RB and BB strategies, 200-300 epochs were required for the training to reach a local optimum. RR strategies required 20-50 epochs before the validation loss started to increase. A few examples of network outputs and some evaluation metrics that were plotted during the training can be found in Figure 6 and Appendix IV.

More specifically, in Figure 6 (a) we observe how a single projection slice of RR strategy can be visualized, and the degree of mismatch between the target and the output. Similarly, in Figure 6 (b) we can observe a better agreement between the DL-generated scatter and the blurred target of a BB strategy.

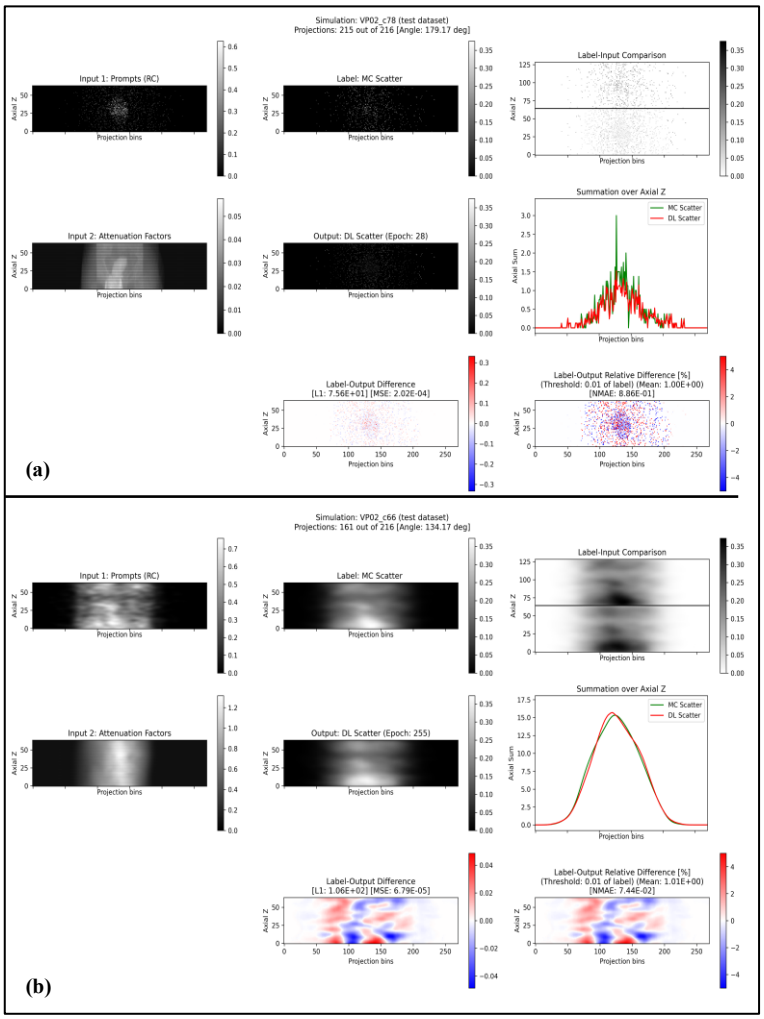


Figure 6. Two examples of the best inferred projection slices of two test cases, InFoV and OoFoV-Bladder, for (a) and (b) respectively. These examples are from training on dataset 16 and 18, again for (a) and (b) respectively. The eight graphs in each of the cases, from the left to right columns, and from the upper row to towards down, we see the two inputs (prompts(RC) and Attenuation Factors), the scatter contributions (MC- and DL- Scatter, and their difference), a side-by-side comparison of the two with inverted colormaps, a profile of the summation over the axial dimension of the MC- and DL- Scatter, and finally the NMAE of the two for which a thresholding of 1% of the MC- Scatter was used for the calculation.

3.4 Inferences and analysis

Once all networks were trained, evaluations on the five test cases were performed. As such, 2D inferences of sinograms and projections were made and concatenated to create full 3D sets of DL-generated scatter contributions. To obtain one such 3D set, 1.45 or 4.8 seconds were required, depending on whether the network was used to generate 64 sinograms or 216 projections, respectively.

The estimated NRMSE between the DL-scatter and the MC-scatter, for each dataset and per test case, can be found in Table 4. The means and standard deviations of NRMSE over all test cases for each dataset are also computed.

Training Phantoms	Projection Type	Preprocessing Strategy (Input-Label)	Dataset no.	Evaluation NRMSE [%]					Analysis NRMSE [%]		
				Analytical Phantoms		Voxelized Phantoms			Mean ± Std	Analytical Phantoms	Voxelized Phantoms
				NEMA IQ	NEMA Scatter	No Bladder	OoFoV Bladder	InFoV Bladder		Mean ± Std	Mean ± Std
Analytical	Sinogram	R-R	1	9.5	5.6	22.1	7.1	12.6	11.4 ± 6.6	7.5 ± 2.0	13.9 ± 6.2
		R-B	2	9.0	12.7	52.6	13.3	18.3	21.2 ± 17.9	10.9 ± 1.8	28.1 ± 17.5
		B-B	3	14.3	3.4	18.6	6.0	11.2	10.7 ± 6.1	8.8 ± 5.4	11.9 ± 5.2
	Projections	R-R	4	7.9	10.6	40.0	11.8	17.0	17.5 ± 13.0	9.2 ± 1.4	23.0 ± 12.3
		R-B	5	8.3	18.4	34.3	9.5	13.1	16.7 ± 10.6	13.3 ± 5.0	19.0 ± 11.0
		B-B	6	12.7	10.7	18.8	6.1	9.5	11.6 ± 4.7	11.7 ± 1.0	11.5 ± 5.4
Voxelized	Sinograms	R-R	7	17.2	10.5	6.3	4.1	2.8	8.2 ± 5.8	13.8 ± 3.3	4.4 ± 1.4
		R-B	8	33.7	5.9	7.8	7.2	5.3	12.0 ± 12.2	19.8 ± 13.9	6.8 ± 1.1
		B-B	9	8.9	9.8	6.1	5.4	3.1	6.7 ± 2.7	9.3 ± 0.4	4.9 ± 1.3
	Projections	R-R	10	22.1	11.7	4.4	4.1	2.8	9.0 ± 8.1	16.9 ± 5.2	3.8 ± 0.7
		R-B	11	27.8	12.9	6.3	4.9	5.1	11.4 ± 9.8	20.4 ± 7.5	5.4 ± 0.6
		B-B	12	13.4	9.3	5.4	3.4	3.2	6.9 ± 4.4	11.4 ± 2.1	4.0 ± 1.0
Analytical + Voxelized	Sinograms	R-R	13	9.2	11.2	8.4	5.3	2.7	7.4 ± 3.4	10.2 ± 1.0	5.5 ± 2.4
		R-B	14	8.7	6.8	7.9	7.8	4.7	7.2 ± 1.6	7.7 ± 1.0	6.8 ± 1.5
		B-B	15	11.4	9.4	7.9	5.9	4.0	7.7 ± 2.9	10.4 ± 1.0	5.9 ± 1.6
	Projections	R-R	16	5.5	7.4	5.6	3.6	2.6	4.9 ± 1.9	6.5 ± 1.0	3.9 ± 1.2
		R-B	17	8.3	13.0	6.4	5.0	3.8	7.3 ± 3.6	10.6 ± 2.4	5.0 ± 1.1
		B-B	18	13.4	6.1	5.7	3.6	2.7	6.3 ± 4.2	9.7 ± 3.7	4.0 ± 1.2

Table 4. NRMSE as computed between the DL-generated scatter estimations and the MC-scatter ground truth, for each of the five test cases.

Comparison	Categories	Included Datasets	NRMSE [%] (mean ± standard deviation)				
			Analytical Phantoms		Voxelized Phantoms		
			NEMA IQ	NEMA Scatter	No Bladder	OoFoV Bladder	InFoV Bladder
Trained Phantom Types	Analytical	1-6	10.3 ± 2.6	10.2 ± 5.3	31.1 ± 13.7	9.0 ± 3.1	13.6 ± 3.4
	Voxelized	7-12	20.5 ± 9.2	10.0 ± 2.4	6.0 ± 1.1	4.9 ± 1.4	3.7 ± 1.1
	An.+Vox.	13-18	9.4 ± 2.7	9.0 ± 2.7	7.0 ± 1.2	5.2 ± 1.6	3.4 ± 0.9
Data Representation	Sinograms	1-3,7-9,13-15	13.5 ± 8.1	8.4 ± 3.1	15.3 ± 15.1	6.9 ± 2.6	7.2 ± 5.5
	Projections	4-6,10-12,16-18	13.3 ± 7.3	11.1 ± 3.6	14.1 ± 13.9	5.8 ± 3.0	6.7 ± 5.3
Preprocessing Strategy (InputsTarget)	RR	1,4,7,10,13,16	11.9 ± 6.4	9.5 ± 2.4	14.5 ± 14.1	6.0 ± 3.1	6.8 ± 6.4
	RB	2,5,8,11,14,17	16.0 ± 11.6	11.6 ± 4.6	19.2 ± 19.7	7.9 ± 3.1	8.4 ± 6.0
	BB	3,6,9,12,15,18	12.3 ± 1.9	8.1 ± 2.8	10.4 ± 6.5	5.1 ± 1.3	5.6 ± 3.7

Table 5. Mean and standard deviations of NRMSE, as calculated for the test cases, over datasets that share the same characteristics. Three types of comparisons are made, in each of which the 18 datasets are grouped depending on the category they belong to.

To promote further investigations of the effects of individual choices in the datasets, the results of their evaluations were also grouped according to the characteristics that the datasets have in common. As such, three comparisons were made. These are targeting the phantom types included in the training, the data representation used by the network to generate the scatter estimate, and the blurring strategy. The mean and standard deviations of the NRMSE of those groups were estimated and can be found in Table 5. These NRMSE values were kept separate for each test case.

3.5 Optimal Dataset

Based on the methods that were followed in the current study, we obtained two different candidates for the best-performing datasets for using DL for scatter corrections.

The first one is based on the direct evaluation of the NRMSE over the chosen test cases, as listed in Table 4. As such, the network that was trained with both analytical and voxelized phantoms, using projections and with the RR preprocessing strategy (dataset 16), generated scatter with the best overall agreement with the MC scatter based on a mean NRMSE value (± standard deviation) of (4.9 ± 1.9)%.

By grouping the results according to the characteristics of the datasets, and by calculating their mean NRMSE over all test cases, we obtain Table 5. In this, the overall effect that the chosen type of phantoms used in the training, the 2D view of the projection data, and the blurring strategy category, have on the overall performance of the NN are summarized. From these estimations, we can observe that using both types of phantoms, the mean NRMSE in 3 out of 5 cases got decreased over the other choices. Similarly, in 4 out of 5 cases, training on and generating projections was more accurate than using sinograms, with only in the NEMA-Scatter phantom performing worse. Lastly, we observe that the BB strategy was optimal in all test cases over the RR and RB ones. Using the combinations of the optimally performing categories, dataset 18 (from Table 5) can be deducted as another optimal one. Coincidently, this dataset ranks as second best on the first ranking method of Table 4.

The common characteristics of these two candidates already indicate the most promising directions. Their only difference, the preprocessing strategy followed, was the target for the next investigation.

3.6 Visualizations of scatter

To further investigate the best-performing networks, and to identify any potential regions of the 3D scatter volumes in which our DL methods underperform, we proceeded with the visualization that was described in 2.7. As such, we selected 9 cases to visualize in Figure 7.

Noteworthy observations include the difference in the ranges of errors in each case. The pre-evaluation visualizations that compare the unblurred MC- and DL-Scatters reach values of 74% error, showcasing the inability of

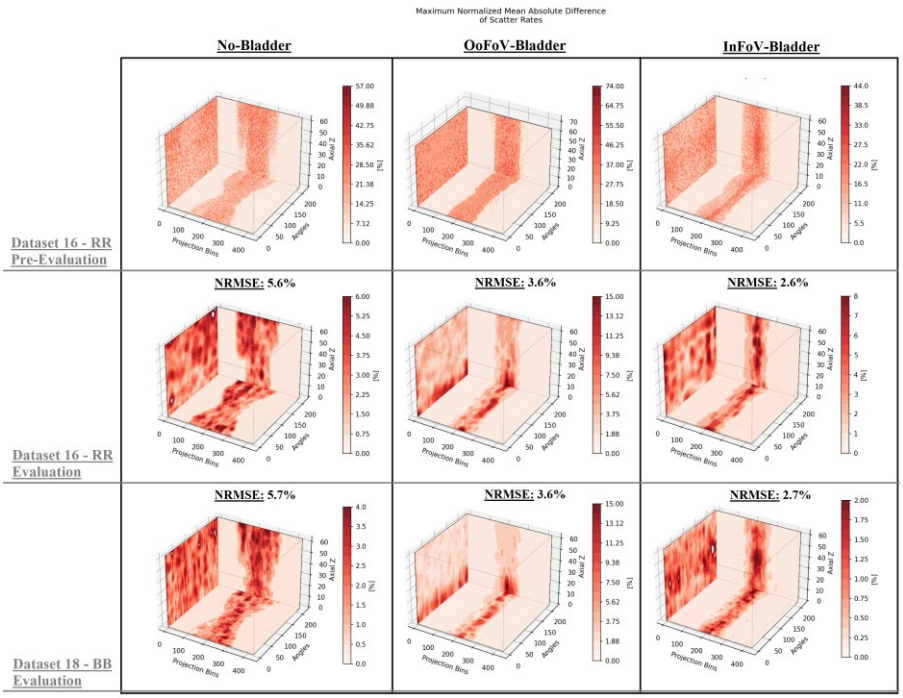


Figure 7. The maximum NMAE values of the MC-Scatter and the DL-Scatter are projected on the 3 directions that define the 3D array of our data. One of them resembles the maximum value over all sinogram slices, and another one over all projection slices. The two optimal datasets, 16 and 18, were selected. Since dataset 16 belongs to the RR strategies, the visualizations before and after the blurring that is described in 2.6 are included. For each case, the maximum errors on the three voxelized phantoms of the test dataset are visualized.

the networks to reproduce the exact sparse information of raw MC-generated data. However, with the application of blurring used in the evaluation, we observe that the mismatch is reduced significantly, to the level of becoming comparable with the mismatch present in the BB strategies.

Inspecting the locations that present the highest amount of errors, we observe that for the No-Bladder cases, these were located on large axial Z, for OoFoV-Bladder on small axial Z, and for InFoV on middle axial Z. These are in line with the positions of high activity organs in those phantoms, such as a heart for the first one, and the bladder for the other two

4. Discussion

4.1 Phantom Types

Analytical and voxelized phantoms have very distinct inputs, which also reflect on their scatter distributions. Analytical phantoms have homogeneous attenuation images and localized activity distributions. On the opposite side, voxelized phantoms based on patients show very inhomogeneous attenuation coefficients and activity distributions.

Due to these differences, by including both types of phantoms in the same training dataset, we can expect a decrease in performance in individual test phantoms of a certain type when compared to training datasets that exclusively contained that type of phantoms.

This was partially verified by our results. In Table 5, for the trained phantom types comparisons, we can observe a decrease of NRMSEs for the No-Bladder and the OoFoV-Bladder cases with the mixed phantom type datasets over the ones exclusively containing voxelized phantoms. Interestingly, this was not the case for the InFoV-Bladder phantom. We suspect that the reason behind this is the high resemblance of the InFoV bladder phantoms to multiple analytical phantoms that were designed for this study, which contained spherical regions of high activity at the center of the FoV (see Appendix I).

On the opposite side, in the analytical phantom (NEMA) test cases, the inclusion of voxelized patient phantoms in the training had a positive effect, by decreasing the mean NRMSE when compared to the datasets with only analytical phantoms. This can be interpreted as the network becoming more robust, despite the different geometries. This highlights the importance of using multiple types of phantoms.

In conclusion, including analytical phantoms in the training datasets should only be considered when there are already sufficient voxelized patient phantoms, to ensure that no significant decrease in performance on test cases of the latter type.

4.2 Data Representation

An overall improvement in the performance of the network when using projections over sinograms for the same imaging volume can be deducted from the comparisons in Table 4. Except for the NEMA-scatter, the networks trained with projections had lower mean NRMSEs over the rest four test cases.

We believe that the key to that performance difference is the ranges of pixel intensities within each sinogram or projection slice. To further elaborate, after normalizing the entire 3D array of MC scatter with the maximum value of prompts(RC) (as mentioned in 2.3), we constructed a set of 64 sinograms and 216 projections. However, despite them representing the same data overall, we observed that the ranges of values within the 64 sinogram slices vary a lot when compared to the ranges of values within 216 projections. This is because each sinogram contains very little information from the axial FoV, while the projections have more, rendering their values more consistent. Using DL with images of highly inconsistent contrast, can make the training process less stable and ultimately worsen the networks' performance.

The decreased spatial correlation between adjacent pixels in sinogram views when compared to projections, could also manifest itself through this difference. Both dimensions in a projection slice have a linear spatial correlation (projection bins – axial Z), while a sinogram slice has linear (projection bins) and angular (angles) spatial correlation between adjacent voxels. This deviation from spatial correlation uniformity might pose a challenge for a CNN to learn.

Another explanation for this performance difference can be deducted from the different number of slices per training dataset. In general, as seen in Table 2, projection datasets had 60% more slices than the equivalent projection ones, even after doubling the number of sinograms by the data-enhancing method that we described in 2.4.

4.3 Blurring Strategy

The motivation behind the investigation of different blurring strategies should be mentioned. After obtaining our data from the MC simulations and visualizing them, it became clear that applying DL by training a CNN on them would be very challenging. This notion was even though these types of networks have proven to be very capable of complex pattern recognition and robust on both regression and classification tasks. The low counts and the sparsity of information on both sinograms and projection, combined with the embedded noise, were rendering a raw-inputs to raw outputs as a task that would resemble more of a voxel-to-voxel classification. In a task as such, each voxel would have to be classified in the correct discrete value of scattered coincidences that it contains. And indeed, CNN was challenged in this task.

Evidence of that can be observed in Figure 6 and Figure 7 which contain comparisons of raw-raw MC- and DL-Scatter. Despite the network managing to reproduce the sparsity and the noise-like nature of the unblurred targets, a pixel-to-pixel match could not be achieved.

At that point, one important realization was made, specific to the task that we aimed to solve. Matching precisely the distribution of the raw scatter coincidences is of no interest to us. Both the MC-generated inputs-targets or even the real-scenario prompts(RC), will always be products of highly stochastic processes, either due to the pseudo-random number generator used in MC or due to the Poisson nature of the acquisition. Additionally, the effect of scatter contribution has a low frequency on the reconstructed image, which could imply that the scatter correction method is less sensitive to pixel-wise comparisons of noisy data. These ideas are supported by the methodology used in MCSC, which requires blurring and scaling of the scatter estimate before it gets subtracted from the acquired data.

Consequently, inspired by MCSC and our initial skepticism on the raw-raw training, it was decided to investigate the RB and the BB strategies in parallel to the RR. Additionally, to reliably compare these strategies based on a metric, such as the NRMSE, the DL-Scatter obtained from the RR strategies had to be blurred similarly to the others.

However, despite these investigations, the optimal choice of blurring remains inconclusive, with the RR and BB strategies being included in datasets 16 and 18 of the two best-performing networks. A way to interpret this ambiguity could be through the way a CNN, like our model, processes input data. The sequential convolution and max-pooling operations over the input data during training can result in a loss of high-frequency information that can resemble blurring. Specifically, in the RR strategies, a form of sampling from these blurred distributions seems to take place in the network, generating output similar to the targets as result. These resulted in noisy-like and sparse outputs that were once sampled from a pool of blurred distributions, which can be partially restored when the evaluation blurring is applied.

In future investigations, more educated choices should be made on the amount of blurring applied, since the values used in MCSC are not reported in the literature. Interestingly, in the RR strategies, only a relatively small number of epochs was required before the validation loss started to increase, indicating quick overfitting of the network's parameters to the training data. This implies that the network was not suitable for this task, and one with fewer learnable parameters could perform even better by using datasets with the RR strategy.

4.4 Simulations results

As expected, simulations generated very sparse data, in which valuable information was often indistinguishable from noise. Especially the voxelized simulations can be characterized as low-count MC simulations since the number of the acquired prompt coincidences was on average ~15% of the ones in a typical clinical scan of the same lower-torso regions. The attempts to increase the statistics by only considering direct sinograms and applying SSRB could only partially compensate for that. Ideally, higher statistics would be needed for a highly accurate scatter correction method using DL.

The visualizations of the attenuation factors exhibit some very distinct artifacts. These are caused by the combination of using non-interpolated projections and sinograms acquired in a scanner geometry with occasional gaps between the detectors. Despite these intense artifacts, several networks managed to perform decently and no artifacts were observed in the generated outputs. This could imply that a CNN might be able to learn to ignore such systematic anomalies.

As expected, scatter and random coincidences were the highest in the OoFoV-Bladder phantoms, as seen in Table 3.

4.5 In and out of FoV activities

In our investigations, no signs of overall performance decrease in cases of OoFoV-Bladder phantoms were observed. Such phantoms measured the highest number of scattered coincidences, as seen in Table 3. Additionally, since the inputs of prompts and attenuation factors do not include any direct information about the contribution of OoFoV scatter, we would expect that the worst-performing cases would be the ones with an OoFoV-Bladder. Despite the analysis and the maximum NMAE values of Figure 7 indicating that the OoFoV-Bladder cases include some of the worst underperformance of the networks trained with the datasets 16 and 18, the overall NRMSE of those cases was consistently lower than the No-Bladder phantom ones.

The implications of these observations could be of great importance. We could hypothesize that despite the lack of direct OoFoV information to the network from these regions, such as the activity or the attenuation mediums, the network could be trained to able to predict a pattern of OoFoV scatter only through the input of prompts(RC) and the target image of scatter, the latter of which is a subset of the first. Since on a high level and for low-resolution information such as the scatter contribution, human anatomies can be considered very similar, the OoFoV scatter could potentially be learned by the network.

Since conventional SSS methods only partially account for OoFoV scatter through scaling, such development could be a key advantage of a DL method over them. It could render a DL method for scatter as either a complementary tool towards

a more accurate scaling of existing scatter correction methods, or as an entire replacement of them.

However, there is hesitance on drawing solid conclusions on this. More investigations are needed that target more methodically the OoFoV-activity contribution aspect of scatter corrections. Our method and our conclusions were based on a minimum amount of test cases. The next logical improvement in the methodology would be the cross-validation of these results by using different subsets of simulated test cases each time.

4.6 Over the evaluation method

For results of higher clinical relevance, training and evaluation only on voxelized patient phantoms would suffice. A study as such, with a rich dataset from multiple bed positions, would be very promising, as shown in this current work and in [15]. The methods of this work deviated deliberately from such a patient-oriented setup, by including analytical phantoms.

This choice was primarily driven by the ambition of achieving a physics-informed NN, which would capable to approximate a physical relationship between the inputs and the target. As such, we hypothesize that a well-trained network could be robust enough to generate scatter contributions from even very distinct and unseen, by the network, phantom geometries. Additional motivation for including analytical phantoms was to work towards a sufficiently robust scatter correction method that could be applied to protocols of PET acquisition and image quality assessment, such as the NEMA ones.

However, the inclusion of analytical phantoms in the test cases influences the dataset evaluation process that was applied in this work, in an unintuitive way. This influence can be observed when attempting to perform cross-comparisons of the categories defined in Table 5. For example, the optimal preprocessing strategies were BB, BB, and RR in 70%, 50% and 60% of the cases using datasets trained with analytical, voxelized and both types of phantoms, respectively.

For these reasons, depending on the resources available to future studies, and also depending on whether their goal is generalisability over specificity, analytical phantoms should be included in the test set only if the dataset is large and diverse enough already.

4.7 Future work

Several improvements could be implemented at different points along the sequence of methods that was followed.

In line with most DL methods, the most impactful way to improve the learning capabilities of the network is by improving the training datasets. In our case, in which we generate our training data by Monte Carlo simulations, this could be done in multiple ways. The most obvious is to increase the simulated patients and the bed positions. Increasing the acquisition time to achieve higher numbers of coincidences would positively impact the performance since more information would be present in each image. At least 7-10 times more counts per bed position to match the clinical standards would be ideal. Similarly, including the entire voxelized phantom in the simulation instead of cropping a section of it would greatly improve the validity of the method.

On the NN aspect, we see multiple directions toward improvement. Hyperparameter tunings of the network and the optimizer were not exhausted in this current work. It could however have a significant impact on the performance of the network. In case a strategy resembling the RR one is followed in the future, a network with fewer learnable parameters would be the best choice since severe overfitting was observed with the current network. In case a large dataset is available, the use of a 3D NN architecture could achieve higher accuracy than its 2D counterpart. Training on full 3D sinogram data to minimize the loss of information by using only direct-plane ones would be challenging due to the higher sparsity and noise, but it could become a topic of investigation.

Of crucial importance before these improvements would be the validation of the DL-generated scatter. The way that a potential mismatch of scatter estimation in projection space can manifest itself in the post-reconstruction image space can not be predicted. As such, we believe that the evaluation of a DL scatter method should be ultimately performed in the final reconstructed images. In this context, and following the conclusions drawn in this study about the phantom types (4.1), the validation of such a method should take place on a commercial scanner using two types of phantoms. In line with the standard PET quality assessment protocols, NEMA phantoms should be one of them. To ensure performance in clinical cases, anthropomorphic phantoms should be the other one. The validation should be based on the quantification accuracy (SUV) in the final, scatter-corrected, reconstructed images. Additional validation by comparisons with the standard SSS could also be of great interest.

Lastly, scatter corrections in combination with other corrections such as attenuation and randoms using DL could also be of interest. One could hypothesize that the correlations between the prompts, attenuation coefficients, randoms and scatter, could be approximated accurately through a DL method.

5. Conclusions

The ability to use a 2D U-Net to generate Monte Carlo- estimated scatter contributions was showcased, rendering the use of a DL method to generate MCSC-grade within clinical timeframes feasible. The use of projections provided more accurate scatter estimates than with sinograms. Including training data from analytical phantoms along with patient-based ones, should only be considered if resources allow it and only if the objective is a more generalized scatter correction method. Both processed and unprocessed MC-generated inputs-targets can be considered for training purposes, provided that blurring of the MC-generated scatter is eventually required for the scatter correction to be applied to the acquired data. Further work and validation of the method on the reconstructed images should be performed.

References

[1] S.R. Cherry, J.A. Sorenson, M.R. Phelps – Physics in nuclear medicine, *Saunders, 4th ed. pp. 323, 332.*

[2] Y. Hirano, K. Koshino, H. Iida (2017) – Influences of 3D PET scanner components on increased scatter evaluated by a Monte Carlo simulation. *Phys. Med. Biol. 62 4017* – doi: 10.1088/1361-6560/aa6644

[3] J. Ollinger (1996) – Model-based scatter correction for fully 3D PET. *Phys. Med. Biol. 41. 153-76.* – doi: 10.1088/0031-9155/41/1/012

[4] C. C. Watson (2000) – New, faster, image-based scatter correction for 3D PET. *IEEE Trans. Nucl. Sci. , vol. 47, no. 4, pp. 1587–1594.* – doi: 10.1109/23.873020

[5] C.H. Holdsworth, C.S. Levin, M. Janecek, M. Dahlbom, E.J. Hoffman (2002) – Performance analysis of an improved 3-D PET Monte Carlo simulation and scatter correction. *IEEE Trans. Nucl. Sci. , vol. 49, no. 4, pp. 83–89* – doi: 10.1109/TNS.2002.998686

[6] T. Heußer, P. Mann, C.M. Rank, M. Schäfer, A. Dimitrakopoulou-Strauss, H.P. Schlemmer, B.A. Hadaschik, K. Kopka, P. Bachert, M. Kachelrieß, M.T. Freitag (2017) – Investigation of the halo-artifact in 68Ga-PSMA-11-PET/MRI. *PLoS One, 12(8):e0183329* – doi: 10.1371/journal.pone.0183329

[7] [Magota 2020] K. Magota, N. Numata, D. Shinyama et al. (2020) – Halo artifacts of indwelling urinary catheter by inaccurate scatter correction in 18F-FDG PET/CT imaging: incidence, mechanism, and solutions. *EJNMMI Phys. 2020;7(1):66* – doi: 10.1186/s40658-020-00333-8

[8] J. Ye, X. Song and Z. Hu (2014) – Scatter correction with combined single-scatter simulation and Monte Carlo simulation for 3D PET, *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2014, pp. 1-3* – doi: 10.1109/NSSMIC.2014.7431033

[9] C.C. Watson, J. Hu, C. Zhou (2018) – Extension of the SSS PET scatter correction algorithm to include double scatter. *IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2014, pp. 1-4* – doi: 10.1109/NSSMIC.2018.8824475

[10] B. Ma, M. Gaens, L. Caldeira, J. Bert, P. Lohmann, L. Tellmann, C. Lerche, J. Scheins, E. Rota Kops, H. Xu, M. Lenz, U. Pietrzyk, J.J.Shah (2019) – Scatter Correction Based on GPU-Accelerated Full Monte Carlo Simulation for Brain PET/MRI. *IEEE Trans. Med. Imaging. 2020 Jan;39(1):140-151* – doi: 10.1109/TMI.2019.2921872

[11] O. Ronneberger, P. Fischer, and T. Brox (2015) – U-Net: Convolutional networks for biomedical image segmentation, *Proc. Med. Image Comput. Comput. Assist. Interv. , vol. 9351, Nov. 2015, pp. 234–241* – arXiv: 1505.04597

[12] I. Häggström, C.R. Schmidtlein, G. Campanella, T.J. Fuchs (2019) – DeepPET: A deep encoder-decoder network for directly solving the PET image reconstruction inverse problem. *Med Image Anal. 2019;54:253-262.* – doi: 10.1016/j.media.2019.03.013

[13] H. Arabi, K. Bortolin, N. Ginovart, V. Garibotto, H. Zaidi (2020) – Deep learning-guided joint attenuation and scatter correction in multitracer neuroimaging studies. *Hum Brain Mapp. 2020 Sep;41(13):3667-3679* – doi: 10.1002/hbm.25039

[14] S. Mostafapour, F. Gholamiankhah, H. Dadgar, H. Arabi, H. Zaidi (2021) – Feasibility of Deep Learning-Guided Attenuation and Scatter Correction of Whole-Body 68Ga-PSMA PET Studies in the Image Domain. *Clin Nucl Med. 2021 Aug 1;46(8):609-615.* – doi: 10.1097/RLU.0000000000003585

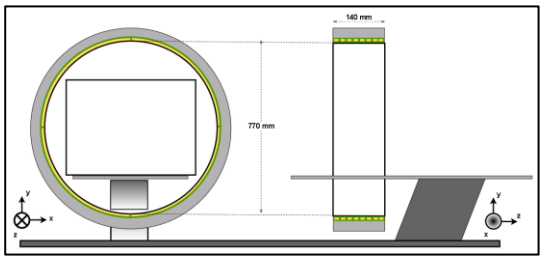
[15] Y. Berker, J. Maier, M. Kachelries (2018) – Deep Scatter Estimation in PET: Fast Scatter Correction Using a Convolutional Neural Network. *1-5.* – doi: 10.1109/NSSMIC.2018.8824594

[16] D. Sarrut, M. Bała, M. Bardiès, et al (2021) – Advanced Monte Carlo simulations of emission tomography imaging systems with GATE. *Phys Med Biol. 2021; vol. 66(10)* – doi: 10.1088/1361-6560/abf276

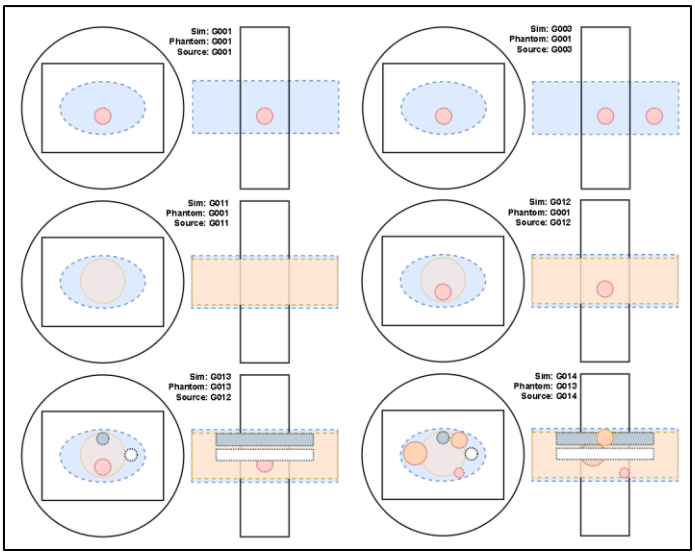
[17] J. Salvadori, J. Labour, F. Odille, et al. (2020) – Monte Carlo simulations of digital photon counting PET, *EJNMI Phys. 2020;7(1):23.* – doi: 10.1186/s40658-020-00288-w

[18] R.M. Lewitt, G. Muehllehner, S.J. Karp (1991) – The 3D image reconstruction for PET by multi-slice rebinning and axial filtering, *IEEE Nuclear Science Symposium, Santa Fe, NM, 5-9* – bibcode: 1991nusc.sympS...5L

Appendix I: Analytical Phantoms



Materials	Activities (MBq)
Air	Low (10)
Water	Medium (15)
SpineBone	High (30)



Appendix II: Technical Implementation

The GATE simulations were run on a workstation equipped with an AMD Ryzen Threadripper 3970X Processor (2x32 cores, 128GB memory). The network models were implemented in Python (v3.8.3) using PyTorch (v1.8.1+cu102) and an NVIDIA Quadro P6000 (3840 CUDA cores, 24GB memory).

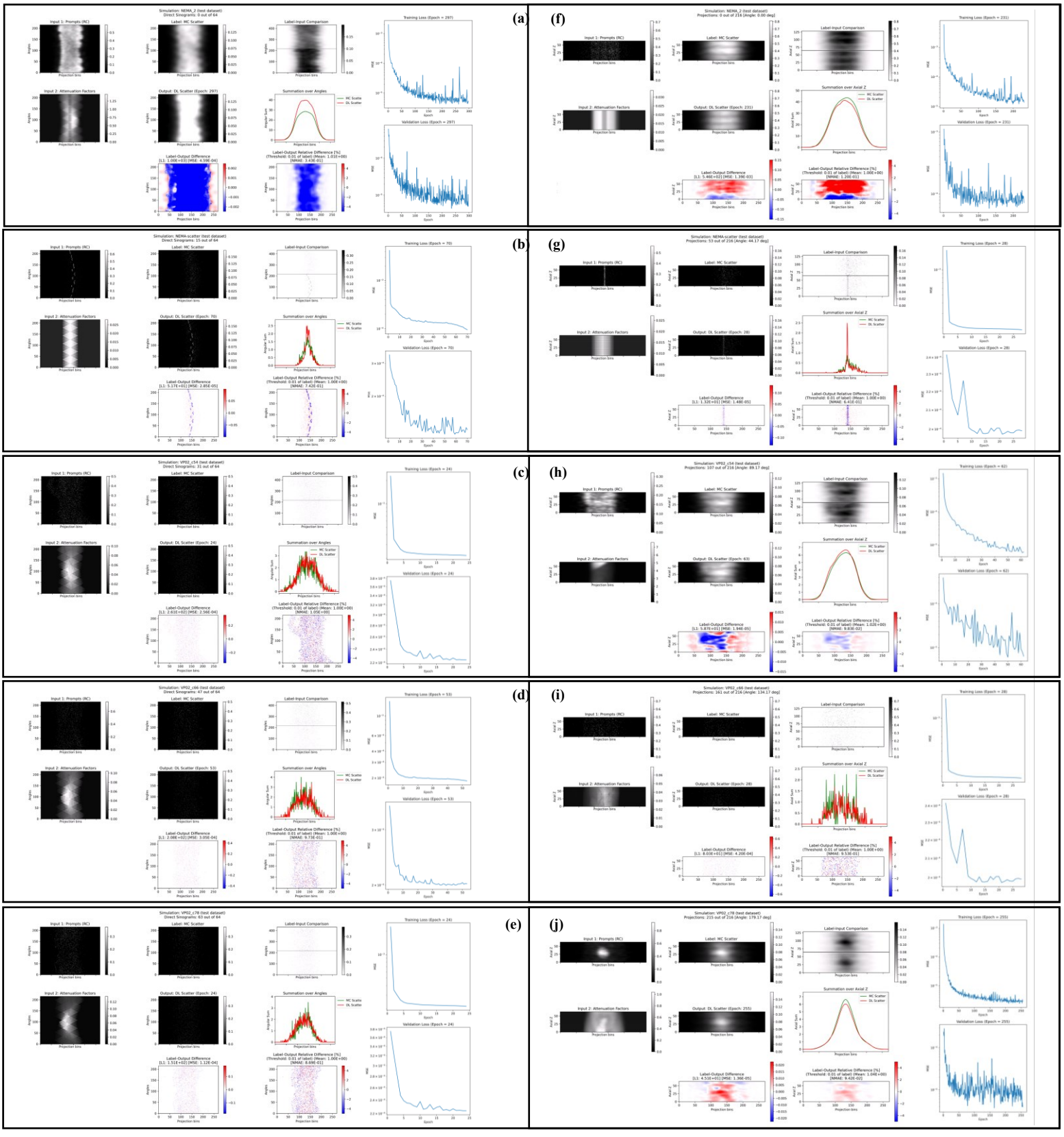
Appendix III: MC Simulations Statistics

Analytical Phantom & Simulation Information								Acquired Coincidences						Coincidence Ratios [%]					
Name	Sim. Time [s]	Sim. Activity [MBq]	InFoV Act. [MBq]	OoFoV Act. [MBq]	InFoV/ Sim.Act. [%]	InFoV/ OoFoV Act. [%]		Prompts	Trues	Scatters	Randoms	Prompts (RC)		Trues / Prompts	Scatter / Prompts	Randoms / Prompts	Scatter / Trues	Ra/Tr %	Sc/Ra %
G001	240	50	50	0,0	100,0			6691290	3868641	1944661	877988	5813302		57,8	29,1	13,1	50,3	22,7	221,5
G003	180	100	50	50,0	50,0			8191456	3855025	1942806	2393625	5797831		47,1	23,7	29,2	50,4	62,1	81,2
G011	180	30,5	6,7	23,8	22,0	28,2		2329063	939212	892517	497334	1831729		95,0	38,3	21,4	95,0	53,0	179,5
G012	240	61	37,2	23,8	61,0	156,3		27486141	22140856	2086161	3259124	24227017		80,6	7,6	11,9	9,4	14,7	64,0
G013	240	61	37,2	23,8	61,0	156,3		27489636	22144510	2085195	3259931	24229705		80,6	7,6	11,9	9,4	14,7	64,0
G014	240	67	18,5	48,5	27,6	38,1		14817975	5071450	4704422	5042103	9775872		34,2	31,7	34,0	92,8	99,4	93,3
NEMA-Uniformity	300	50	17	33,0	34,0	51,5		4147973	1685086	1088842	1374045	2773928		40,6	26,2	33,1	64,6	81,5	79,2
NEMA-IQ	300	23	14,3	8,7	62,2	164,4		5743274	2419512	1518117	1805645	3937629		42,1	26,4	31,4	62,7	74,6	84,1
NEMA-Scatter	300	31	6,2	24,8	20,0	25,0		3244237	1578327	1062247	603663	2640574		48,7	32,7	18,6	67,3	38,2	176,0

Voxelized Phantom & Simulation Information											Acquired Coincidences					Coincidence Ratios				
					Activity [%]															
Patient	Position of Center [cm]	Class	Sim Time	System	Simulated	In FoV	Out FoV	In/Tot %	In/Out %	Prompts	Trues	Scatters	Randoms	Prompts (RC)	Tr/Pr %	Sc/Pr %	Ra/Pr %	Sc/Tr %	Ra/Tr %	Sc/Ra %
1	41	No Bladder	150s	Snellius	33,47	15,22	18,25	45,5	83,4	1178158	536212	421520	220426	957732	45,5	35,8	18,7	78,6	41,1	191,2
	53	OoFoV Bladder	150s	Snellius	47,35	12,54	34,81	26,5	36,0	1070817	405630	347349	317838	752979	37,9	32,4	29,7	85,6	78,4	109,3
	65	InFoV Bladder	150s	Local	42,48	23,59	18,89	55,5	124,9	1349813	551174	498296	300343	1049470	40,8	36,9	22,3	90,4	54,5	165,9
2	54	No Bladder	150s	Local	38,89	13,85	25,04	35,6	55,3	1446047	452979	445267	351922	1094125	31,3	30,8	24,3	98,3	77,7	126,5
	66	OoFoV Bladder	150s	Snellius	43,16	8,91	34,25	20,6	26,0	1058284	370538	337218	350528	707756	35,0	31,9	33,1	91,0	94,6	96,2
	78	InFoV Bladder	150s	Local	36,04	22,39	13,65	62,1	164,0	1891051	1003891	629742	257418	1633633	53,1	33,3	13,6	62,7	25,6	244,6
3	36	No Bladder	150s	Local	28,65	13,67	14,98	47,7	91,3	1461340	777142	470507	213691	1247649	53,2	32,2	14,6	60,5	27,5	220,2
	48	OoFoV Bladder	150s	Snellius	29,96	7,84	22,11	26,2	35,5	968214	430657	326882	210675	757539	44,5	33,8	21,8	75,9	48,9	155,2
	60	InFoV Bladder	150s	Local	19,54	11,09	8,45	56,8	131,2	1135116	666738	373151	95227	1039889	58,7	32,9	8,4	56,0	14,3	391,9
4	56	No Bladder	150s	Snellius	18,2	6,9	11,3	37,9	61,1	785698	432311	264908	88479	697219	55,0	33,7	11,3	61,3	20,5	299,4
	68	OoFoV Bladder	150s	Snellius	25,67	6,41	19,26	25,0	33,3	790611	353398	295709	141504	649107	44,7	37,4	17,9	83,7	40,0	209,0
	80	InFoV Bladder	150s	Local	21,69	14,71	6,98	67,8	210,7	1308118	771961	433793	102364	1205754	59,0	33,2	7,8	56,2	13,3	423,8
5	56	No Bladder	150s	Local	36,02	12,94	21,08	35,9	61,4	1345077	631814	423978	289285	1055792	47,0	31,5	21,5	67,1	45,8	146,6
	68	OoFoV Bladder	150s	Snellius	34,01	10,3	23,71	30,3	43,4	1003062	406908	361629	234525	768537	40,6	36,1	23,4	88,9	57,6	154,2
	80	InFoV Bladder	150s	Snellius	24,63	15,67	8,96	63,6	174,9	1102804	570895	405402	126507	976297	51,8	36,8	11,5	71,0	22,2	320,5
6	48	No Bladder	150s	Local	31,07	9,76	21,31	31,4	45,8	497399	225468	172058	99873	397526	45,3	34,6	20,1	76,3	44,3	172,3
	60	OoFoV Bladder	150s	Snellius	31,96	10,17	21,79	31,8	46,7	858301	344576	330497	183228	675073	40,1	38,5	21,3	95,9	53,2	180,4
	72	InFoV Bladder	150s	Snellius	24,7	16,55	8,14	67,0	203,3	1103471	578023	410296	115152	988319	52,4	37,2	10,4	71,0	19,9	356,3
7	48	No Bladder	150s	Snellius	34,65	12,94	20,71	37,3	62,5	934986	392603	341386	200997	733989	42,0	36,5	21,5	87,0	51,2	169,8
	60	OoFoV Bladder	150s	Snellius	45,89	10,13	35,76	22,1	28,3	831386	301712	284251	245423	585963	36,3	34,2	29,5	94,2	81,3	115,8
	72	InFoV Bladder	150s	Snellius	36,17	24,9	11,27	68,8	220,9	1580280	844545	546566	189169	1391111	53,4	34,6	12,0	64,7	22,4	288,9
8	58	No Bladder	150s	Snellius	26,29	7,88	18,41	30,0	42,8	695383	315131	249455	130797	564586	45,3	35,9	18,8	79,2	41,5	190,7
	70	OoFoV Bladder	150s	Snellius	29,34	7,35	21,99	25,1	33,4	721290	288359	280530	152401	568889	40,0	38,9	21,1	97,3	52,9	184,1
	82	InFoV Bladder	150s	Snellius	24,99	16,56	8,43	66,3	196,4	1222058	667450	432568	122040	1100018	54,6	35,4	10,0	64,8	18,3	354,4

Appendix IV: Training Results - Best Test Cases in 2D

Examples of inferred sinograms and projections of the five cases of the test dataset: the NEMA-IQ (a,f), NEMA-Scatter (b,g), and Voxelized No-Bladder (c,h), OoFoV-Bladder (d,i), InFoV-Bladder (e,j), with sinograms in the first column (a,b,c,d,e) and projections on the second one (f,g,h,i,j). These examples are of the same slices, however from different networks. The second best performing network was chosen for each case, as ranked by the NRMSE of Table 4. In each of the cases, from the left column towards the right, we see the two inputs (prompts(RC) and Attenuation Factors), the scatter contributions (MC-ground truth, DL and their difference), a comparison



Appendix V: Dataset 16 Evaluation

- The training dataset from which the optimal network performance occurred was estimated to be Dataset 16. It was trained using both analytical and voxelized phantoms, the projection views over the data, and with the RR preprocessing strategy.
- The following 5 groups of graphs correspond to the test cases used for the performance evaluation of the networks: NEMA-IQ, NEMA-Scatter, No-, OoFoV- and InFoV-Bladder phantoms, with (a),(b),(c),(d) and (e) respectively.
- As discussed, the outputs (DL Scatter) of the networks trained with RR datasets along their targets (MC Scatter), were blurred for the evaluation metric (NRMSE) to be estimated.
- In each of the following graphs, the top row corresponds to the pre-evaluation blurring and the bottom one to the post-evaluation blurring.
- The grayscale graphs represent MC and DL Scatter 3D arrays, whose mean scatter rates are being projected along the 3 different dimensions. For example, in each 3D plot, the mean scatter rates over the entire set of sinograms are visible on the bottom of the rectangle.
- The redscale graphs represent the MC-DL difference, as measured with the NMAE. The maximum values are being projected along the 3 different dimensions. For example, in each 3D plot, the mean scatter rates over all sinograms is visible on the bottom of the rectangle. This visualization is made to assist the locate the areas that contribute to the MC-DL scatter difference the most.