Machine Learning: Health

Crossmark

RECEIVED dd Month yyyy
REVISED dd Month yyyy

PAPER

Learning-Based Estimation of Spatially Resolved Scatter Radiation Fields in Interventional Radiology

Felix Lehner[1,3], Pasquale Lombardo[2], Susana Castillo[3,5], Oliver Hupe[1], and Marcus Magnor[3,4,5]

[1]Physikalisch-Technische Bundesanstalt (PTB), Braunschweig, Germany
[2]Belgian Nuclear Research Centre (SCK CEN), Boeretang, Mol, Belgium
[3]Institute for Computer Graphics, Technical University Braunschweig, Braunschweig, Germany
[4]Physics and Astronomy, University of New Mexico, New Mexico, USA
[5]Cluster of Excellence PhoenixD, Leibniz University Hannover, Hannover, Germany

E-mail: lehner@cg.cs.tu-bs.de, felix.lehner@ptb.de

Keywords: Interventional Radiology, Monte Carlo Simulation, FCNN, U-Net, Dataset, X-ray

Abstract

We present three variants of a lightweight, fully connected artificial neural network, suited for interactive estimation of three-dimensional, spatially resolved volumes of scattered radiation fields and a corresponding training pipeline for radiation protection dosimetry in medical radiation fields, such as those found in interventional radiology and cardiology. Accompanying, we present three different synthetically generated datasets with increasing complexity for training, generated using RadField3D, a Monte Carlo simulation application based on Geant4. As the primary scatter object, we employed the torso of a male Alderson RANDO phantom. On those datasets, we evaluate convolutional and fully connected architectures of neural networks to demonstrate which design decisions work well for reconstructing the fluence and spectra distributions over the spatial domain of such radiation fields. All our datasets, as well as our training pipeline, are published as open source in separate repositories.
To evaluate the presented neural networks, we define and assess several metrics. Across these measures, the model variants demonstrate good spatial agreement between predicted and ground-truth radiation fields, particularly within specific regions of interest within the radiation field. Of particular relevance for potential application in out-of-field dosimetry is the SMAPE of the scatter radiation field, which represents the most challenging metric and was consistently above 84 %.

1 Introduction

The spatial distribution of radiation in a volume is of interest for various applications in radiation protection and occupational dosimetry. Knowledge gained from the analysis of irradiation scenarios can be used for optimizing radiation exposure. This is important for work, for example, in nuclear power plants, radioactive waste disposal, or in radiological medicine, such as Interventional Radiology (IR), in which medical staff performs surgeries under fluoroscopy.

The current state of the art relies on the utilization of personal dosemeters, which are sufficient for spatially homogeneous radiation distributions, but inherently misestimate personal doses in inhomogeneous radiation fields.

In the context of IR, medical staff are exposed to inhomogeneous radiation fields due to their proximity to the patient, which complicates the accurate assessment of individual doses. The reliability of current personal dosimetry methods is contingent upon the uniform distribution of radiation fields. However, this assumption of uniformity does not hold in IR settings. To address this issue, previous research has proposed the use of computational dosimetry systems [1], which are designed to monitor the locations and postures of all individuals involved in an IR procedure. In order to assess doses in these exposure scenarios, such systems make use of currently available methods, that are mainly based on Monte Carlo Simulations (MCSs) [2].

Nonetheless, the existing MCSs of radiation transport lack the necessary speed for real-time dose calculations, a limitation that also affects the software used for Virtual Reality (VR) or Augmented Reality (AR) training. VR training systems can play a pivotal role in diminishing radiation exposure during interventions by enhancing the radiation awareness of medical staff, as evidenced by at least two research projects [3, 4]. Notwithstanding their significance, there is a paucity of reliable and realistic radiation dosimetry data to be used in real-time visualization with these systems.

To advance radiation protection, it is necessary to develop accelerated methods for the simulation of radiation transport. In this research work, we explore the surrogate model approach for MCS acceleration, whereby we learn spatially resolved radiation field distributions from a Monte Carlo Simulation. Subsequently, the neural networks serve as efficient models for radiation-field reconstruction, providing a real-time capable radiation-transport simulation method. Furthermore, we have generated and made available a variety of datasets for the training of our models and analogous ones. The generation of our datasets was facilitated by the extension and utilization of the RadField3D [5] simulation software which itself is built upon the Geant4 [6] MCS framework to ensure physical correctness, as Geant4 is well established and validated for the targeted use-case.

Our primary innovation is the creation of three datasets of spatially resolved, three-dimensional fluence and spectra distributions for the training and evaluation of neural networks for the reconstruction of radiations fields for IR procedures. We accompany these with concrete neural network implementations and design recommendations based on ablation studies. Additionally, we define several metrics useful for evaluating and comparing spatially resolved radiation field estimators.

1.1 Monte-Carlo Simulation in Radiation Protection

The technique of using MCS to calculate the radiation transport of complex measurements is already widely spread in various disciplines of dosimetry, from radiation protection to medical physics. Therefore, a set of well-established and tested general-purpose MCS toolkits for radiation transport already exists, such as Geant4 [6], MCNP [7], and EGSnrc [8]. As these toolkits are meant for general purpose applications ranging from astrophysics over nuclear reactor physics to dosimetry calculations, fine-tuning the MCS parameters for each specific use case is required. There are frameworks built upon these toolkits, such as GATE [9], which is based on Geant4, and it has already been used for assessing doses to medical staff in nuclear medicine by tailoring simulation models based on the real tracking of personnel movements as demonstrated by Rondón et al. [10]. However, as highlighted by Rondón et al., current MCS capabilities are still far from real-time as they require, at best, several tens of seconds if not minutes to be performed. This is particularly challenging in IR, as the calculation of radiation-fluence distributions in the space around the patient during interventions is

computationally demanding and can require hours of simulations. However, real-time simulation capabilities are becoming increasingly necessary for computational radiation protection dosimetry of staff, as proposed by prior research projects, such as the PODIUM project [1].

Besides the real-time dosimetry, the AR and VR applications for training interventional radiologists and associated personnel demand even faster execution speeds. Nevertheless, especially in AR and VR, we can trade absolute accuracy for speed within reasonable bounds. These applications have already been presented at various times in the last years, for example in 2012 by a c-arm training extension for VirtX [11] or by the perceptual study of Rainford *et al.* in 2023 [3]. All current implemented systems would benefit from the display of physically correct radiation fields during training, but at the moment these systems are only able to render radiation fields generated by simplified deterministic algorithms due to strict timing requirements. For these applications, achieving frame rates of at least 90 to 120 fps requires total rendering times per volume of approximately 8 to 11 ms to avoid perceptible visual latency [12].

1.2 Computational Personal Dosimetry

Real-world measurements in IR scenarios employing Active Personal Dosimeters (APDs) have already revealed the complexity of contemporary radiation fields and high local dose rates that professionals are exposed to [13]. These measurements also implied that the definition of a representational location for wearing a personal dosemeter on the body of the medical staff is not possible, and therefore, it is likely to both under- or overestimate the received doses by the staff. In the course of the PODIUM project, the researchers developed a camera-based tracking system that was integrated with a MCS application based on the MCNP [7] toolkit. The simulation divided the room into frustums projected from the surface of a sphere placed around the patient at the isocenter of the scene [14, 15]. The evaluation of each event was conducted as it traversed one of the surface segments of the sphere, effectively projecting it along the normal of that surface segment through the scene. The validation of the simulation was based on these frustums, with measurements of APDs worn by medical personnel during real interventions serving as the ground-truth. These interventions involved tracking the locations and postures of staff members.

1.3 Hybrid Methods and Surrogate Models for Radiation Transport MCS

Due to its notable slowness, as evidenced in the research by Vanhavere & Van Hoey [16], the MCS is ill-suited for real-time or even interactive applications. To address this issue, recent studies have explored the prospect of enhancing its calculation speed. These efforts entail the replacement of the most computationally inefficient components of the radiation transport simulations, which is referred to as a hybrid approach. These hybrid methods can be composed of denoising algorithms. The most promising results in this field are delivered by deep learning methods, that use artificial neural networks for transforming the noisy results of an early aborted MCS to those of a fully executed MCS, but with shorter execution times [17, 18, 19]. Although this approach has been the subject of more extensive research, it exhibits a significant disadvantage, as the MCS needs a certain minimum of statistics to be denoise-able which inevitably comes at the price of a lower processing speed compared to a single neural network. Consequently, it does not meet the requirements for real-time applications especially in the context of VR and AR applications previously mentioned.

Other approaches focus on the complete replacement of the MCS algorithm, with artificial neural networks. This innovative approach is called surrogate models and its feasibility was already demonstrated in a few studies for different scenarios [20, 21, 22, 23]. Most of these studies were conducted in the context of nuclear power plants, thus those methods aim to predict neutron radiation fields induced by nuclear isotopes. Therefore, those works were heavily focused on modeling shielding and its positioning regarding a static radiation source. For this use case,

Wang *et al.* proposed the use of a residual Fully Connected Neural Network (FCNN) for predicting neutron radiation fields under various shielding configurations [23]. As they could assume a static radiation source, they split the room into regions with similar conditions regarding the radiation field and trained multiple networks for each region. Concurrently, Ye *et al.* implemented a 3D-Convolutional Neural Network (CNN) for 3D reconstruction in a similar scenario [21]. This 3D-CNN takes the unshielded radiation fields and shielding voxel geometry, convolutes those two channels down to a flat vector which is then rearranged as a 3D radiation field after the shielding process. Another use case for FCNN is demonstrated in the research work of Yisheng *et al.* [22], where a FCNN is used to predict the spatially resolved photon flux in a room based on the data of a single detector at a known position. The output of the neural network is a low resolution grid with 2000 voxels. An example of the use of neural networks in the domain of this research work in the context of IR was provided by the research work of M. Villa Arias [20]. Here, a 3D-UNet is used to transform a CT-Scout-Scan to the surrounding radiation field around the patient induced by the CT-scanner.

This work aims to contribute to the corpus of existing surrogate model approaches by focusing on predicting dynamic radiation fields using lightweight FCNNs. This enables the prediction of partial radiation fields within the region of interest, thereby supporting real-time capable radiation field estimation and, in theory, also allowing for higher resolutions of radiation fields. In contrast to prior approaches, our approach additionally provides the local energy spectra, making it possible to use those networks for correcting the energy dependence of real world dosemeters. In the following, we present FCNNs inspired by the architecture of the networks used for Neural Radiance Fields (NeRFs) [24]. As these networks model geometry implicitly, and our datasets do not incorporate variations in patient geometry, a direct comparison with the work of Arias *et al.* [20] is neither meaningful nor feasible, given that their approach explicitly encodes patient geometry.

2 Materials and Methods

As there is a lack of publicly available, and in this context, usable datasets our first contribution is the generation and provision of three datasets for training spatially resolved dose rate predicting neural networks. For these three datasets, we implemented two distinct neural network architectures with additional variants for each, in order to analyze the influence of these architectures and specific hyperparameters on the quality of the predictions. For this purpose, we considered two basic architectures: an FCNN architecture inspired by NeRFs [24] and a 3D-U-Net [25] architecture which is based on the CNN architecture, but with an extra transposed path for reconstructing all voxels at once.

2.1 Datasets

We generated and provided three synthetic datasets of radiation fields for different situations with increasing parameter spaces. The datasets were all generated using the RadField3D [5] dataset generator. Per field, the generation took about 20 min on our AMD Epyc 9474F 48-core CPU. The resulting datasets consist of one radiation field per fixed set of parameters. These fields are stored in the RadFiled3D file format and contain three channels. One for the photons of the direct X-ray beam, one for the resulting scattered fields and one containing the binary voxel occupation of the modeled geometry. Each of the first two channels contains the following information per voxel: The simulated volumetric *photon fluence* Φ^3_γ (fluence per cubic volume of a voxel), the volumetric *photon energy distribution* $p(E_\gamma)$ and the *statistical error* ϵ_{rel} as described in the research work about the MCS application RadField3D. Additionally, all input parameters, such as the X-ray tube output spectra or the tubes position and direction, are stored per field.

DS-01: The first dataset we generated, contains 1250 radiation fields. Each field has a resolution of 50×50×50 voxels with each voxel having an extent of 2 cm in every dimension resulting in a total field extent of 1 m per dimension. The varying parameter in this first dataset is the direction of the beam with $\phi, \theta \in [0, 2\pi)$, while the X-ray tube output is fixed to a reference radiation quality of H-100 according to ISO 4037-1 [26] and the beam shape is fixed to a cone beam with an opening angle of $\angle_{\text{Beam}} = 10°$. The distance from the X-ray tube relative to the origin is 2.5 m. At the origin of the field, the headless torso of a male Alderson RANDO phantom [27] is placed to act as the scatter object for the radiation beam. This setting is derived from the validation measurements conducted for the RadField3D data generator [5], and, although simplified, it is considered realistic for modeling this first exposure scenario.

DS-02: The second dataset contains 2156 radiation fields and extends the previous one by the radiation source spectra as a varying parameter. Here we used random, but realistic, spectra for modeling primary radiation in the context of IR. The range of the parameters was defined based on empirical feedback from medical physicist and radiologists, mainly gained during the PODIUM project. Therefore, we defined an energy range for the tube output peak energy $E_{\text{tube}} \in [40, 125]$ keV. Further, we added aluminum with a random thickness $t_{\text{Al}} \in [2.5, 7.5]$ mm and, optionally, copper, with a random-thickness $t_{\text{Cu}} \in [0.0, 0.9]$ mm for simulating realistic filtering material used to improve the X-ray beam quality. As the anode material, we set tungsten, which is a standard reference material in IR imaging devices, and used a random anode angle of $\angle_{\text{Anode}} \in [8, 12]°$. From that parameter space for the radiation source, we sampled sets of parameters and passed them to SpekPy [28] to generate reasonable X-ray spectra for the X-ray beams in this dataset.

DS-03: In the third and most complex dataset, dynamic X-ray tube distancing is introduced. A total of 3779 distinct radiation fields were generated. For this dataset, we mostly carried over the same base parameter space as described in the preceding datasets, but in contrast to the previous datasets, we sampled the X-ray tube distance from the scenes isocenter as $d_{\text{tube}} \in [35, 75]$ cm. In order to make the dataset even more realistic in representing typical real-world scenarios, the X-ray beam was collimated to a rectangular shape, with fixed dimensions of 40 cm × 30 cm in the plane passing through the origin and thus intersecting the phantom.

Table 1. Listing of the statistical errors, with respect to the fluence, from the scattered radiation fields separated by dataset. mean(*error*) refers to the mean over all voxels in the dataset, while median(*error*) refers to the mean over the median statistical error of each file inside the dataset.

Dataset	mean(*error*)	median(*error*)
DS-01	9.7 %	0.03 %
DS-02	9.9 %	0.03 %
DS-03	9.9 %	0.03 %

RadField3D uses the variation of the relative voxel fluences during the simulation process to measure the statistical error per voxel. Those statistical errors from the scattered radiation field layer are evaluated continuously to determine the current overall statistical error. Once 90 % of those voxels exhibit a statistical error equal to or less than 10 %, the simulation is aborted. This behavior is reflected by Tab. 1 where we show the mean and median statistical errors present in our previously described datasets, as calculated by RadField3D.

An example of the spatial distribution of relative air kerma rates in air $\dot{K}_{\text{air}}$ can bee seen in Fig. 1. The two projections were generated from a random field taken from DS-03 by

excluding all doserates below 0.5 % of the maximum doserate for improved visual understanding.

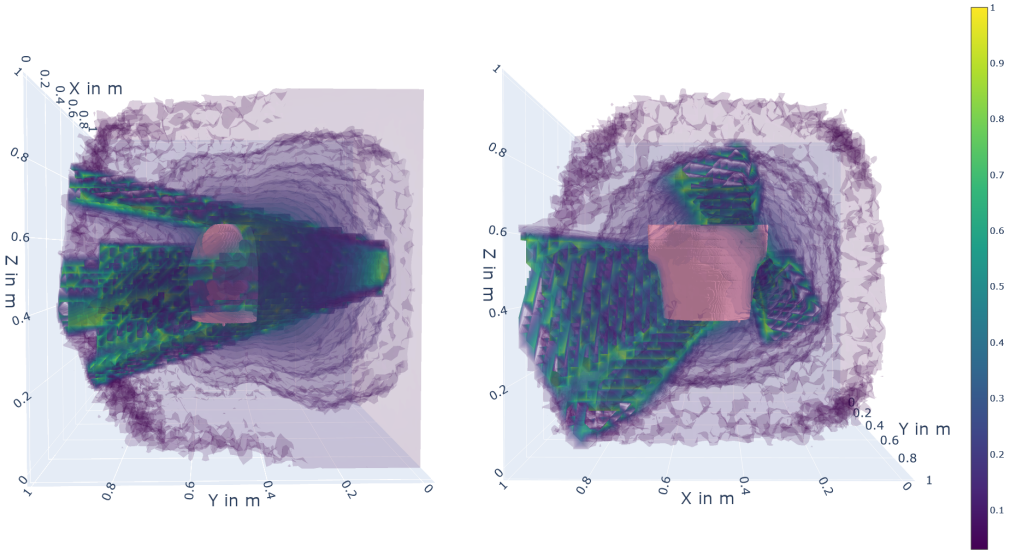

Figure 1. Two perspective projections of a random radiation field from DS-03: One projection along the X-axis (left) and one along Y-axis (right). The spatially resolved relative doserate $\dot{K}_{\text{air}}$ (air kerma in air) is color coded from transparent ($< 0.5\,\%$) over blue ($0.5\,\%$) to yellow ($100\,\%$). The used scatter geometry (Male Alderson RANDO phantom torso) is colored in red.

In Tab. 2, we have summarized the distribution of each input parameter inside the three datasets and additionally the distribution of the contained fluences by providing the dynamic ranges DR_{dB} and the Gini coefficient G_{Gini}, which are defined as:

$$DR_{\text{dB}} = 10 \log_{10}\left(\frac{\max(\Phi_\gamma^3)}{\min(\Phi_\gamma^3)}\right)$$

$$G_{\text{Gini}} = \frac{1}{n^2 \mu} \sum_{i=1}^{n} x_i (2i - n - 1)$$

where DR_{dB} is used to describe the actual ranges needed for reconstruction and G_{Gini} expresses the deviation of the present fluence distribution from a uniform distribution. For the calculation of G_{Gini}, the fluences are expected to be sorted in ascending order , where μ denotes the mean fluence and n the total count of voxels across the dataset. Finally, a Gini coefficient of $G_{\text{Gini}} = 0$ indicates that the dataset is uniformly distributed, while $G_{\text{Gini}} = 1$ indicates that all fluences of the dataset are accumulated in one single bin of the fluence histogram.

Table 2. Statistics over the presented datasets including all varying input parameters.

Dataset	$\overline{\angle}_{\text{Beam}} \pm \sigma$	$\overline{dist}_{\text{tube}} \pm \sigma$	$\overline{\text{E}} \pm \sigma$	$\overline{\text{keV}_\text{p}} \pm \sigma$	DR_{dB}	G_{Gini}
DS-01	$10.0° \pm 0.0$	$2.5\,\text{m} \pm 0.0$	$80.7\,\text{keV} \pm 0.0$	$100.0\,\text{keV} \pm 0.0$	43.9 dB	0.790
DS-02	$10.0° \pm 0.0$	$2.5\,\text{m} \pm 0.0$	$48.7\,\text{keV} \pm 9.3$	$82.1\,\text{keV} \pm 24.6$	53.5 dB	0.872
DS-03	–	$0.6\,\text{m} \pm 0.1$	$48.7\,\text{keV} \pm 9.3$	$82.1\,\text{keV} \pm 24.6$	76.0 dB	0.991

By observing the differences between our three datasets, one can see, that by reducing the volume of the direct beam inside the radiation field volume, the dynamic range DR_{dB} is

drastically increased. In the case of DS-03 when smaller opening angles of the beam and smaller distances of the X-ray tube from the isocenter are used, particle fluences Φ^3_γ span over nearly 8 orders of magnitude, instead of the spanning of about 5 orders of magnitude for the other datasets. Concurrently, the distribution of fluences is more challenging, as the amount of voxels with fluences below 0.1 % of the maximum fluence, rises by a factor of approximately 253 from 0.3 % to 76 %.

Overall, for all three datasets, the dataset generator RadField3D ensures, that 90 % of all voxels do not exceed a statistical error of 10 % with respect to fluence.

Data normalization: All datasets used are provided normalized by RadField3D. This applies both to the voxelwise photon energy spectra, which are probability distributions with unit integral, and to the voxelwise fluences, which are normalized with respect to the number primary particles of the simulation. This effectively scales all fluence values relative to the maximum, ensuring that all fluences fall within the [0..1] interval.

2.2 Architectures

In order to access the effectivity of our neural network architectures, we implemented each architecture's neural network model in three variants to handle the different provided parameters for each dataset. With this approach, for each architecture there is one model that can only handle X-ray tube rotations, one that can additionally handle varying spectra and one that can additionally handle varying tube distances.

In general, our models all estimate the relative topology of the spatial radiation field distribution conserving the relation of intensities between all voxels inside one specific radiation field. This is important, as the actual fluence scales between different settings of the beam would introduce a bias towards high X-ray tube currents and thus harm the training of the neural networks. On the downside, in a real-world application, this inherently requires our approach to run concurrently to at least one area dosemeter that reports dose rates at a known point in space. As these devices, especially those with real-time measurement capabilities, are known to exhibit a certain energy and thus spectrum dependence, one needs to correct them accordingly. This is specifically of importance in our targeted use case, as we aim to predict the scattered field around the patient, where the spectrum is not homogeneous and is difficult to predict. Thus, we decided to train all our networks not using the actual target measurand, *e.g.*, air kerma rate $\dot{K}_{\text{air}}$ free-in-air or ambient dose equivalent $H^*(10)$, but on the components of those measurands as provided by RadField3D, which are the volumetric *photon fluence* Φ^3_γ and the volumetric *photon energy distribution* $p(E_\gamma)$ composed of 32 bins and $E_{max} = 150\,\text{keV}$. In fact, the

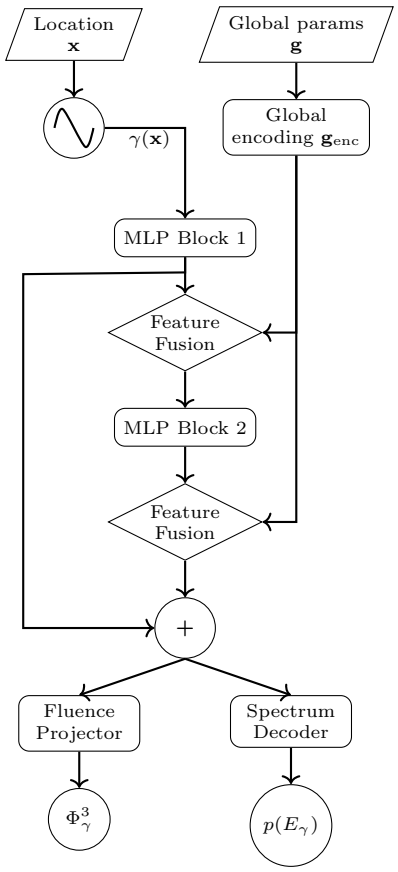

Figure 2. Schematic of the fully connected architecture used as the backbone for SRBFNet, SPERFNet and PBRFNet outputting a spectrum and a fluence for each voxel location.

concurrent measurements of an area dosemeter are only needed to correct the fluence. Therefore, one can potentially use the estimated spectra to correct an area dosemeter measurement for its energy and angle dependence in turn.

2.2.1 Fully-Connected-Voxelwise-Networks. The primary family of neural networks that we explored is the family of Fully Connected Neural Networks. Those networks are not only the most primitive, but also the most flexible ones. They have long been proven to be universal function approximators [29] and as such, given the right input representation, they can effectively learn to reconstruct structures of various kinds. More specifically, we were inspired by the NeRF approach of Mildenhall *et al.* [24] which is used for Novel View Synthesis (NVS). We chose this approach, as the reconstruction of radiance fields especially by follow up works for the real-time rendering application, seems promising for the reconstruction of radiation fields in real-time. As we chose to not only predict a scalar dose rate per voxel, but the photon fluence Φ_γ^3 together with the normalized local photon spectrum $p(E_\gamma)$, this matches the NeRF architecture, where the reconstruction of visual density and color of a given point is separated as well.

For the training process of our FCNNs, we used a cosine annealing [30] learning rate scheduler with $\text{eta}_{\text{min}} = 10^{-6}$ together with an Adam optimizer using weight decay [31] of 10^{-4} and $\beta_1 = 0.9 \wedge \beta_2 = 0.99$. Additionally, we preceded our main learning rate scheduler by a linear warmup phase that reaches our target learning rate after 1000 steps.

Static Rotatable Beam Field Network (SRBFNet): This neural network forms the basis for reconstructing the presented datasets. For each dataset, we present variations of this network. In general, SRBFNet receives the same input parameters as the original NeRF did: The voxel location in cartesian coordinates ($x, y, z \in [0, 1]$), normalized per dimension, and a normalized direction vector, which in this case is the X-ray tube direction. The overall structure is shown in Fig. 2, where global parameters $\mathbf{g}$ in this case is the encoded direction vector.

Our network begins by feeding the location vector through an encoder using frequency encoding composed of sine and cosine functions with different periodicities which were initially introduced by A. Vaswani *et al.* [32] for the transformer networks and reused for the original NeRF implementation. Even though, the direction vector can also be represented by a simple frequency encoding, we instead used a spherical harmonics based encoding as the cyclic nature of a rotation angle can be better represented by that encoding. This was demonstrated for NeRF-like networks by InstantNGP of T. Müller *et al.* [33]. For efficiency reasons, we used the spherical harmonics implementation as provided by the tiny-cuda-nn package [34] with version 2.0. For both encodings, we appended the raw input to the encoded vector. In contrast to the classical NeRF application, our network does not aim to learn the scatter objects 3D geometry with visual densities for volumetric rendering. Thus, we skipped the pre-processing path where the model should optimize the voxel location representation towards the local density independently from the direction. We want the network to learn the implicit representation of the fluence distribution, which is similar to the visual density, but in our case is ultimately dependent on the radiation direction. Thus, we extracted the fluence as well as the spectrum from the last layer. In order to build a robust backbone for solving the radiation field estimation under more complex parameter spaces, we did not just concatenate the feature vectors that describe the global beam conditions, such as the view direction in NeRF. Instead, we experimented with various feature fusion methods and finally compared the Feature-wise Linear Modulation (FiLM) [35] approach, which was originally used in CNNs, with a Gated Multimodal Unit (GMU) [36] approach and as the baseline, with the simple concatenation of feature vectors as is done in classical NeRF networks. Ahead of fusion both feature vectors, the encoded X-ray tube direction vector was passed through the global encoding block, a Multilayer

Perceptron (MLP) block that is similar to the MLP processing the encoded location vector with one hidden layer and a model width that is equal to the width of the main path of the network.

The actual hyperparameters used for this and the extensions of this network, such as the frequencies of the frequency encoding, spherical harmonics degrees, model widths and feature fusion layers are reported in the results section as we determined them using hyperparameter tuning. Through all layers of the main path of this network, except for the last ones, we use Sigmoid Linear Unit (SiLU) activation functions. For the output layer, we used two different activation functions. For the fluence, we used the sigmoid activation, to keep the output between 0 and 1:

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

For the local spectra, we use a simple histogram normalization $\mathrm{norm}_{\mathrm{hist}}$ as the activation function, to guarantee normalized spectra for the losses.

$$\mathrm{norm}_{\mathrm{hist}}(x) = \frac{x}{\sum_{i=0}^{n-1}(x_i)}, n = 32$$

SRBFNet was designed to be trained on the parameter space provided by **DS-01**.

Spectral Enhanced Radiation Field Network (SPERFNet): This network is the first extension of SRBFNet. It extends the global parameters $\mathbf{g}$ by the X-ray spectrum of the radiation source. Each radiation field file contains the 150 bin histogram of the X-ray tube output spectrum using a bin width of 1 keV. In order to improve the radiation field distribution and include the reconstruction of the local spectra, we encoded that spectrum into our network. First, we reduced the dimensionality drastically by bilinear resampling the bins of the histogram down to 64 bins. The reduced histogram was then passed through a MLP with one hidden layer using a width of 32 neurons and the SiLU activation function. After the first linear layer, we implemented a layer-normalization to reshape the activation distribution.

The two feature vectors of the X-ray tube direction and the X-ray tube output spectrum were then concatenated and combined by the global encoding block in Fig. 2 that previously processed the direction vector.

SPERFNet was designed to be trained on the parameter space provided by **DS-02**.

Parametric Beam Radiation Field Network (PBRFNet): This neural network targets the limitation of the previous described variant, that is only able to learn a fixed, static beam distance. Therefore, this model adds an additional encoder to allow for the additional beam parameter introduced by **DS-03** while keeping the same overall structure from SPERFNet. This requires one new scalar input parameter, that is the tube distance as the opening angle of the rectangular beam is directly dependent on the distance. We observed, that simply passing this scalar raw leads the neural network to ignore it. The previously used frequency encoding is not useful here either, as the input data is neither of high frequency nor periodical. Thus, we added a mini MLPs into the architecture, which has a width of 16 neurons and in an optimal way encodes the distance information for the global parameters vector $\mathbf{g}$.

PBRFNet was designed to be trained on the parameter space provided by **DS-03**.

2.2.2 Convolutional-3DVolume-Networks. Instead of predicting every single voxel by itself, which is inefficient when the aim is to always predict all voxels, especially for fine resolutions, one can use CNNs. These networks rely on using convolution kernels with multiple channels that

sweep over the image space and by that, encodes local information. Initially, these networks were only used for Computer Vision, but by transposing this operation, these networks can be used for generative purposes.

Beam2Scatter U-Net: Our Beam2Scatter (B2S) U-Net is in general a CNN, that encodes the raw radiation field of the direct beam prior to the interaction with the patient geometry and from that estimates the scattered radiation field as it is induced by the patient. Therefore, its encoder-decoder architecture follows the 3D-U-Net description [25], which itself is just an adaption to the third dimension of the original U-Net, as it was initially used for biomedical image segmentation [37]. We adopted this structure and gave our network the fluence map of the direct beam as a one-channel input and let the network predict a 33 dimensional output with the first 32 channels for the spectra and the last channel for the fluence. Another addition is, that we introduced FiLM [35] layers after each up-sampling step in the decoder path, to modulate the weights of the decoder depending on the encoded input spectrum of the X-ray tube, which is encoded the same way as in SPERFNet. This way we can inject that information into the U-Net structure. The networks outputs had the prior introduced activation functions applied: sigmoid and $\text{norm}_{\text{hist}}$. Thanks to the direct beam radiation distribution being the input of this network, it can be used directly with all three datasets. For **DS-01**, the spectra encoding and fusion by using FiLM layers was disabled.

2.3 Training pipeline

To ensure that the models are all comparable, the training pipeline for all models was identical. Regardless, whether the model itself was capable of predicting partial or whole fields, the evaluation of the loss functions and the sub sequential update of the weights was performed in the same way. All models were trained for a maximum of 200 epochs, which could be prematurely aborted when the validation loss was not improving for the last 10 epochs. For the training loop, we used the pytorch lightning [38] framework in version 2.5.6 which implements the Cross-Validation policy with a final test using a dedicated validation and test dataset. The ratio of the split into training, validation and test dataset was 70 %, 15 % and 15 %. At first, a number of fields were loaded from the dataset that are transformed to pytorch tensors, preserving the channels and single layers, and as such they were batched together. We used a physical batch size of 4 full radiation fields. Multiple batches of those predicted and ground-truth fields were aggregated together to an effective batch size of 64 for training. We applied data pre-processing to each field prior to the prediction. This pre-processing included not only the option of joining the direct beam and scattered field channels, but also normalizing the ground-truth and input tensors. Finally, the loss functions for each of the neural networks normalized outputs (fluence and spectra) are calculated. The metrics are calculated on the inverse-transformed data in the original data space.

The initial learning rates were estimated per neural network by using the pytorch lighting implementation of a learning rate finder, which follows the descriptions of L. N. Smith [39]. We limited it to search on the interval of $[10^{-2}, 10^{-4}]$ for 250 steps.

Per voxel training: For the FCNN models, which operate per voxel, we added an optional extra step to the training pipeline, that is typical for NeRF approaches. We refer to it as voxel location randomization, which is essentially a perturbation of the voxel position around its center with a maximum jitter of half the voxel's dimension. This was used as a data augmentation to stabilize training in NeRF variants like InstantNGP [33].

2.3.1 Loss functions Across all our models we used the same two loss functions, that we optimized concurrently: $\mathcal{L}_{\Phi^3_\gamma}$ the loss regarding the volumetric fluence Φ^3_γ and $\mathcal{L}_{p(E_\gamma)}$ the loss regarding the local photon spectrum $p(E_\gamma) \in \mathbb{R}^{32}$ for $\sum_{i=0}^{31} p(E_{\gamma,i}) = 1$.

$$\mathcal{L}_{\Phi^3_\gamma}(t,p) = \frac{1}{3}\Big(\mathrm{L}_1(t,p) + \mathrm{L}_2(t,p) + \big(1 - \mathrm{SSIM}(t,p)\big)\Big), \text{ with } t,p \in \mathbb{R}$$

$$\mathrm{CumSum}_i(x) = \begin{cases} x_i, & i = 0, \\ \mathrm{CumSum}_{i-1}(x) + x_i, & i > 0. \end{cases}$$

$$\mathrm{Wasserstein}(t,p) = \frac{1}{n}\sum_{i=0}^{n-1} |\mathrm{CumSum}_i(p) - \mathrm{CumSum}_i(t)|, \text{ with } t,p \in \mathbb{R}^n$$

$$\mathcal{L}_{p(E_\gamma)}(t,p) = \frac{3}{10}L_1(t,p) + \frac{7}{10}\mathrm{Wasserstein}(t,p), \quad \text{with } t,p \in \mathbb{R}^{32}$$

With t being the ground truth (target) and p being the prediction of the neural network. A quick exploration of the parameter space had lead to the presented ratio between Wasserstein and L_1-Loss and was beneficial for achieving satisfactory results. Here, L_1 and L_2 refer to the L1- and L2-Norms using mean aggregation to produce a scalar loss for higher dimensional inputs.

2.3.2 Hyperparameter tuning For the hyperparameter tuning, we used the Optuna [40] framework with version 4.5. For the tuning of each network, we defined the search space for possible parameters as listed in Tab. 3. The tuning was, in general, performed on every model we discuss: B2S U-Net, SRBFNet, SPERFNet, PBRFNet. Anyway, not all of the parameters could be tested for each model. For instance, the SRBFNet does not encode any spectra information, therefore the spectra encoding related parameters were not probed.

Table 3. Probed hyperparameter spaces for each parameter of the models (B2S U-Net, SRBFNet, SPERFNet, PBRFNet), where applicable.

Model parameter	Parameter space
model width	[64, 96, 128, 192, 256, 384]
location encoding frequencies	[10, 12, 14]
direction encoding harmonics	[4, 6, 8]
feature fusion	[Concatenation, FiLM, ResFiLM, GMU]
random voxel location	[true, false]
encoded spectra dimensions	[16, 32, 64]

2.4 Validation Metrics

To assess the quality of the developed models, we need to analyze their performance on the test section of our datasets. As the distribution of voxel values inside our dataset are unique to our use case and we have special requirements as well, we did not aim for perceptual accuracy as in most other reconstruction tasks, we first needed to assess which metrics and which variants of metrics are applicable for neural network evaluation of this kind. In general, we measure our metrics over the air kerma rate free-in-air fields $\dot{K}_{\mathrm{air}}$ of ground-truth and predicted fields using the following three metrics classes.

Symmetric Mean Absolute Percentage Error (SMAPE): The first and most obvious kind of metric we employed is the SMAPE which provides the mean relative error across all voxels of the test dataset. As the dataset is strongly biased towards low voxel values we introduced this metric in two variants: $\text{SMAPE}_{acc}^{x\,\%}$ for the metric applied over all voxels whose values exceed $\max(K_{\text{air}}) * (1 - \frac{x}{100})$ and $\text{SMAPE}_{acc}^{scatter}$, which applies the metric on all voxels between 5 % and 0.5 % of $\max(K_{\text{air}})$.

The SMAPE is defined in the following, where $\mathbf{P}$ and $\mathbf{T}$ are two sets of voxels, present in two corresponding K_{air} fields. $\mathbf{T}$ is the set of ground truth voxels and $\mathbf{P}$ is the set of predicted voxels:

$$\text{SMAPE} = \frac{2}{n} \sum_{i=1}^{n} \frac{|P_i - T_i|}{|T_i| + |P_i|}$$

$$\text{SMAPE}_{\text{acc}} = 1 - \frac{1}{2}\text{SMAPE}$$

With $\text{SMAPE}_{\text{acc}}$ referring to the SMAPE accuracy metric reported in this work.

Structural Similarity Index Measure (SSIM): This metric is a well established approach to measuring the similarity in terms of structure between a target and a prediction image or volume [41]. This metric is commonly used for reconstruction tasks and expresses the structural similarity between target and prediction by an intuitive value between 0 and 1, where 1 indicates perfect accordance between both structures.

$$\text{SSIM}(p,t) = \frac{(2\mu_p\mu_t + C_1)(2\sigma_{pt} + C_2)}{(\mu_p^2 + \mu_t^2 + C1)(\sigma_p^2 + \sigma_t^2 + C_2)}$$

with μ_x being the mean of the window x, σ_x being the variance of the window x, and σ_{xy} being the covariance matrix of the two windows x and y. The $\text{SSIM}(p,t)$ is calculated for small windows p and t of the compared volumes $\mathbf{P}$ and $\mathbf{T}$, and the global SSIM is calculated by the weighted sum of all windows. C_1 and C_2 are calculated from the dynamic range of the compared volumes as stabilization constants to prevent terms from degrading to zero.

Gamma Passing Ratio (GPR): The GPR within the scope of a gamma evaluation is a common metric in the field of radiotherapy radiation transport simulation validation [42]. It reflects the relative number of voxels that pass a certain criterion within a given radiation field. The metric variant we implemented follows the idea of the Maximum Allowed Dose Difference (MADD) as described by Jiang *et al.* [43]. For our validation we used two different criteria. The strict criteria, which measures the number of voxels with a MADD of 3 % using a spatial resolution of 4 cm and the more relaxed criterion, which measures the number of voxels with a MADD of 10 % and a spatial resolution of 6 cm. To calculate the GPR, the field is shifted by the allowed Δ_d in each dimension, respecting the extent of a single voxel. For $\Delta_d = 4\,\text{cm}$ this results in 2 shifts per dimension for the case of a voxel size of 2 cm. The minimum distance weighted error for each voxel across the applied shifts is used as the error for the scoring of the passing voxels, thus the error $\gamma(\vec{x})$ for a voxel in a voxel grid $\mathbf{X}$ at position $\vec{x} \in \mathbf{X}$ is defined as:

$$\gamma_{\Delta_{\dot{D}}}^{\Delta_d}(\vec{x}) = \min_{\vec{r} \in \mathbf{V}_{\Delta_d}(\vec{x})} \left(\sqrt{\left(\frac{\|\vec{x} - \vec{r}\|}{\Delta_d}\right)^2 + \left(\frac{\dot{D}_p(\vec{x}) - \dot{D}_t(\vec{r})}{\Delta_{\dot{D}}}\right)^2} \right), \mathbf{V}_{\Delta_d}(\vec{x}) = \{\vec{v} \,|\, \|\vec{v} - \vec{x}\| <= \Delta_d \wedge \vec{v} \in \mathbf{X}\}$$

with $\dot{D}_t$ being the target dose rate in K_{air} at a given point and $\dot{D}_p$ being the predicted one.

The reported metric $\text{GPR}_{p\,\%}^{d\,\text{cm}}$ follows from that as:

$$\text{GPR}_{p\,\%}^{d\,\text{cm}} = \frac{|\{\vec{x} \,|\, \gamma_p^d(\vec{x}) \leq 1 \wedge \vec{x} \in \mathbf{X}\}|}{|\mathbf{X}|}$$

Spectrum Accuracy: As the spectra represent a special kind of information that can be used for purposes other than direct dosimetry, we evaluate them separately. Therefore, we use the Intersection over Union (IoU), where we calculate the ratio between the combined area of ground truth and prediction spectra and the overlapped area between them. This accuracy is defined as in the following, where **P** and **T** are two sets of voxels in two corresponding spectra radiation fields. **T** is the set of ground truth voxels and **P** is the set of predicted voxels:

$$\mathrm{Spec}_{acc} = \frac{\sum_{i=1}^{n} \min(P_i, T_i)}{\sum_{i=1}^{n} \left(P_i + T_i - \min(P_i, T_i)\right)}$$

3 Results

In general, the qualities of the developed models by this research work are condensed in Tab. 4. Therein, each metric value belongs to a network, trained, tested and optimized in terms of hyperparameters on one of our datasets. The first dataset **DS-01**, was used to train our base FCNN architecture called SRBFNet. We compared the results of that network against the implementation of the U-Net that we introduced previously as Beam-to-Scatter (B2S) U-Net. As this network does not need any architectural changes to be applied on multiple datasets, as its inputs are the raw primary beam layer of a radiation field together with the X-ray tube output spectra, we used this as a reference for each dataset and variant of our FCNN architecture. We report the best metric values our networks achieved after we optimized the models hyperparameters by using Optuna as mentioned in the previous section. The specific found hyperparameters for each network are listed in Tab. 5.

Table 4. Reported reconstruction metrics for each network in its optimal configuration of hyperparameters. All metrics were calculated from the test dataset section, grouped by used dataset. For the training of the models a NVIDIA H100 GPU was used.

Dataset	Model	$\mathrm{SMAPE}_{acc}^{90\,\%}$	$\mathrm{SMAPE}_{acc}^{scatter}$	SSIM	$\mathrm{GPR}_{3\,\%}^{6\,\mathrm{cm}}$	$\mathrm{GPR}_{10\,\%}^{4\,\mathrm{cm}}$	Spec_{acc}
DS-01	SRBF	**96.4** %	**95.5** %	**0.961**	**99.8** %	**99.98** %	**84.2** %
	B2S U-Net	87.1 %	64.0 %	0.868	84.9 %	99.0%	72.5 %
DS-02	SRBF	90.0 %	83.3 %	0.923	96.8 %	99.8 %	59.5 %
	SPERF	**97.1** %	**96.2** %	**0.952**	**99.7** %	**99.9** %	**86.2** %
	B2S U-Net	88.0 %	55.5 %	0.844	72.7 %	98.1%	77.1 %
DS-03	PBRF	**84.4** %	**84.8** %	**0.902**	**87.2** %	**94.4** %	**86.7** %
	B2S U-Net	19.5 %	19.4 %	0.802	21.8 %	30.5 %	14.2 %

The resulting metrics on the test dataset, which we reported in Tab. 4, were achieved using the following training seeds per dataset-model combination: DS-01(SRBF): 1161473710, DS-01(B2S U-Net): 4043505070, DS-02(SRBF): 1161473710, DS-02(SPERF): 3408822816, DS-02(B2S U-Net): 6194256017, DS-03(PBRF): 996092023, DS-03(B2S U-Net): 4178513325.

On **DS-01**, one can easily see, that the FCNN architecture outperforms the U-Net architecture. Even though, the U-Net also learns the structure of the radiation field, as reflected by the SSIM scores in Tab. 4, the edges between primary beam and scattered radiation are not reproduced as sharply as the FCNN does. This can be concluded from the relation between SMAPE accuracy on both regions and the SSIM scores, which drops less than the accuracy. This observation can be reproduced for **DS-02**. Additionally, we intended to highlight the effectiveness of our spectra encoding for the expected air kerma rates $\dot{K}_{\mathrm{air}}$. Thus, we trained the base SRBFNet model, which cannot handle different output spectra of the X-ray tube, on **DS-02**. The effectivity of our spectra encoding can be seen by comparing SRBFNet against SPERFNet, where we get a nominal difference of 12.9 % for $\mathrm{SMAPE}_{acc}^{scatter}$ just from the missing spectra information. Furthermore, when comparing SRBF on DS-01 with SPERF on DS-02 in Tab. 4, we

can conclude, that training our FCNN architecture on a bigger dataset using more degrees of freedom, namely the X-ray tube spectrum, actually improves the SMAPE_{acc}, even though the spatial resolution , according to the drop in the SSIM score, got slightly, but not notably, worse.

For **DS-03** and PBRFNet, we observed, that using models with a higher neuron count by increasing the width of our model per layer beyond 192 neurons, *e.g.*, 256 or even 384, the reconstruction capability increases by up to 1.5 % regarding $\text{SMAPE}_{acc}^{scatter}$. A similar increase of the SSIM on the other hand side, and thus the overall shape of the radiation field, is invariant to a change of the model width beyond 192 across all our datasets and models. As the benefit of using broader models is not significant enough to sacrifice performance for it, we limited the hyperparameter selection to models with a maximum model width of 192.

From Tab. 6, we can see, that the use of GMU feature fusion is doubling the inference durations and decreases the learning capabilities consistently over all datasets and FCNN based models. The use of FiLM feature fusion, on the other hand, is faster and often consistently beneficial for the SMAPE accuracy metric over the scattered radiation field region. In general, an inference duration for a whole radiation field estimation of about 20 ms was achieved by all our FCNN based models, when used with concatenation or FiLM based feature fusion. Please note, that the corresponding B2S-U-Net implementation achieved a mean inference duration of 7.5 ms ± 2.0 ms on the same datasets and hardware using FiLM conditioning in the center layer.

Table 5. Used model configurations, where L denotes the number of frequencies used for the Fourier feature encoding of the voxel location, l_{max} denotes the maximum degree used for the encoding of the radiation direction by spherical harmonics and $\dim(\mathbf{g}_{\text{spec}})$ indicates the dimensionality of the encoded input spectrum. Model width refers to the number of neurons in one full layer of the MLP main path.

Dataset	Model	Model Width	L	l_{max}	$\dim(\mathbf{g}_{\text{spec}})$	Conditioning
DS-01	SRBF	192	10	4	–	FiLM
	B2S U-Net	–	–	–	–	–
DS-02	SRBF	192	10	4	–	FiLM
	SPERF	192	10	4	16	FiLM
	B2S U-Net	–	–	–	–	FiLM
DS-03	PBRF	192	14	4	16	FiLM
	B2S U-Net	–	–	–	–	FiLM

Table 6. Inference durations per field (50^3 voxels) of each model, regarding a specific feature fusion method, using non-compiled pytorch models and a NVIDIA RTX 4090 GPU in a Intel i9-13900K system with 128 GB of RAM installed. All models were using the following configuration with a model width of 192 and encoding parameters $l_{max} = 4 \wedge \text{L} = 10$ for SRBFNet and SPERFNet, with one exception for PBRFNet, which was using $\text{L} = 14$.

Model	Feature fusion	Inference duration per field	$\text{SMAPE}_{acc}^{scatter}$
SRBF	Concat	20.7 ms ± 2.8 ms	94.3 %
	FiLM	20.6 ms ± 1.9 ms	**95.5** %
	ResFiLM	24.5 ms ± 1.8 ms	94.4 %
	GMU	41.0 ms ± 2.6 ms	88.7 %
SPERF	Concat	21.8 ms ± 2.6 ms	95.2 %
	FiLM	21.8 ms ± 2.6 ms	**96.2** %
	ResFiLM	24.7 ms ± 1.6 ms	95.4 %
	GMU	41.7 ms ± 2.5 ms	93.7 %
PBRF	Concat	21.0 ms ± 1.3 ms	79.2 %
	FiLM	22.2 ms ± 2.5 ms	**84.8** %
	ResFiLM	24.3 ms ± 1.9 ms	67.8 %
	GMU	40.4 ms ± 2.4 ms	40.9 %

4 Discussion

The configuration of the best performing models during the hyperparameter tuning are listed in Tab. 5. When additionally reviewing the table of Tab. 4, we are convinced, that using a model width of 192 and a location encoding with frequencies within a range of 10 to at max 14 is sufficient for the presented use case. Consistently over all datasets, the angular resolution of the beam needed to solve this problem was found to be low enough, that a harmonics count of 4 for the spherical harmonics encoding of the beam direction is sufficient. Further, two plain FiLM layers for feature fusion, one at an early and one at a late position in the network, was proven to be most effective for this use case compared to using GMU layers, Residual-FiLM layers and feature concatenation. Even though, the impact of the FiLM layers on the inference duration is measurable, it is small compared to the concatenation that is commonly used in the field of Novel View Synthesis, as it can be seen in Tab. 6. In that table, we concentrate on the differences regarding $\mathrm{SMAPE}_{acc}^{scatter}$, as this metric is reflecting the accuracy on the most interesting part of the predicted volume for the computational dosimetry. That is because the staff, during an IR procedure, will be mainly monitored in the $\mathrm{SMAPE}_{acc}^{scatter}$ volume.
From Tab. 6, one can also draw the conclusion, that the use of FiLM layers did not only achieve the best accuracies, but also the shortest inference durations compared to GMU or Residual-FiLM. Moreover, we demonstrated, that the radiation transport simulation can be learned by the same deep, but narrow structure that is used to learn light transfer as in the use case of NVS. As the provided implementation does not use custom, native GPU codes like implementations in CUDA or OpenCL, its inference time does not enable real-time usage in the sense of the required 8 ms to 11 ms for VR- or AR-applications. Nevertheless, an inference time of about 20 ms is suitable for interactive applications. After all, with the provided FCNN models, state-of-the-art inferences times, that are suitable for the VR- and AR-application, should be achievable by implementing the network as CUDA or OpenCL kernels in the future, as has already been demonstrated by InstantNGP [33].

5 Conclusion

We have presented three datasets for training machine learning and especially deep learning algorithms on spatially resolved and physically accurate radiation field distributions as they could arise during IR procedures. Additionally, we have implemented and compared different variants of fully connected and convolutional neural networks, following the architectural design of U-Nets and NeRF-like architectures. Our results indicate that NeRF-like architectures are effective and suitable for radiation protection applications, and are therefore likely applicable to the training of medical staff in the context of IR procedures. We have also pointed out, which design decisions are effective in this special context. Our presented datasets together with the presented findings on the architectural decisions for neural reconstruction networks create a profound foundation for enabling further research on this topic, *e.g.*, extending our or similar approaches by uncertainty handling in neural networks or incorporation of dynamic scatter geometry.

6 Future work

We leave it as future work to implement the presented FCNN using fused CUDA kernels to improve their performance. In doing so, it is expected that the networks will achieve inference times suitable for AR/VR rendering, as we have presented lightweight NeRF-like architectures with two FiLM layers. We demonstrated that these FiLM layers only add minimal overhead. In general, this approach was already demonstrated for NeRF variants like InstantNGP [33].
Further, we empathize the importance of public available datasets for training and comparing neural networks for dosimetry. Especially, more datasets with increasing realism are needed by

adding beam collimation that is independent from the tubes distance from the isocenter or by adding various patient shapes and poses to the dataset generation process.

References

[1] U. O'Connor, C. Walsh, D. Gorman, G. O'Reilly, Z. Martin, P. Madhavan, R. T. Murphy, R. Szirt, A. Almén, M. Andersson, A. Camp, V. Garcia, M. A. Duch, M. Ginjaume, M. Abdelrahman, P. Lombardo, and F. Vanhavere. Feasibility study of computational occupational dosimetry: evaluating a proof-of-concept in an endovascular and interventional cardiology setting. *Journal of Radiological Protection*, 42(4):041501, oct 2022.

[2] N. Chatzisavvas, G. Priniotakis, M. Papoutsidakis, D. Nikolopoulos, I. Valais, and G. Karpetas. Monte Carlo computational software and methods in radiation dosimetry. *Annals of Emerging Technologies in Computing*, 5(3):36–50, 06 2021.

[3] L. Rainford, A. Tcacenco, J. Potocnik, C. Brophy, A. Lunney, D. Kearney, and M. O'Connor. Student perceptions of the use of three-dimensional (3-D) virtual reality (VR) simulation in the delivery of radiation protection training for radiography and medical students. *Radiography*, 29(4):777–785, 2023.

[4] A. Fujiwara, S. Fujimoto, R. Ishikawa, and A. Tanaka. Virtual reality training for radiation safety in cardiac catheterization laboratories - an integrated study. *Radiation Protection Dosimetry*, 200(15):1462–1469, 09 2024.

[5] F. Lehner, P. Lombardo, S. Castillo, O. Hupe, and M. Magnor. RadField3D: a data generator and data format for deep learning in radiation-protection dosimetry for medical applications. *Journal of Radiological Protection*, 45(2):021508, May 2025. Publisher: IOP Publishing.

[6] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, and et al. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303, 2003.

[7] C. J. Werner, J. S. Bull, C. J. Solomon, F. B. Brown, G. W. McKinney, M. E. Rising, D. A. Dixon, R. L. Martz, H. G. Hughes, L. J. Cox, A. J. Zukaitis, J. C. Armstrong, R. A. Forster, and L. Casswell. MCNP Version 6.2. *Release Notes*, 2 2018.

[8] National Research Council of Canada. Metrology Research Centre. Ionizing Radiation Standards. EGSnrc: software for Monte Carlo simulation of ionizing radiation, 2021. Collection: NRC Publications Archive. Record identifier: 7fbd0900-681b-4e95-95b8-b0b6082e7776 ER.

[9] S. Jan, G. Santin, D. Strul, S. Staelens, and et al. GATE: a simulation toolkit for PET and SPECT. *Physics in Medicine & Biology*, 49(19):4543, sep 2004.

[10] D. Santiago Rondón, P. Lombardo, M. Abdelrahman, L. Struelens, F. Vanhavere, W. Schoonjans, and N. Bergans. Computational system for extremity dosimetry in nuclear medicine. *Journal of Radiological Protection*, 45(4):041502, oct 2025.

[11] M. Wagner, K. Dresing, and W. Ludwig. SIScaR-GPU: fast simulation and visualization of intraoperative scattered radiation to support radiation protection training. *Studies in Health Technology and Informatics*, 180:968–972, 2012.

[12] J. Wang, R. Shi, W. Zheng, W. Xie, D. Kao, and H.-N. Liang. Effect of Frame Rate on User Experience, Performance, and Simulator Sickness in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2478–2488, May 2023.

[13] O. Hupe and U. Ankerhold. Determination of the dose to persons assisting when X-radiation is used in medicine, dentistry and veterinary medicine. *Radiation Protection Dosimetry*, 144(1-4):478–481, 11 2010.

[14] M. Abdelrahman, P. Lombardo, F. Vanhavere, A. Seret, C. Phillips, and P. Covens. First steps towards online personal dosimetry using computational methods in interventional radiology: Operator's position tracking and simulation input generation. *Radiation Physics and Chemistry*, 171:108702, 2020.

[15] F. Vanhavere, O. Van Hoey, P. Lombardo, M. Abdelrahman, M. Amor Duch, A. Almén, U. O'Connor, R. Tanner, and E. Carinou. *D9.121 – Final Report of the PODIUM project.* EC - European Commission, Belgium, January 2020. Score=1.

[16] F. Vanhavere and O. Van Hoey. Advances in personal dosimetry towards real-time dosimetry. *Radiation Measurements*, 158:106862, 2022.

[17] M. Raayai Ardakani, L. Yu, D. R. Kaeli, and Q. Fang. Framework for denoising Monte Carlo photon transport simulations using deep learning. *Journal of Biomedical Optics*, 27(8):083019, August 2022.

[18] T. Bai, B. Wang, D. Nguyen, and S. Jiang. Deep dose plugin: towards real-time Monte Carlo dose calculation through a deep learning-based denoising algorithm. *Machine Learning: Science and Technology*, 2(2):025033, apr 2021.

[19] Z. Peng, H. Shan, T. Liu, X. Pei, G. Wang, and X. G. Xu. MCDNet – a denoising convolutional neural network to accelerate Monte Carlo radiation transport simulations: A proof of principle with patient dose from X-ray CT imaging. *IEEE Access*, 7:76680–76689, 2019.

[20] M. Villa Arias. *Radiation exposure in x-ray guided interventions by using deep learning and Monte Carlo simulations.* Theses, Université de Bretagne occidentale - Brest, May 2023.

[21] R. Ye, D. Niu, L. Li, and X. Hou. Research on a three-dimensional radiation field reconstruction algorithm based on an improved 3D CNN. *Applied Radiation and Isotopes: Including Data, Instrumentation and Methods for Use in Agriculture, Industry and Medicine*, 214:111540, December 2024.

[22] H. Yisheng, W. Zhen, P. Yanheng, Z. Yuhang, Q. Rui, Z. Hui, and L. Junli. Validation of the neural network for 3D photon radiation field reconstruction under various source distributions. *Frontiers in Energy Research*, 11, April 2023. Publisher: Frontiers.

[23] X. Wang and B. Li. Research on 3D radiation fields reconstruction based on FNN. *Advances in Engineering Technology Research*, 2025.

[24] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *Commun. ACM*, 65(1):99–106, December 2021.

[25] Ö. Çiçek, A. Abdulkadir, and S. S. Lienkamp. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In Sebastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gozde Unal, and William Wells, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 424–432, Cham, 2016. Springer International Publishing.

[26] International Organization for Standardization. ISO 4037-1: Radiological protection — X and gamma reference radiation for calibrating dosemeters and doserate meters and for determining their response as a function of photon energy. https://www.iso.org/standard/66872.html, 2019. Last Accessed: 2024-10-23.

[27] Radiology Support Devices, Inc. Rando male anthropomorphic phantom. https://www.rsdphantoms.com, 2025. Model RAN-110.

[28] G. Poludniowski, A. Omar, R. Bujila, and P. Andreo. Technical Note: SpekPy v2.0 — a software toolkit for modeling X-ray tube spectra. *Medical Physics*, 48(7):3630–3637, 2021.

[29] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, January 1989.

[30] I. Loshchilov and F. Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

[31] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[32] A. Vaswani, N. Shazeer, and N. Parmar. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[33] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4), July 2022.

[34] T. Müller. Tiny CUDA Neural Networks, 2021.

[35] E. Perez, F. Strub, and H. de Vries. FiLM: Visual Reasoning with a General Conditioning Layer. *AAAI Conference on Artificial Intelligence*, 32(1), April 2018.

[36] J. Arevalo, T. Solorio, M. Montes-y Gómez, and F. A. González. Gated multimodal networks. *Neural Comput. Appl.*, 32(14):10209–10228, July 2020.

[37] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[38] William Falcon and The PyTorch Lightning team. PyTorch Lightning, March 2025.

[39] L. N. Smith. Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472, 2017.

[40] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A Next-generation Hyperparameter Optimization Framework. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 2623–2631, New York, NY, USA, July 2019. Association for Computing Machinery.

[41] Z. Wang, A. C. Bovik, and H. R. Sheikh. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.

[42] L. Yu, T. L. S. Tang, and N. Cassim. Analysis of dose comparison techniques for patient-specific quality assurance in radiation therapy. *Journal of Applied Clinical Medical Physics*, 20(11):189–198, October 2019.

[43] S. B. Jiang, G. C. Sharp, and T. Neicu. On dose distribution comparison. *Physics in Medicine and Biology*, 51(4):759–776, February 2006.