

Gradient-descent-based reconstruction for muon tomography based on automatic differentiation in PyTorch

Jean-Marco Alameddine,^{a,*} Felix Sattler,^a Maurice Stephan^a and Sarah Barnes^a

^a*German Aerospace Center, Institute for the Protection of Maritime Infrastructures,
Fischkai 1, Bremerhaven, Germany*

E-mail: jean-marco.alameddine@dlr.de

Muon scattering tomography is a well-established, non-invasive imaging technique using cosmic-ray muons. Simple algorithms, such as PoCA (Point of Closest Approach), are often utilized to reconstruct the volume of interest from the observed muon tracks. However, it is preferable to apply more advanced reconstruction algorithms to efficiently use the sparse muon statistics that are available. One approach is to formulate the reconstruction task as a likelihood-based problem, where the material properties of the reconstruction volume are treated as an optimization parameter.

In this contribution, we present a reconstruction method based on directly maximizing the underlying likelihood using automatic differentiation within the PyTorch framework. We will introduce the general idea of this approach, and evaluate its advantages over conventional reconstruction methods. Furthermore, first reconstruction results for different scenarios will be presented, and the potential that this approach inherently provides will be discussed.

Fifth MODE Workshop on Differentiable Programming for Experiment Design (MODE2025)
8-13 June 2025
Kolymbari, Crete, Greece

*Speaker

1. Introduction

Muon tomography is a powerful imaging technique which exploits the properties of naturally occurring cosmic-ray muons. The Earth is constantly hit by high-energy particles, notably protons and heavier atomic nuclei, originating from the cosmos. As these particles interact with the Earth's atmosphere, cascades of secondary particles are initiated. From these cascades, especially muons are capable of reaching the Earth's surface, leading to a nearly constant flux of one muon per cm^2 per min [1]. Due to their large mass, which is more than 200 times higher than the mass of an electron, as well as the absence of strong interactions, muons possess a tremendous penetrating power, making them highly valuable for tomographic tasks.

As muons pass through a medium, they interact with atomic nuclei through Coulomb scattering, leading to a deflection from their normally straight path. Since the magnitude of these deflections is correlated to the properties of the traversed material, mainly the mass density and the atomic number, it is possible to measure the muons before and after passing an unknown object, and obtain material information from the observed directional changes. This method is called muon scattering tomography. Compared to alternative imaging techniques, such as X-ray scanning systems, muon scattering tomography offers several advantages: Since muons occur naturally, no active sources are needed, which means that no additional radiation exposure for humans and animals is necessary, reducing the bureaucratic effort of dealing with radiation protection and eliminating any possible health hazards. Furthermore, the penetrating power of muons allows for the examination of heavily shielded materials, which poses a problem for traditional methods. Both points make muon scattering tomography especially interesting for an application in the security domain, for example for contraband detection at border crossings, airport, or harbors [2].

Since muon detectors are well-understood and commercially available [2], the most important remaining challenge in muon scattering tomography is the most efficient usage of the limited available muon statistics to be able to reconstruct materials with adequate accuracy in an acceptable time. Simple geometric reconstruction approaches, such as the Point of Closest Approach (PoCA) algorithm [3] or the Angle Statistics Reconstruction (ASR) algorithm [4], are inherently limited in their capabilities due to their underlying simplifications. On the other hand, more advanced statistical methods [5] are computationally expensive and often numerically unstable, requiring careful tuning of these methods.

In this contribution, a likelihood-based reconstruction method for muon scattering tomography is presented. The main idea of this algorithm is the direct minimization of the negative log-likelihood of the problem, using the automatic differentiation framework provided by the Python machine learning library PyTorch [6]. This workflow allows for the utilization of modern optimizing techniques provided by PyTorch, as well as an easy path to extend the algorithm in the future. After introducing and formulating the reconstruction task as a likelihood problem in Section 2, the implementation of the algorithm is detailed in Section 3, together with the presentation of first reconstruction results based on simulation data. Lastly, future prospects of the presented approach are outlined in Section 4.

2. Methodology

2.1 Reconstruction Task in Muon Tomography

When a muon traverses a medium, its direction is continuously altered due to multiple scattering. For a muon covering a distance H , multiple scattering can be defined via the variable $\Delta\theta$, describing the angular change between the initial direction \vec{d}_{in} and the final direction \vec{d}_{out} , and the variable Δx , describing the displacement of the muon. Both variables are visualized in Figure 1. Accurately describing the distributions of $\Delta\theta$ and Δx is complex. Therefore, a commonly used simplification is to describe them as a joint Gaussian distribution [7], which provides a good approximation of small scattering angles. In this case, the likelihood of a deflection $\Delta\theta$ after a distance H is given as [5, 7]

$$f(\Delta\theta) d(\Delta\theta) = \frac{1}{\sqrt{2\pi}\sigma_\theta} \exp\left(-\frac{\Delta\theta^2}{2\sigma_\theta^2}\right) d(\Delta\theta), \quad (1)$$

with

$$\sigma_{\Delta\theta}^2 = \lambda H (p_0/p)^2, \quad \lambda \equiv (15/p_0)^2 1/L_{\text{rad}}, \quad (2)$$

where p is the momentum of the muon, p_0 is a reference momentum, λ is the scattering density, and L_{rad} is the material-dependent radiation length. The variance of the Gaussian distribution describing Δx and its correlation with $\Delta\theta$ are given as [7]

$$\sigma_{\Delta x}^2 = \frac{H}{3} \sigma_\theta^2, \quad \rho_{\Delta\theta\Delta x} = \frac{\sqrt{3}}{2}. \quad (3)$$

This means that within this approximation, the traversed medium is only characterized by λ . Obtaining the distribution of λ within a medium, given the measured scattering of muons, is the reconstruction task at hand. For this purpose, the scanned medium is often discretized into voxels, where λ is assumed to be constant in each voxel. The obtained map of λ can either be used directly for material identification, or fed to subsequent analysis steps such as anomaly detection.

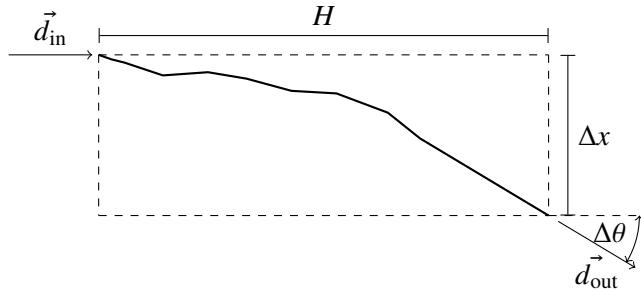


Figure 1: Visualization of the multiple scattering variables $\Delta\theta$ and Δx . Adapted from [8].

One of the simplest geometric approaches to perform this reconstruction task is the PoCA algorithm, which works under the assumption that the entire scattering along the muon path occurred at a single point [3]. This simplification restricts the capability of the approach to

reconstruct complex scenes, and it leads to an inefficient use of the limited available statistics since every muon contributes scattering information only to a single voxel. The ASR algorithm [4] tries to improve the latter by assuming that the scattering occurred in an extended region around the point of closest approach. While adding additional information to the reconstruction, this approach leads to a typical elongation of the reconstruction results along the direction of the ingoing muons. As an alternative to these geometric methods, statistical methods can be used: Based on the probability distributions of Δx and $\Delta\theta$ in (1), a likelihood \mathcal{L} is formulated. This likelihood, which will be introduced in Section 2.2, describes how well the voxel map of λ is capable of describing the observed scattering data. In [9], minimizing this likelihood directly by optimizing λ has been successfully attempted, however, the approach was found to be unsuitable for real-time application due to high computational demands. Subsequently, an approach to optimize the likelihood via an iterative expectation maximization algorithm has been developed first in [5].

2.2 Likelihood Formulation

The measurable observables in muon scattering tomography are the positions and directions of the muon when entering and when exiting a VOI (volume of interest). From this, the data vector $\mathbf{D}_i = (\Delta\theta_i, \Delta x_i)^T$ is defined for each muon i , where the definitions of $\Delta\theta_i$ and Δx_i are visualized in Figure 1. In analogy to the Gaussian multiple scattering distribution in (1), the likelihood of λ in a voxelized VOI for a single muon event is given as [5]

$$\mathcal{L}(\mathbf{D}_i|\lambda) = \frac{1}{2\pi\sqrt{|\Sigma_i|}} \exp\left(-\frac{1}{2}\mathbf{D}_i^T \Sigma_i^{-1} \mathbf{D}_i\right), \quad (4)$$

with the covariance matrix

$$\Sigma_i = (p_0/p_i)^2 \sum_j (\lambda_j \mathbf{W}_{ij}), \quad (5)$$

where the index j runs over all voxels that are hit by the muon path. It is important to note that the muon trajectory within the VOI is unknown, which means that assumptions about the real muon path based on the measured ingoing and outgoing positions need to be made. Two possible assumptions are the *straight line path*, where the known ingoing and outgoing positions are connected by a straight line, and the *PoCA path*, where the muon path is drawn from the ingoing muon position to the point of closest approach to the outgoing muon position. While p_0 is a fixed reference value, as defined in (2), p_i is the absolute muon momentum per event. For p_i , usually no or only rough estimations are available since a precise estimation on a per-event basis is currently not feasible. In many cases, p_i is therefore set to a value that does not necessarily represent reality. The weight matrix \mathbf{W}_{ij} contains the geometric information about the assumed voxel path, and is defined by [5]

$$\mathbf{W}_{ij} = \begin{bmatrix} L_{ij} & L_{ij}^2/2 + L_{ij}T_{ij} \\ L_{ij}^2/2 + L_{ij}T_{ij} & L_{ij}^3/3 + L_{ij}^2T_{ij} + L_{ij}T_{ij}^2 \end{bmatrix}, \quad (6)$$

where L_{ij} denotes the distance muon i covers in voxel j , and T_{ij} the distance of muon i from the end of voxel j until the end of the VOI. From the likelihood $\mathcal{L}(\mathbf{D}_i|\lambda)$ for a single muon, the negative

log likelihood for M muons is obtained via

$$\begin{aligned}
 -\log(\mathcal{L}) &= -\log \left(\prod_i^M \mathcal{L}(\mathbf{D}_{i,x}|\lambda) \cdot \mathcal{L}(\mathbf{D}_{i,y}|\lambda) \right) \\
 &= -\sum_i^M \left[\log \left(\frac{1}{2\pi\sqrt{|\Sigma_i|}} \exp \left(-\frac{1}{2} \mathbf{D}_{i,x}^T \Sigma_i^{-1} \mathbf{D}_{i,x} \right) \right) + \log \left(\frac{1}{2\pi\sqrt{|\Sigma_i|}} \exp \left(-\frac{1}{2} \mathbf{D}_{i,y}^T \Sigma_i^{-1} \mathbf{D}_{i,y} \right) \right) \right] \\
 &= \sum_i^M \left(\log(|\Sigma_i|) + \frac{1}{2} \mathbf{D}_{i,x}^T \Sigma_i^{-1} \mathbf{D}_{i,x} + \frac{1}{2} \mathbf{D}_{i,y}^T \Sigma_i^{-1} \mathbf{D}_{i,y} \right) + \text{const.}, \tag{7}
 \end{aligned}$$

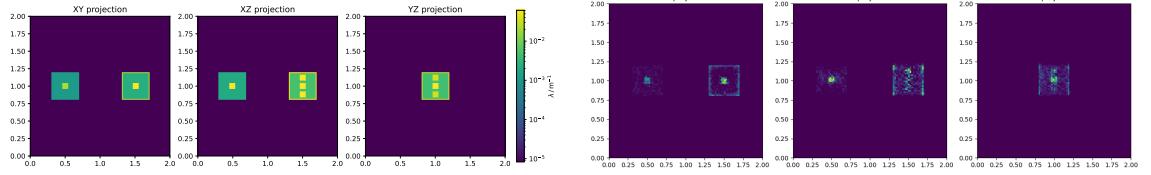
where $\mathbf{D}_{i,x}$, $\mathbf{D}_{i,y}$ describe the data vector projected onto the xz -plane, respectively the yz -plane.

3. Implementation of the Method and Evaluation on Simulation Data

Since the negative log likelihood in (7) quantifies how likely it is that the voxel map λ describes the observed muon data, minimizing $-\log(\mathcal{L})$ with respect to λ can be used as the reconstruction approach. For that, the calculation of $-\log(\mathcal{L})$ is implemented in Python within the PyTorch framework [6]. This allows for the usage of the `torch.autograd` functionality, which offers automatic differentiation: After calculating a result (here: $-\log(\mathcal{L})$), `torch.autograd` automatically collects the gradients from all numerical calculations with respect to a given set of parameters (here: λ), and combines them using the chain rule. The calculated gradient is used in combination with an optimizer (here using the `adam` algorithm [10]) over several epochs to find an optimal value for λ .

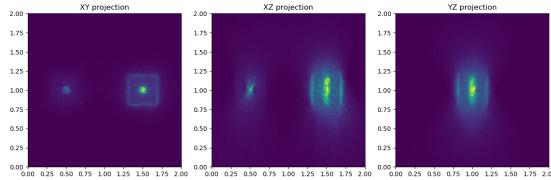
To validate this method, two muon tomography simulation datasets are created using realistic cosmic-ray muon distributions generated with the CRY library [11], muon propagation performed with the GEANT4 toolkit [12], and 3D scenes generated with the B2G4 framework for usage within GEANT4 [13]. The first scene, see Figure 2a, features a water cube and an iron box, each containing smaller tungsten cubes. The second scene, see Figure 3a, consists of three randomly distributed solid objects. These scenes are chosen since they represent scenarios where imaging is challenging, involving objects that block or enclose each other. For each scene, 1.5×10^7 muons are injected on a $10\text{ m} \times 10\text{ m}$ plane, corresponding to a measurement time of 15 min. The detectors have a dimension of $2\text{ m} \times 2\text{ m}$ with a detector distance of 1 m for the first scene, respectively $1\text{ m} \times 1\text{ m}$ and 0.65 m for the second scene, where a perfect detector resolution is assumed.

The reconstruction results using the PoCA algorithm [3], the ASR algorithm [4], and the gradient descent reconstruction presented in this work are shown in Figure 2 and Figure 3. For the gradient descent approach, the algorithm is stopped after 25 epochs, since no significant improvement in the reconstruction result is visible afterward. As an initialization, all voxels are initialized with $\lambda = \lambda_{\text{air}}$, which means that no prior information is introduced. Furthermore, the algorithm is performed in a mini-batch gradient descent approach: This means that for each epoch, the dataset is split randomly into M_{batch} batches (here: $M_{\text{batch}} = 15$, corresponding to 1×10^6 events per batch), and the optimization step is performed separately for each batch. The path assumption made for the voxel tracing is the *PoCA path* approximation, as explained in Section 2.2. The momentum is set to $p = 750\text{ MeV}$ in (5) for all muons, assuming that no momentum estimation is available, with a reference momentum of $p_0 = 3000\text{ MeV}$. These numerical values are chosen via a simple hyperparameter search since they provide the best convergence behavior.

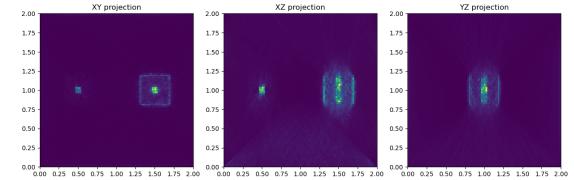


(a) Ground truth of the scene. The left object is a water cube, the right object is a hollow iron box. The small cubes within are made out of tungsten.

(b) Reconstruction result of the scene using the PoCA algorithm [3].

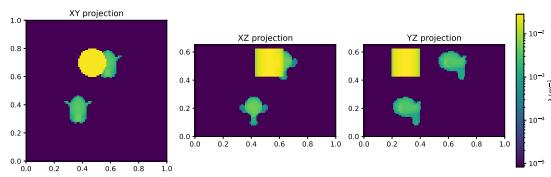


(c) Reconstruction result of the scene using the ASR algorithm [4].

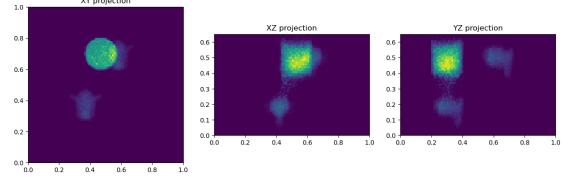


(d) Reconstruction result of the scene using the gradient descent method presented in this work.

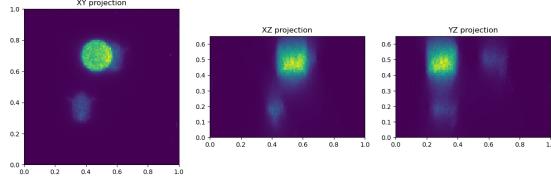
Figure 2: Evaluation of different reconstruction algorithms based on simulation data for the first scene.



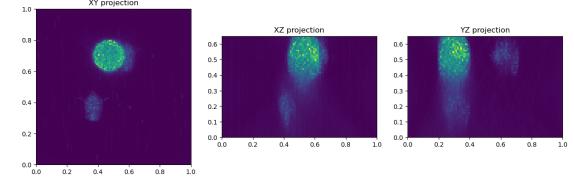
(a) Ground truth of the scene. The cylinder is made of iron, while the two sculptures are made of CaCO_3 .



(b) Reconstruction result of the scene using the PoCA algorithm [3].



(c) Reconstruction result of the scene using the ASR algorithm [4].



(d) Reconstruction result of the scene using the gradient descent method presented in this work.

Figure 3: Evaluation of different reconstruction algorithms based on simulation data for the second scene.

Visually, the results show that the xy projection is well reconstructed with all algorithms in both scenes. For the xz and yz projections, the general shapes are clearly visible, although no algorithm is able to clearly distinguish between the individual blocks in the first scene. In addition, the ASR algorithm contains a visible smearing along the z direction, which is characteristic for this approach, while the PoCA algorithm is unable to fully reconstruct the small boxes within the first scene. A quantitative analysis of the similarity between the reconstruction results and the ground truths is given in Table 1. The applied metrics are the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index Measure (SSIM) [14], and the Learned Perceptual Image Patch Similarity (LPIPS) [15]. Note that for the PSNR and SSIM, higher values correspond to a higher similarity, while lower values correspond to a higher similarity for the LPIPS. For the first scene,

Table 1: Numerical evaluation of the reconstruction results using different metrics. The best result for each scene and each reconstruction is highlighted in bold.

	Scene 1			Scene 2		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
PoCA	22.324	0.870	0.477	19.666	0.737	0.378
ASR	22.321	0.829	0.477	19.699	0.685	0.395
Gradient descent	22.348	0.863	0.480	19.960	0.838	0.326

Table 2: Computational benchmarking of the algorithms for the first scene. Note that for the gradient descent, convergence and runtime of the algorithm depend on the exact problem and the chosen hyperparameters.

	Memory consumption	Execution time
PoCA	≈400 MB	≈1.4 s
ASR	≈2300 MB	≈14.5 s
Gradient descent	≈3400 MB	≈21.7 s

the quantitative results are very similar, with a different best-performing algorithm for each metric. For the second scene, the gradient descent algorithm performs the best, although again the margins are very small, especially for the PSNR. This result shows that the gradient descent algorithm is already capable of performing at least as good as existing standard algorithms without extensive optimization, and without the usage of its further potentials which will be outlined in Section 4.

A benchmarking of the memory consumption and the execution times of the algorithms is presented in Table 2. Both the memory consumption and the execution time are the smallest for the PoCA algorithm due to its simplicity. In the use case presented here, the benchmark results for the ASR algorithm and the gradient descent algorithm are comparable. Note that the memory consumption of the gradient descent algorithm can be steered via the batch size M_{batch} , since only the relevant data from each batch needs to be stored at a time. This makes the gradient descent method scalable in terms of memory consumption. For the scenes at hand, batch sizes between $M_{\text{batch}} = 1$ and $M_{\text{batch}} = 500$ all produced similar reconstruction results, validating that the approach is also viable for smaller batches where the statistical variance for each batch is larger.

4. Further Prospects and Conclusions

In this work, a likelihood-based gradient descent reconstruction method for muon scattering tomography has been presented. After formally introducing the likelihood, the algorithm has been implemented in PyTorch, making use of the `torch.autograd` automatic differentiation functionality to perform the optimization. Using two created simulation datasets, it has been shown that the reconstruction results are already qualitatively and quantitatively able to compete and even surpass results obtained with traditional reconstructions methods such as the PoCA and ASR algorithms. The required computational resources are comparable to those required by the ASR method, with scalability possible due to the mini-batch gradient descent approach.

The greatest potential of the presented method lie within the improvements to the algorithm that are intrinsically possible due to its likelihood formulation. This allows for the introduction of penalty terms $p(\lambda)$, with $-\log(\mathcal{L}) \rightarrow -\log(\mathcal{L}) + p(\lambda)$. With that, prior information can be introduced to the problem, for example knowledge about expected materials or geometric shapes. As introduced in Section 2.1, the multiple scattering distribution is described as Gaussian. While this approximation is valid for small scattering angles, it is unable to correctly incorporate the large scattering angles caused by the tails of the real scattering distribution. This issue can be encountered by formulating the likelihood in (4) as a Gaussian scale mixture model [16],

$$\mathcal{L}_{\text{GSM}} = \sum_k w_k g(\Sigma_{i,k}) \quad \text{with} \quad g(\Sigma_{i,k}) \propto \exp\left(-\frac{1}{2} \mathbf{D}_i^T \frac{1}{s_k} \Sigma_{i,0}^{-1} \mathbf{D}_i\right),$$

where the parameters w_k and s_k can either be set to appropriate values, or treated as free parameters during optimization.

References

- [1] C. Grupen, *Astroparticle physics; Second edition*, Undergraduate texts in physics, Springer (2020).
- [2] S. Barnes, A. Georgadze, A. Giannanco, M. Kiisk, V.A. Kudryavtsev, M. Lagrange et al., *Cosmic-Ray Tomography for Border Security*, *Instruments* **7** (2023) .
- [3] W.C. Priedhorsky, K.N. Borozdin, G.E. Hogan, C. Morris, A. Saunders, L.J. Schultz et al., *Detection of high-Z objects using multiple scattering of cosmic ray muons*, *Review of Scientific Instruments* **74** (2003) 4294.
- [4] M. Stapleton, J. Burns, S. Quillin and C. Steer, *Angle Statistics Reconstruction: a robust reconstruction algorithm for Muon Scattering Tomography*, *Journal of Instrumentation* **9** (2014) P11019.
- [5] L.J. Schultz, G.S. Blanpied, K.N. Borozdin, A.M. Fraser, N.W. Hengartner, A.V. Klimenko et al., *Statistical Reconstruction for Cosmic Ray Muon Tomography*, *IEEE Transactions on Image Processing* **16** (2007) 1985.
- [6] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan et al., *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, 2019.
- [7] PARTICLE DATA GROUP collaboration, *Review of Particle Physics*, *Prog. Theor. Exp. Phys.* **2022** (2022) 083C01.
- [8] J. Alameddine, *Enabling Next-generation Particle Cascade Simulations*, Ph.D. thesis, TU Dortmund University, 2024.
- [9] L.J. Schultz, *Cosmic ray muon radiography*, Ph.D. thesis, Portland State University, 2003.
- [10] D.P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, *3rd International Conference on Learning Representations (ICLR)* (2015) .
- [11] C. Hagmann, D. Lange and D. Wright, *Cosmic-ray shower generator (CRY) for Monte Carlo transport codes*, *2007 IEEE Nuclear Science Symposium Conference Record* **2** (2007) 1143.
- [12] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce et al., *Geant4—a simulation toolkit*, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **506** (2003) 250.
- [13] Á. Bueno Rodríguez, F. Sattler, M. Pérez Prada, M. Stephan and S. Barnes, *B2G4: Integrating 3D Blender models in Geant4 simulations for synthetic muography data generation*, *Muographers2023 - International workshop on muography* (2023) .
- [14] Z. Wang, A. Bovik, H. Sheikh and E. Simoncelli, *Image quality assessment: from error visibility to structural similarity*, *IEEE Transactions on Image Processing* **13** (2004) 600.
- [15] R. Zhang, P. Isola, A.A. Efros, E. Shechtman and O. Wang, *The Unreasonable Effectiveness of Deep Features as a Perceptual Metric*, *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018) 586.
- [16] G. Wang, L.J. Schultz and J. Qi, *Statistical Image Reconstruction for Muon Tomography Using a Gaussian Scale Mixture Model*, *IEEE Transactions on Nuclear Science* **56** (2009) 2480.