CoReSi: a GPU-based software for Compton camera reconstruction and simulation in collimator-free SPECT

Vincent Lequertier, Etienne Testa, Voichita Maxim

CoReSi: a GPU-based software for Compton camera reconstruction and simulation in collimator-free SPECT

Vincent Lequertier[1], Étienne Testa[2], Voichiţa Maxim[1]

[1] INSA-Lyon, Université Claude Bernard Lyon 1, CNRS, Inserm, CREATIS UMR 5220, U1294, F-69100, LYON, France
[2] University of Lyon, Université Claude Bernard Lyon 1, CNRS/IN2P3, IP2I Lyon, F-69622, Villeurbanne, France

E-mail: voichita.maxim@creatis.insa-lyon.fr

20 December 2024

Abstract.
Objective: Compton cameras are imaging devices that may improve observation of sources of γ photons. The images are obtained by solving a difficult inverse problem. We present CoReSi, a Compton Reconstruction and Simulation software implemented in Python and powered by PyTorch to leverage multi-threading and for easy interfacing with image processing and deep learning algorithms. The code is mainly dedicated to medical imaging and for near-field experiments where the images are reconstructed in 3D.
Approach: The code was developed over several years in C++, with the initial version being proprietary. We have since redesigned and translated it into Python, adding new features to improve its adaptability and performances. This paper reviews the literature on Compton camera mathematical models, explains the implementation strategies we have adopted and presents the features of CoReSi.
Main results: The code includes state-of-the-art mathematical models from the literature, from the simplest, which allow limited knowledge of the sources, to more sophisticated ones with a finer description of the physics involved. It offers flexibility in defining the geometry of the Compton camera and the detector materials. Several identical cameras can be considered at arbitrary positions in space. The main functions of the code are dedicated to the computation of the system matrix, leading to the forward and backward projector operators. These are the cornerstones of any image reconstruction algorithm. A simplified Monte Carlo data simulation function is provided to facilitate code development and fast prototyping.
Significance: As far as we know, there is no open source code for Compton camera reconstruction, except for MEGAlib, which is mainly dedicated to astronomy applications. This code aims to facilitate research as more and more teams from different communities such as applied mathematics, electrical engineering, physics, medical physics get involved in Compton camera studies. Implementation with PyTorch will also facilitate interfacing with deep learning algorithms.

1. Introduction

Interest in Compton camera (CC) has been driven in recent years by technological advances in γ-ray detection and the emergence of new applications. The number of teams involved in CC-related research and development activities is growing steadily.

Initially used for astronomy, which was the catalyst for progress in the first decades with the Comptel telescope as the first achievement, homeland security has recently taken over as the driving force for CC research. It is widely recognized that the compact size, portability, and ability of CCs to image a broad range of energies make them an attractive option for nuclear and waste monitoring. However, medical applications, particularly for monitoring proton and targeted radionuclide therapies, present greater challenges. Among these, alpha therapies have gained increasing interest in clinical trials due to their high lethality against cancer cells and minimal damage to healthy tissues (Seo 2019). Several studies have focused on ^{225}Ac, which decays into alpha-emitting daughters such as ^{221}Fr (11.6% emission probability) and ^{213}Bi (26.1% emission probability), subsequently emitting 218 keV and 440 keV photons detectable by γ cameras. Collimated cameras currently employed in Single Photon Emission Computed Tomography (SPECT) generally suffer from low spatial resolution and sensitivity and their collimators become increasingly transparent at these energy levels.

CCs have three main advantages over regular SPECT with collimated cameras: large gain in detection efficiency, portability, and the ability to detect higher energy γ rays ($>\mathbf{300\,keV}$), which is particularly useful for the aforementioned medical applications (Takeda et al. 2015, Fontana et al. 2017). CC could therefore reduce the injected dose for imaging and increase the throughput of imaging systems. It could also reduce examination time for patients and improve the workload of healthcare professionals, thus potentially improving the quality of care. Detailed reviews of CC systems and applications can be found elsewhere, for example in (Parajuli et al. 2022).

The forward model describing CC data is related to integrals on conical shapes and to conical Radon transforms. Specific analytical reconstruction algorithms have been developed for their inversion, but they remain limited to relatively ideal data, with low noise, negligible measurement uncertainties and generally no attenuation of the γ-ray on its way from the source to the detector (Cree & Bones 1994, Moon & Haltmeier 2017). Although these algorithms can be fast (Maxim 2014), they suffer from the blurring induced by binning the data into sinogram-like tables and also from unfulfilled Tuy completeness conditions related to constraints in the acquisition geometry (Tuy 1983, Smith 2005, Terzioglu et al. 2018). Iterative algorithms can operate in list mode, avoiding data binning, and are also less affected by data incompleteness. More importantly, they specifically account for the statistical data distribution via the maximum likelihood formulation and for the number of sampling uncertainties via the direct model (Wilderman, Clinthorne, Fessler & Rogers 1998, Muñoz, Barrientos, Bernabéu, Borja-Lloret, Llosá, Ros, Roser & Oliver 2020). CC data ($\boldsymbol{y}$) are realizations of a Poisson distributed random variable,

$$\boldsymbol{y} \sim \mathscr{P}(\boldsymbol{T}\boldsymbol{\lambda}), \tag{1}$$

where $\boldsymbol{\lambda}$ is a vector containing the mean emissions in the voxels of the volume and $\boldsymbol{T}$ is the matrix of the system. When the source produces γ-rays of different energies, as in proton and alpha therapy, spectral reconstructions are more accurate. The spectral model takes into account incomplete absorption probabilities and reconstructs the images at all energies simultaneously (Muñoz et al. 2021). In this case the vector $\boldsymbol{\lambda}$ contains as many volumes as discrete values or bins of energy. The quality of the reconstructed images depends heavily on the quality of the data and the accuracy of the models. Careful selection of events that are not strongly affected by measurement errors is key. Selection strategies have been described for example in (Draeger

et al. 2017, Watanabe et al. 2014, Muñoz et al. 2021). A large proportion of the data is discarded in this step. Accurate models allow not only to retain events that would otherwise be discarded, but also to compensate for misclassification.

Once the mathematical model has been established, the next step is to translate it into an algorithm. Line-based projection and backprojection operators can be found in a number of software packages and could be used to implement projections through the generatrices of the cone. However, due to the sparse nature of CC data, which is distributed across a space of at least five dimensions, ray tracing algorithms, effective in other tomography modalities, are inadequate for Compton imaging. The computing demands are significant, requiring carefully designed implementation strategies that fully utilize the multi-threading capabilities of modern GPUs and libraries. For all these reasons, producing a code that adapts to different geometries, detector materials and levels of confidence in the measurements is a challenging task. To our knowledge, there is no freely available source code for CC image reconstruction, except for the Mimrec library from the MEGAlib package which was developed for astrophysics (Zoglauer et al. 2011). Yet open source facilitates research, reproducibility and comparisons in classical tomographic modalities. Codes as as STIR (Thielemans et al. 2012), RTK (Rit et al. 2014), ASTRA (van Aarle et al. 2015) and CASToR (Merlin et al. 2018), developed in C++ with CUDA accelerators, implement data processing modules, projection and backprojection operators and some regularized reconstruction methods. More recently, Python libraries have been released for classical tomographic modalities. They allow model-based (iterative) and data-based (deep learning) algorithms to be combined in end-to-end image reconstruction pipelines. These include ODL (https://github.com/odlgroup/odl), DeepInverse (Tachella et al. 2023) and PyTomography (Polson et al. 2024), the latter specifically designed for emission tomography. However, these software packages are not adapted to Compton camera and do not currently include modules for it.

The aim of this paper is to describe the implementation of a Compton reconstruction tool using PyTorch (Paszke et al. 2019) for a fast, parallelized reconstruction tailored for medical applications and running on either CPU or GPU, using a model and data-centric approach and paving the way towards deep learning-enhanced imaging. This tool is called CoReSi, for Compton Reconstruction and Simulation, and its source code is available at https://github.com/CoReSi-SPECT/coresi. The different configurations and features of the Python code were tested on MC simulated data and validated against the previous CoReSi version, written in C++ and used in (Maxim et al. 2016, Hilaire et al. 2016, Feng et al. 2019, Feng et al. 2021, Muñoz et al. 2022). Quantitative studies with 2D and 3D reconstructions in nuclear medicine and proton-therapy scenarii can be found therein.

This paper is organised as follows. Section 2 describes Compton imaging, the mathematical models of acquisition, the system matrix and the sensitivity used in the reconstruction algorithms. The list-mode MLEM algorithm is also recalled. Section 3 presents the structure of the CoReSi code and the algorithmic choices for implementing image reconstruction and data simulation. Section 4 gives details of the MC simulation carried out to produce data on which some of the capabilities of the code are illustrated. Finally, Sections 6 and 7 are devoted to discussions and conclusions.

2. SPECT imaging with a Compton camera

A γ-ray is recorded as a Compton event if it interacts with the camera at least twice. Under ideal imaging conditions, each event has a virtual Compton cone where the source of the photon would be located. In real-world imaging, due to the measurement uncertainties, the point of emission of the photon belongs to a thick conical envelope. This section presents the mathematical model of the acquisition process, which is also the direct model of the reconstruction problem. The list-mode MLEM image reconstruction algorithm is recalled in its monoenergetic and spectral versions.

2.1. The imaging device

CCs typically consist of a scatterer and an absorber, though alternatives such as monoblock detectors (e.g., the commercially available H3D camera), liquid detectors (Aprile et al. 2020), and gaseous detectors (Azevedo et al. 2013) also exist. The choice of detector materials is aimed at optimizing Compton scattering in the scatterer block and achieving full photon absorption in the absorber block. Full absorption is particularly crucial when the initial energy E_0 of the photon is unknown. Each detector block may be composed of multiple layers. A camera with three scatterer layers (a,b, and c) and three absorber layers (1,3, and 4) is illustrated in Figure 1.

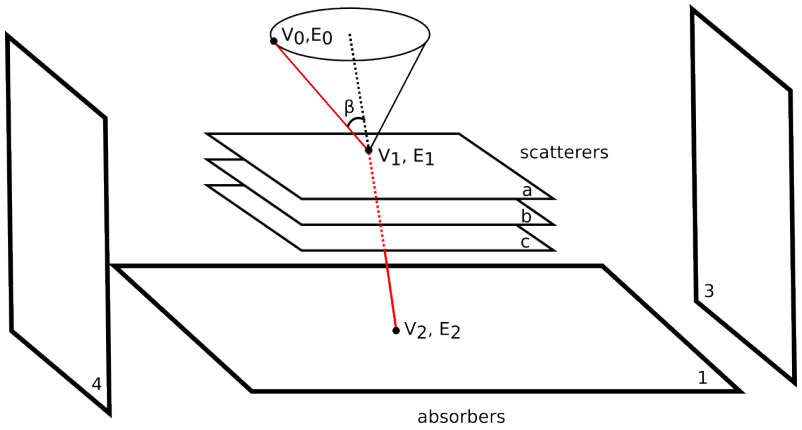


Figure 1: A CC consisting of a scatterer and an absorber. The layers are marked with the letters a, b, c for the scatterer and with the numbers 1, 3 and 4 for the absorber. The latter follow the numbering convention used in CoReSi. Absorber layers 2 and 5 could be added in front of and behind the scatterer. An event $e = (V_1, E_1, V_2, E_2)$ is measured when at least two coincident hits are detected.

2.2. Events

Compton imaging is based on the detection of Compton interactions undergone by the γ ray emitted by the source. A photon emitted with initial energy E_0 should hit the detector at least twice to be recorded. The first hit must be a Compton scattering, resulting in an energy transfer from the γ ray to the electron ejected from the atom. The second hit allows the trajectory of the scattered photon to be measured. Events

with more than two hits are possible. Since determining the order of the hits in the sequence is beyond the time resolution of the detector, the order is estimated using the energies and geometric considerations. In the case of two-hit events, the first hit is generally assumed to be the one in the scatterer or the one with the lowest deposited energy. In the rest of this paper only two-hit events are considered, ordered in a pre-processing step.

For each hit, the position V and the deposited energy E are measured (see Figure 1). An event $e = (V_1, E_1, V_2, E_2)$ is recorded, containing the measurements (V_1, E_1) associated with the first hit, the position V_2 of the second hit and E_2, the total deposited energy in the other hits. The scattering angle β is given by the Compton scattering formula:

$$\cos\beta = 1 - \frac{m_e c^2 E_1}{E_0(E_0 - E_1)}, \tag{2}$$

where m_e is the mass of the electron, $m_e c^2 = 511\,\mathrm{keV}$ and E_0 is either known or estimated as the sum $E_{tot} = E_1 + E_2$ of energies deposited in the coincident hits. The Compton cone is the circular cone with the axis $\overrightarrow{V_2V_1}$ and the half-opening angle β. Events $e = (V_1, E_1, V_2, E_2)$ are recorded in list mode because they lie in a high-dimensional space that is difficult to sample efficiently in sinogram-like tables.

2.3. System matrix computation

In list mode, the system matrix (SM) T contains one row per detected event e_i and its elements are $t_{ij} = \mathbb{P}(e_i|j)$, *i.e.*, the probability that e_i was produced by a photon emitted in voxel j, $j = 1, \ldots, N_v$. A line of the SM has the same size as the volume, *i.e.,* has N_v elements for a volume with N_v voxels. If N_e events were kept for image reconstruction, the size of T is $N_e \times N_v$.

Depending on the fidelity of the MC simulation or the characteristics of the real detector, different models and calculation strategies can be used. These choices have a much greater impact on the result than for imaging modalities based on line-integral transforms.

Each element t_{ij} of the system matrix is a product of factors divided into two categories, which will be described in the following subsections. The first contains geometrical factors that take into account the position of the voxel j with respect to the cone defined by the event i. They include measurement uncertainties, modelled with a probabilistic distribution that is maximum on the cone and decreases as we move away from it. The second includes probabilities of interaction or escape, which are a function of the material and shape of the layers and the energy of the photon.

The mono-energy system matrix is calculated with the equation (6) from (Hilaire et al. 2016). For a given vector $\overrightarrow{w}$, we denote $\theta_{\overrightarrow{w}}$ the angle between $\overrightarrow{w}$ and the normal to the camera. Let O_j be the center of the voxel v_j and $K(\delta_{O_j}, E_0)$ the Compton scattering cross section with parameters calculated from the data. The function h_i models the uncertainties on the measurements from the i-th event from the list, β_i is the Compton angle and δ_{O_j} is the angle between the vectors $\overrightarrow{V_1O_j}$ and $\overrightarrow{V_2V_1}$. With these notations, equation (6) from (Hilaire et al. 2016) is:

$$t_{ij} = K(\delta_{O_j}, E_0) \frac{|\cos(\theta_{\overrightarrow{V_2V_1}})|}{||\overrightarrow{V_2V_1}||_2} \frac{|\cos(\theta_{\overrightarrow{V_1O_j}})|}{||\overrightarrow{V_1O_j}||_2^2} h_i(O_j), \text{ if } |\delta_{O_j} - \beta_i| \leq 3\sigma_{\beta_i}, \tag{3}$$

where h_i models the uncertainties and is taken as a Gaussian function with zero mean and standard deviation σ_{β_i}. Above some threshold, taken in (3) as equal to $3\sigma_{\beta_i}$, $t_{ij} = 0$.

For spectral reconstruction, each event is associated to several cones, one for each possible known initial energy E_0^m, $m = 1, \ldots, M$. Each line of the SM has the size of M side-by-side volumes, thus $M \times N_v$ elements reshaped in a 1D vector. For the spectral version, equation (4) from (Muñoz et al. 2022) has been implemented. An element $t_{imj} = \mathbb{P}\left(e_i | E_0^m, j\right)$ of the SM is computed as:

$$t_{imj} = \frac{|\cos\left(\theta_{\overrightarrow{V_1 O_j}}\right)|}{||\overrightarrow{V_1 O_j}||_2^2} e^{-\mu_0 \lambda_0} \frac{K(\delta_{O_j}, E_0^m)}{(E_0^m - E_1)^2} e^{-\mu_1 \lambda_1} \sigma_{E_2}(E_2 | E_0^m - E_1) h_{i,m}(O_j), \quad (4)$$

where μ_k and λ_k are the linear attenuation coefficient and distance inside the detectors traversed by the incident(0) and scattered(1) photons, and $\sigma_{E_2}(E_2 | E_0^m - E_1)$ is the cross section of the interaction in the absorber. It has three possible values, according to the possible physical interactions in the absorber, namely photoelectric, Compton or pair creation. The function $h_{i,m}$ models the uncertainties for event i, supposing the initial energy is E_0^m.

2.3.1. Geometrical modelling For ideal events, the non-zero elements of the system matrix correspond to voxels intersecting the surface of the Compton cone. They can be computed by considering conic sections in 2D (Wilderman, Rogers, Knoll & Engdahl 1998), uniformly (Kim et al. 2007) or non-uniformly sampled generatrices on the cone (Lojacono et al. 2011), or the distance from the centre of the voxel to the surface of the Compton cone. The latter is the choice made for CoReSi. We implemented this model by giving the Compton cone what we call a parallel thickness (see figure 2 (a)), which consists in giving a strictly positive value to all voxels whose orthogonal distance to the cone surface is less than a threshold. Setting the threshold to a small value, e.g. the diagonal of a voxel, allows to mimic ideal events. The values can be set to decrease with increasing distance to the apex of the cone and to the surface of the Compton cone (Maxim et al. 2016).

For non-ideal events, the energy uncertainties translate into angular uncertainties (Ordonez et al. 1997). In CoReSi they are included in the geometry-related factors and are implemented under the angular thickness model introduced in (Hilaire et al. 2016). The values in the SM row decrease with increasing angular distance δ_j from the voxel j to the cone defined by the measurement (see Figure 2 (b)). A probabilistic distribution is chosen to represent the angular resolution measure. This distribution can be a Lorenzian, a von Mises (Sauve et al. 1999), a Gaussian (Maxim et al. 2016) or a mixture of Gaussians (Feng et al. 2021). Hit position uncertainties are not currently implemented in the code. They introduce an error in the axis direction which adds up to the much more critical angular uncertainties. We use parallel GPU-accelerated PyTorch vectorized processing to implement the distance to the cone simultaneously for all voxels of the volume.

2.3.2. Physical modelling Photons can interact through various processes, namely photo-electric effect, Compton scattering and pair creation for photon energies larger than 1.022 MeV. The probabilities of their occurrence can be considered in the system matrix. They depend on the event and, to a lesser extent, on the voxels of the volume. For example, they take into account the probability of the photon reaching

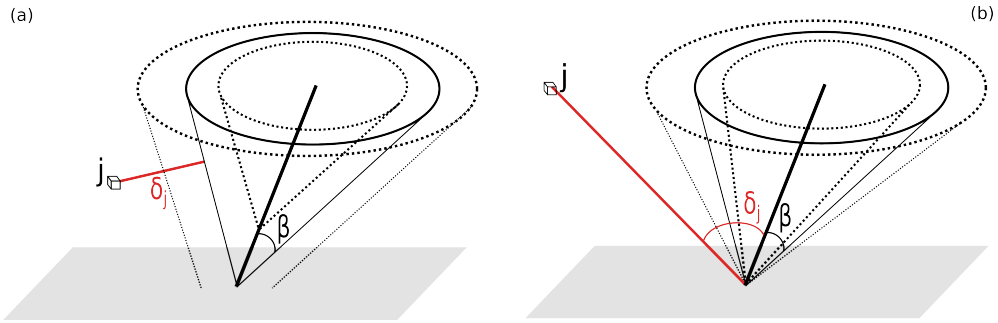


Figure 2: An ideal event is modelled with a parallel thickness of the Compton cone (a). Measurement uncertainties lead to uncertainties on the Compton angle and are modelled with an angular thickness (b).

the scatterer and then the absorber, and the probability of the scattered photon to go through some scatterer layers without any interaction. For non-spectral reconstruction and negligible uncertainties, they represent a factor that multiplies the row of the system matrix. These factors can therefore cancel out during image reconstruction and their calculation becomes unnecessary in these cases. In the spectral mode, differential cross sections and probabilities of all possible initial energies E_0^m, $m = 1, \ldots, M$ are taken into account.

2.4. Sensitivity computation

The role of the sensitivity vector $\boldsymbol{s} = (s_1, \ldots, s_{N_v})$ is to compensate for variations in the probability of detection of γ particles emitted through the volume. Its element s_j is the probability of detection for a photon emitted in voxel j. For poly-energetic sources, the sensitivity vector has size $M \times N_v$ and its elements are denoted s_{mj} hereafter, although $\boldsymbol{s}$ is still a one-dimensional vector. The element s_{mj} is the probability of a photon emitted with energy E_0^m at voxel j being detected.

For a system matrix containing a row for each measurable event, the sensitivity corresponds to the backprojection of a vector of ones, $\boldsymbol{s} = \boldsymbol{T}^T 1$, and is equal to the sum of the rows of the system matrix. Since in list mode the rows of $\boldsymbol{T}$ are restricted to observed events, the identity $\boldsymbol{s} = \boldsymbol{T}^T 1$ no longer holds.

The best estimate of $\boldsymbol{s}$ can be obtained by performing a large MC simulation of a uniform source covering the entire volume and recording for each voxel the number of photons detected by the camera. This simulation is relatively easy to perform and the size of the resulting vector is the size of the volume. However, for prototyping purposes, it is useful to compute the sensitivity within the reconstruction algorithm.

As for the system matrix, the sensitivity values depend primarily on geometrical factors such as the dimensions of the camera and the position of each voxel with respect to it. Secondarily, they depend on physical properties such as the materials making up the detectors and the energy of the γ rays. Following this classification and the increasing complexity of the model, CoReSi supports three sensitivity calculation modes: (i) central layer solid angle (CLSA), (ii) multi-layer solid angle (MLSA) and (iii) like system matrix (SM-like). The first is based on the model from (Maxim et al. 2016). It consists in computing at each voxel of the volume, the approximated

solid angle of a virtual layer placed at half-height of the scatterer. This model is fast to compute and adapted to small-sized cameras. The second considers all the scatterer layers individually and has a more precise implementation of the solid angle, better suited to large scatterers ((Feng et al. 2021), equation (15)). Both neglect the importance of the second interaction, implying that the probability of producing a Compton event is the same for all scattered photons. The first two calculation methods are analytical, while the third is based on MC integration. Derived from the model in (Muñoz et al. 2018), the last model was modified to account for measurement uncertainties. Additional differences in implementation will be presented in section 3.2.

2.5. Maximum-likelihood expectation maximization for image reconstruction

Instantiating the expectation maximization (EM) algorithm (Dempster et al. 1977) for the maximum likelihood (ML) of Poisson-distributed data in emission tomography gives the MLEM algorithm (Shepp & Vardi 1982). Its list-mode (LM) version corresponds to the limit case where there is at most one event in each virtual bin of data (Wilderman, Clinthorne, Fessler & Rogers 1998). The estimation of the mean emissions per voxel volume is computed iteratively by the formula:

$$\hat{\lambda}_j^{k+1} = \frac{\hat{\lambda}_j^k}{s_j} \sum_i \frac{t_{ij}}{\sum_\ell t_{i\ell} \hat{\lambda}_\ell^k}, \tag{5}$$

starting from a vector $\hat{\lambda}^0$ with strictly positive entries. In equation (5), j is the voxel index, i the event index. The factor $\sum_\ell t_{i\ell} \hat{\lambda}_\ell^k$ is the projection of the current volume following the cone of the i^{th} event. The MLEM algorithm consists in multiplying the current solution $\hat{\lambda}_j^k$ with the backprojection of the ratio between the number of events measured per bin (which is one in list mode) and their estimated values $\sum_\ell t_{i\ell} \hat{\lambda}_\ell^k$ and weighting with the sensitivity.

Iterations in equation (5) are best suited for monoenergetic sources. For polyenergetic sources, spectral reconstruction algorithms have been proposed (Xu & He 2005, Muñoz, Barrientos, Bernabéu, Borja-Lloret, Llosá, Ros, Roser & Oliver 2020). The iterations of the spectral LM-MLEM algorithm are:

$$\hat{\lambda}_{mj}^{k+1} = \frac{\hat{\lambda}_{mj}^k}{s_{mj}} \sum_i \frac{t_{imj}}{\sum_m \sum_\ell t_{im\ell} \hat{\lambda}_{m\ell}^k}. \tag{6}$$

Instead of a single volume, M volumes are reconstructed simultaneously, one for each emission energy.

2.6. Missing data, noise and regularization

The prototypes currently used are small compared to the size of the observed objects, so projections on a large proportion of the cones could not be measured. This lack of data cannot be compensated for by the sensitivity factor. Data truncation leads to typical elongation artefacts due to the ill-posed nature of the reconstruction problem, which violates Tuy's conditions. The artefacts are further exacerbated by the presence of statistical noise and measurement uncertainties. One solution is to use multiple cameras or a moving camera, which requires either more expensive hardware or longer acquisition times. Software solutions consist in using a priori knowledge. This is

incorporated into the algorithms through variational formulations as denoising steps. The total variation regularization convergent algorithm from (Feng et al. 2021, Maxim et al. 2023) has been implemented in the CoReSi code. It is likely that in the near future, deep learning methods will provide data-based priors or post-processing methods capable of making the difference to the current state of the art.

3. The CoReSi code

This section is a description of the structure of the CoReSi code and gives some implementation details.

3.1. Classes composing CoReSi

Table 1 lists the classes that make up CoReSi with a short description. Each class has attributes representing either data or functions to manipulate it.

Table 1: Classes composing CoReSi.

Name	Description
`Camera`	Position, orientation, a list of layers for the scatterer and a list of layers for the absorber of a CC
`Layer`	Dimensions, material, ... of a layer of a scatterer or an absorber
`Material`	Hold physics constants for materials (e.g. Silicon)
`Event`	Measured energies and positions for a Compton event
`Model`	Algorithms for the computation of the system matrix and of the sensitivity
`MLEM`	Parameters and functions for the LM-MLEM reconstruction algorithm
`Image`	PyTorch array with its dimensions in centimeters and voxels along with methods
`Points`	Coordinates of a point in 3D, with overloaded mathematical operators

Figure 3 is a simplified UML diagram of CoReSi that shows the main attributes and methods of each class. The `Model` class serves as the cornerstone of the CoReSi source code, encapsulating the implementation of functions for computing the system matrix rows and the sensitivity. The list-mode MLEM algorithm is currently implemented for image reconstruction via a `run()` function that loops over the iterations.

3.1.1. The `Camera` and `Layer` classes The `Camera` class held information about the cameras, such as their positions, number of layers and material composition. The `Layer` class is used to describe the layers, such as their dimension, thickness and centre.

A generic camera has an arbitrary number of parallel scatterer layers. Up to 5 absorber layers are also supported, forming an open box around the scatterer. A camera with three scatterer layers and absorber layers 1, 3 and 4 is shown in Figure 1. Absorbers 2 and 4 can be added to complete the box laterally. Layers in a detector

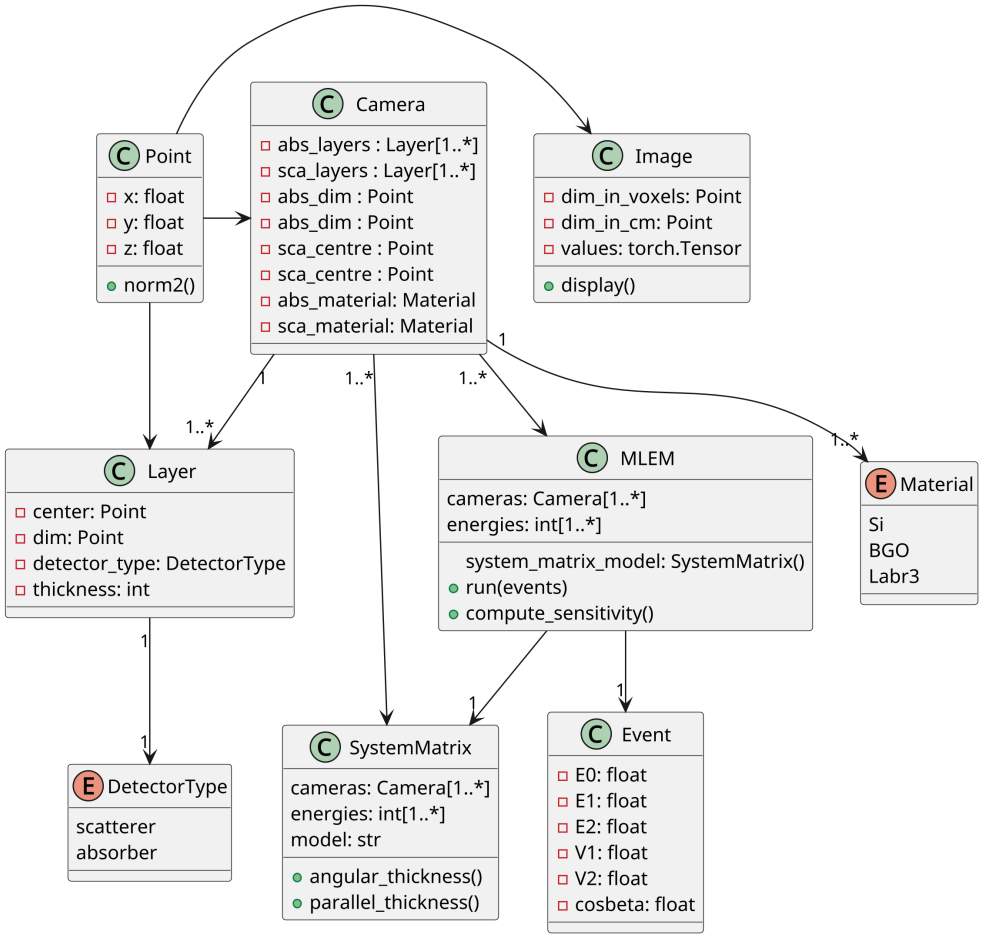


Figure 3: Simplified UML diagram of the CoReSi software, with the Classes (C) and Enums (E) composing it.

need not have identical relative positions and dimensions, but should be of the same material. CoReSi assumes the relative position of the absorber layers according to their order in the configuration file. This allows to simplify the description of the geometry of the camera, which is further used in the `Event` class. Missing absorber layers with respect to this convention can be disabled.

Once this generic camera is defined, an arbitrary number of copies can be placed around the volume to be imaged. The position of each copy is defined by giving the local coordinate system of the camera. The Oz axis should be the one orthogonal to the scatterer and the Ox and Oy axes can be freely chosen in a plane parallel to it.

3.1.2. The `Event` class The `Event` class contains information about a Compton event, namely its ID, the energy and position of the first and second hit, the initial energy if known and the Compton angle. CoReSi processes each hit position of an

event to associate it with a camera and layer where it occurred. Events with interaction positions outside the camera's layers are discarded. The material of the layer where the interaction occurred is identified to further compute interaction probabilities. Subsequently, a list of valid events is generated and stored in memory.

3.1.3. The `Model` class This class serves to compute a row of the system matrix. Unlike other modalities of emission tomography, where the model is based on line integrals and the system matrix is never explicitly computed, in this code the projection and backprojection operators consist in applying matrix multiplications between the system matrix and the image or its transpose and the data. The projection and backprojection operators are implemented in this class.

Two methods have been implemented to account for the uncertainties in the energy measurements: the parallel cone thickness and the angular cone thickness. The former is intended for ideal events where a small value of the thickness parameter is chosen. The latter can account for Doppler broadening or energy measurement uncertainties. The greater the uncertainties, the greater the broadening should be. The figures in this paper were produced with parameters corresponding to Doppler broadening in silicon. To this purpose, a mixture of two Gaussians was fitted to the angular resolution measure at different energies (Feng et al. 2021). The parallel thickness model produces a conical shell centred on the Compton cone with a constant thickness throughout the cone. For the angular thickness model, the conical shell widens as the distance from the apex increases (see figure 2).

Once measurement uncertainties are modelled, the system matrix is no longer sparse. Therefore, we have chosen not to store it in memory. Sensitivity is computed using the functions implemented in this class and stored on disk for further use.

3.1.4. The `MLEM` class Algorithm 1 details the implementation of the list-mode MLEM algorithm. The system matrix is computed on the fly at each iteration, one row at a time. Each row, indexed by i, is used to compute the corresponding element of the forward projection, i.e. $p_i = \sum_\ell t_{i\ell} \hat{\lambda}_\ell^k$, and a term from the sum that gives the backprojection of the errors, i.e. the vector $(t_{ij}/p_i)_{j=1,\dots,N_v}$. It is also possible to implement each iteration using explicit projection and backprojection operators, requiring two computations of the system matrix per iteration. The code allows iterations to be started from a volume stored on disk. Events that produce cones that do not intersect the volume are discarded and removed from the list of events for future iterations. An optional total variation regularization can be added as a denoising step between iterations.

Algorithm 1: List-mode MLEM algorithm

```
Data: e: the events
      K: the number of MLEM iterations
      s: the sensitivity
Result: λ the reconstructed image
λ ← 1
for i in range(n):
    next_value ← 0
    for event in e:
        line ← SM_line(e)
        if i = 0:
            next_value ← next_value + line
        else:
            forward_proj ← line · λ // dot product
            next_value ← next_value + line / forward_proj
    if i = 0:
        λ ← next_value
    else:
        λ ← λ / s × next_value

return λ
```

3.1.5. The `Image` and `Point` classes The `Image` class was used to store all multi-dimensional vectors computed by CoReSi, such as the row of the system matrix, the sensitivity or the reconstructed volume λ. In addition to the values, this class allows to store volume metadata such as the number of voxels and the size of the volumes in centimeters. To store the 3D Cartesian coordinates, a `Point` class was used. The class is a child of the numpy array class, so its vectorised operations could be used. Custom attributes `x`, `y` and `z` were added to make the code easier to read, as well as custom methods such as calculating coordinates in a new reference frame or normalization.

3.2. Monte Carlo sensitivity computation in CoReSi

The sensitivity is computed separately for each camera in the system and the results are summed. The analytical sensitivity models implemented in CoReSi were briefly presented in section 2.4. For the SM-like implementation, the algorithm loops through the cameras and the list of known energies of the source. For each voxel of the volume, a given number of events is simulated. Each event consists of a random hit drawn with uniform probability in a random scatterer layer and another random hit selected in a random absorber layer from the current CC. The Compton angle is computed geometrically and the deposited energies are then obtained from formula 2.

The sensitivity can be obtained from these simulated events in at least two different ways. The first one is to evaluate numerically the integral from (Muñoz et al. 2018). One drawback is that since these events are ideal, measurement uncertainties seem more difficult to account for. The method currently implemented in the software aims to approximate the identity $s = T^T 1$, true when all possible events are observed. Measurement uncertainties are treated in the same manner as in the system matrix. In practice, a system matrix row is computed for each of the

random events. The sensitivity image is then the summed backprojection of the list of events generated by this MC simulation, creating a coherence between the sensitivity calculation and a subsequent image reconstruction.

In spectral mode, the backprojection is done separately for each energy under consideration. This ensures that one energy does not have an impact on the others. This choice can be easily modified. Subtle differences exist between the sensitivities calculated for the same energy in spectral and mono-energetic modes. First, incomplete absorption is allowed in spectral mode. For instance, for a source emitting at both 364 and 511 keV, the probability of an event detected at 364 kev being produced by a photon emitted at the same energy (total absorption of the scattered photon) is less than one. This probability depends on the energy after scattering and is thus different from one event to another. Second, physical factors are not computed in the non spectral version (section 2.3.2). These factors depend only on the event but not on the voxel and cancel out in the reconstruction algorithm. Ignoring them improve computational speed. In theory, these factors should be taken into account in the computation of the sensitivity and is possible to do so by forcing the spectral mode in the mono-energetic setup. The real influence on the final images is still to be evaluated.

Several physical and even geometrical differences subsist between CoReSi and a dedicated particle-matter interaction code. For instance, in a physically correct simulation, a particle is emitted with an arbitrary direction in 4π, has a probability of interaction or escape in the scattering layers, scatters at an angle sampled from the Klein-Nishina differential cross-section, and then hits the absorber. Back-scattering leading to absorber-scatterer interactions is also possible. The chances of a γ ray scattering near the edge of the scatterer reaching the absorber are generally lower than for a γ ray scattering near the centre. Consequently, the distribution of hits between and within the layers is not uniform in reality, unlike in our implementation.

3.3. Event simulation tool

A simulation tool has been implemented to test the CoReSi code in different camera geometries. This tool allows the generation of ideal events. The user provides the source as a voxelised volume, the emission energy E_0, the camera characteristics and the desired number of events. The code produces two-hit events, with a first hit in a scatterer layer and a second hit in an absorber layer. The events are written to an ASCII text file.

For each event, a voxel is first randomly sampled from the source with a probability proportional to the voxel's intensity in the volume. A random point V_0 is drawn in the voxel following a uniform distribution. Second, a point V_1 is sampled from a random layer in a scatterer and an angle β is randomly chosen according to the Klein-Nishina differential cross-section corresponding to the given initial energy. Third, a given number of points $V_{2,i}$, $i = 1, \ldots, n$ are randomly chosen in the absorber. The geometric Compton scattering angle β_i is computed for each of them. A hit in the absorber is chosen as $V_{2,\iota}$, with:

$$\iota = \arg\min\{|\beta_i - \beta| \ : \ |\beta_i - \beta| \leq \epsilon, \ i = 1, \ldots, n\}, \tag{7}$$

where ϵ is a threshold value given by the user. If the set of values above the threshold is empty, the trial is invalidated and a new particle emission is simulated. This approach avoids calculating the intersection between a cone and a box, while maintaining a

Compton angle distribution consistent with the Klein-Nishina cross-section. Finally, the energy of the scattered photon E_1 is computed using the equation (2) and E_2 is computed by difference from the initial energy. Spatial and energy uncertainties can easily be added to the data in a post-processing step.

3.4. CoReSi configuration parameters and features of the code

The CoReSi variables are set in a YAML configuration file. It contains the path to the data file, to the pre-computed sensitivity if given, the path to the directory where the results will be saved. It contains a `cameras` section, listing the properties of the Compton cameras such as their number and positions, the position of scatterer and absorber layers in the generic camera and the materials they are made of. The `MLEM` section was used to set the number of iterations and gives the possibility to invoke the total variation regularization. Iterations can be started from the beginning or from a given iteration if the result for that iteration is stored in the right place. The user can specify if the energies of the γ-rays emitted by the source are known and give the list of E_0 values. A number of events to be used for the reconstruction, less than the total number of events acquired, can be specified.

A separate configuration file is used to define the constants associated with the materials, such as cross sections for various photon-matter interactions and the constants for the Gaussians that model the angular resolution induced by Doppler broadening in the system matrix. This makes it easy to support new materials.

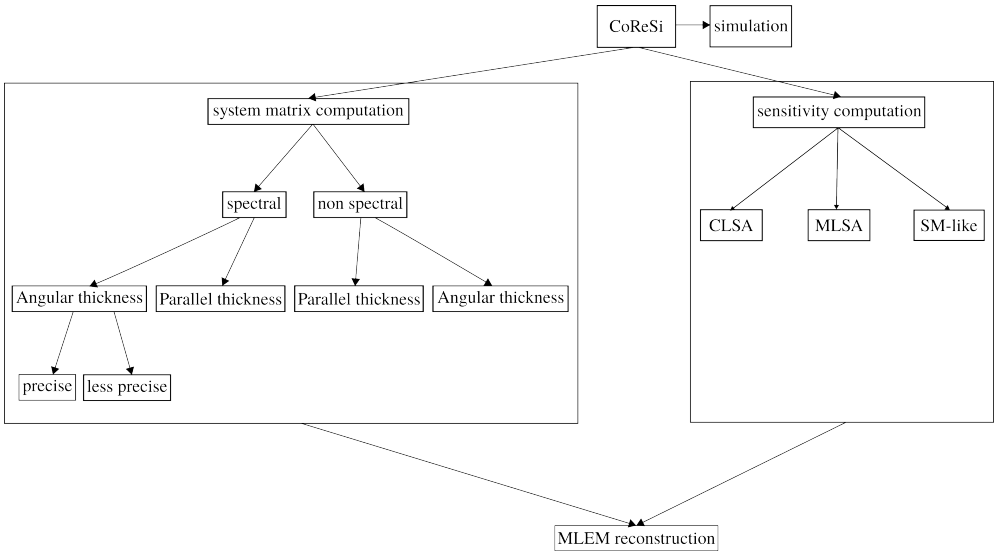


Figure 4: Features of CoReSi. CLSA: central layer solid angle, MLSA: multi-layer solid angle and SM-like: like system matrix.

CoReSi supports three modes: image reconstruction mode, sensitivity calculation and data simulation. The image reconstruction mode supports either a polyenergetic (spectral) or a monoenergetic (non-spectral) source. There are two options for the construction of the system matrix, namely angular or parallel thickness. Figure 4 shows a graph representation of the CoReSi features set. The three modes are mutually

exclusive: CoReSi can be used either for image reconstruction, sensitivity calculation or data simulation, which means that, for example, sensitivity has to be computed and stored on disk before image reconstruction. The main code is responsible for parsing the command line arguments, setting the loggers and either computing the sensitivity, reconstructing an image or performing a simulation depending on the command line arguments.

CoReSi includes a display option based on the matplotlib (Hunter 2007) library, which can be used either as a stand-alone tool or directly at the end of the image reconstruction.

For ease of deployment and use, a package has been created that allows users to easily install CoReSi along with its Python dependencies and to use CoReSi as an executable.

The CoReSi code currently implements near-field two-interaction CC image reconstruction. The typical geometry has a scatterer, possibly composed of several layers, and an absorber. Other geometries, *e.g.* multiple absorber layers (Nakano et al. 2020), are allowed although the interaction/escape probabilities are modelled with less precision. Stepwise rotating cameras (Omata et al. 2022) can be implemented as multiple cameras. Single detector cameras (Turecek et al. 2020, Lee et al. 2022), where the scatterer and absorber are confounded and the interactions are measured in 3D, can be considered with the same limits in accuracy of the physical model as for the multiple absorber cameras. The code can be used to reconstruct images for nuclear medicine and homeland security applications (including nuclear decommissioning), in single energy or spectral mode. Doppler broadening has a large effect in this low energy configuration and its modelling is supported in the code. The initial energy (or energies) of the source can be set as unknown, although a known initial energy results in better image quality. For hadron therapy prompt γ imaging, where the energy spectrum can be seen as continuous and broad (up to ~ 10 MeV), the user can set either the unknown initial energy or a discrete spectral mode. Far-field imaging in spherical coordinates (Zhang et al. 2023), three-γ events (Barrientos et al. 2021) and electron tracking (Kurosawa et al. 2010) are not currently supported. Table 2 shows a small selection of CC studies for medical applications (diagnosis in nuclear medicine, targeted radionuclide therapy, hadron therapy), radioprotection and homeland security (in near-field mode) with their compatibility to CoReSi.

Table 2: Examples of CC designs for medical applications, radioprotection and homeland security along with their compatibility with CoReSi. NM: Nuclear medicine, HS: Homeland security, HT: Hadrontherapy verification, AS: astronomy. The number of layers of one component of the camera (Sca or Abs) is the number between the parenthesis. N/A means the information is Not Available

Ref	Camera geometry	Material	Purpose	Reconstruction method	Compatibility with CoReSi
(Sullivan et al. 2009)	Sca(3), Abs(1)	Si / CsI	N/A	LM-MLEM	yes
(Kurosawa et al. 2010)	Sca(1), Abs(1)	gaseous / LaBr3(Ce)	NM, AS	N/A	no (electron tracking)
(Alnaaimi et al. 2011)	Sca(1), Abs(1)	Ge	NM	backprojection	yes
(Azevedo et al. 2013)	Sca(1), Abs(1)	gaseous (Xe) / CsI	NM	N/A	no (electron tracking)
(Krimmer et al. 2015)	Sca(7), Abs(1)	Si / BGO	HT	N/A	yes
(Hmissi et al. 2018)	Sca(1), Abs(1)	$CeBr_3$ / LYSO:Ce	HS	LM-MLEM	yes
(Nakano et al. 2020)	Sca(1), Abs(3)	Si / CdTe	NM	backprojection	yes (bi-energy)
(Turecek et al. 2020)	single detector	CdTe	N/A	backprojection	yes
(Yabu et al. 2021)	Sca(1), Abs(1)	Si/CdTe	NM	LM-MLEM	no (moving sample)
(Barrientos et al. 2021)	3 stacked detectors	$LaBr_3$	HT	LM-MLEM	no (three γ)
(Omata et al. 2022)	Sca(1), Abs(1)	N/A	NM	LM-MLEM	yes (4 moving cameras)
(Lee et al. 2022)	single detector	GAGG	HS	LM-MLEM	yes
(Borja-Lloret et al. 2023)	Sca(1), Abs(1)	$LaBr_3$	HT	LM-MLEM	yes (spectral)
(Zhang et al. 2023)	3 stacked detectors	$Gd_3Al_2Ga_3O_{12}$:Ce	NM	LM-MLEM	no (far-field)

4. Data simulation and evaluation of CoReSi

Features of the code are illustrated on data simulated for the study (Muñoz et al. 2022). The simulations were performed with the module CC-Mod (Etxebeste et al. 2020) of GATE (Sarrut et al. 2021) version 9.0, using the standard electromagnetic physics list. A source consisting of six cylinders of diameters 10, 13, 17, 22, 28 and **37 mm** and thickness **4 mm** emitted simultaneously and with equal probability at **140 keV**, **245 keV**, **364 keV** and **511 keV** for the spectral test. For the mono-energy test the source emitted at **364 keV**. Figure 5 shows a 2D slice through the source (a) and the acquisition geometry (b).

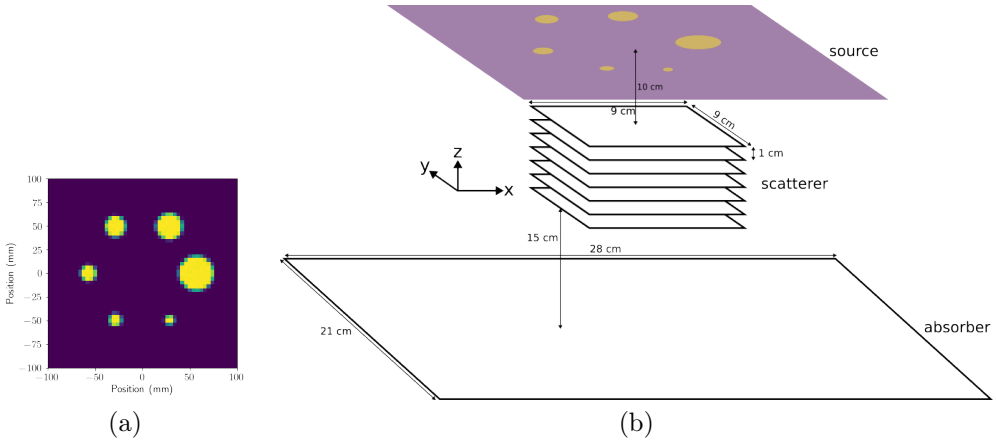


Figure 5: Simulated acquisition geometry. Image of the source composed of six cylinders in the xy plane (this image is represented in voxelized geometry, although the geometry is analytically defined in GATE) (a) and three dimensional representation of the camera and the GATE source (b).

The camera had a scatterer with 7 silicon layers of dimensions $9\,\text{cm} \times 9\,\text{cm} \times 0.2\,\text{cm}$ with a **1 cm** gap between each. The closest was **10 cm** away from the centre of the source. There was an absorber layer of bismuth germanium oxide (BGO) **15 cm** below the scatterers with dimensions of $28\,\text{cm} \times 21\,\text{cm} \times 3\,\text{cm}$. Ideal events were recorded, with a first hit in the scatterer and a second hit in the absorber. Only the Doppler broadening was taken into account, no spatial or energy uncertainties were added to the simulated values. Incomplete absorptions are allowed for photons that escape the camera after a Compton interaction in the absorber.

The CoReSi reconstruction tool was validated against the previous CoReSi implementation in C++, which was single-threaded and processed the source voxels sequentially. Perfect match with results from (Muñoz et al. 2022) was observed. To determine the speedup of the new implementation, reconstruction timings were compared for different reconstructed volume sizes and for different numbers of events. These tests were performed on an NVIDIA v100 GPU. The imaging experiments were performed on a Quadro RTX 8000 GPU.

For the purpose of testing the simulation capability within CoReSi, a number of 2×10^4 ideal events were simulated from a thin cylindrical source emitting at **364 keV** placed at **10 cm** from the same camera. The source had a diameter of 27.5

mm and a thickness of 2.5 mm. Uniform emission probabilities were considered in this experiment, but this is not a constraint from the code which can treat any voxelized source distribution.

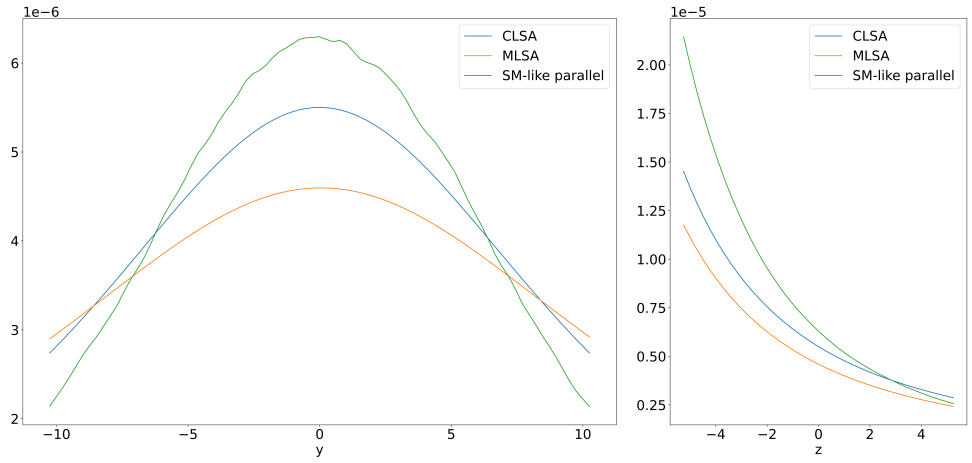
For all imaging experiments, three-dimensional volumes were reconstructed with 20 iterations of the LM-MLEM algorithm. Reconstructions (and sensitivities) were computed for a volume of $20.25\,\text{cm} \times 20.25\,\text{cm} \times 10.25\,\text{cm}$ and $81\,\text{voxels} \times 81\,\text{voxels} \times 41\,\text{voxels}$, centred at $(0, 0, 0)$. The $4\,\text{mm}$ thick source (Gate simulation) is located in the three slices from the centre of the volume, the 2.5 mm source (CoReSi simulation) is located in the middle slice.

5. Results

This section showcases some of the features of the code and present a comparison in terms of computing time with a single-threaded CPU implementation in C++.

5.1. Comparison of sensitivity models

Three sensitivity models are currently implemented in CoReSi: central layer solid angle (CLSA), multi-layer solid angle (MLSA) and like system matrix (SM-like). Figure 6 shows central profiles through the sensitivity of the camera in the monoenergetic configuration. As the parallel model was chosen for the SM-like sensitivity computation, neither physical factors (probabilities of interaction) nor Doppler broadening are accounted for, thus the values do not depend on the energy. All the sensitivity distributions have been normalized to their mean values in 3D.



(a) $x = z = 0$ (b) $x = y = 0$

Figure 6: Profiles from the CLSA, MLSA and SM-like sensitivities, the last one with a parallel thickness model for a mono-energetic source. Profiles through the center of the volume: parallel (a), perpendicular to the camera (b). CLSA: central layer solid angle, MLSA: multi-layer solid angle and SM-like: like system matrix. The vertical axes represent normalized counts.

Figure 7 shows central profiles of the spectral SM-like sensitivity for the energies **140 keV**, **245 keV**, **364 keV** and **511 keV**, with angular thickness (a) and parallel thickness (b) models. The mono-energy sensitivity at **364 keV** with the same model is shown for comparison. In each figure, the five three-dimensional sensitivity distributions have been normalised to their mean.

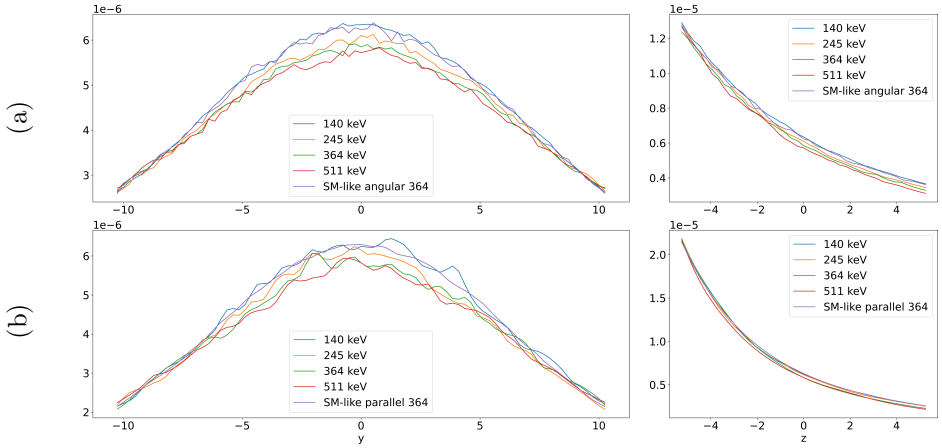


Figure 7: Spectral SM-like sensitivity, for angular (a) and parallel (b) thickness models. In each panel, the profile from the mono-energy SM-like sensitivity at 364 keV is shown for comparison. Profiles parallel to the camera are shown in the first column, perpendicular to it in the second. All profiles pass through the origin, which is also the centre of the volume. The vertical axes represent normalized counts.

5.2. Reconstructed images

Slices from the three-dimensional volumes representing the monoenergetic source are shown in Figure 8. They have been reconstructed with the angular (a) and parallel (b) thickness models from 1.1×10^5 events, with the Compton angle computed with $E_0 = 364$ keV. The sensitivities are those computed in the previous test with the SM-like model and 20 LM-MLEM iterations were performed. No normalisation or denoising was applied to the volumes. A slice parallel to the camera and a slice perpendicular to it are shown for each volume.

Figures 9 and 10 show slices from the four volumes reconstructed in spectral mode, for the source emitting at **140 keV**, **245 keV**, **364 keV** and **511 keV** energies. The images from figure 9 were reconstructed with the angular thickness and the images from figure 10 were reconstructed with the parallel thickness model. In both cases, the same 1.1×10^5 events were processed and 20 LM-MLEM iterations were performed. The sensitivities computed with the SM-like model. The results are shown without any post-processing.

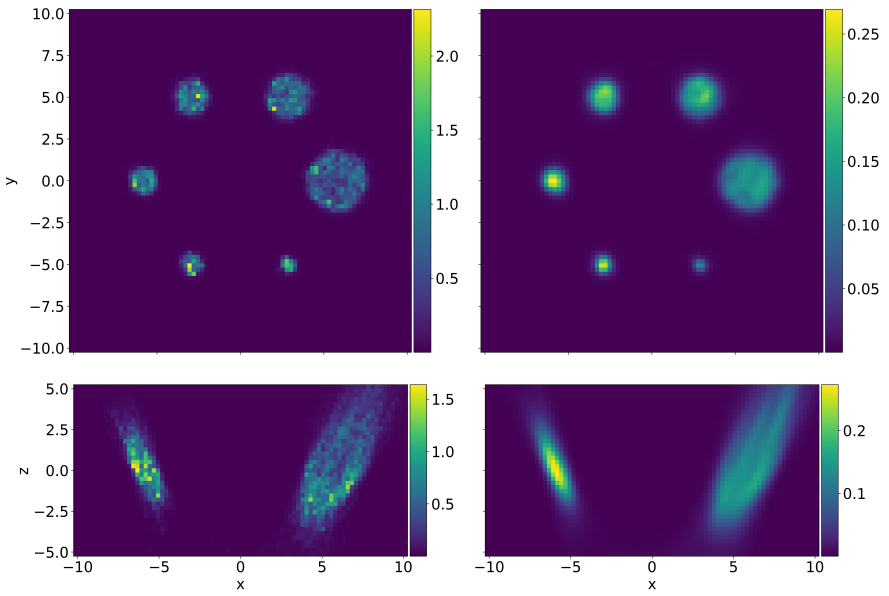


Figure 8: Reconstruction of a source emitting at **364 keV** with angular (a) and parallel (b) thickness in non-spectral mode. Slices at $z = 0$ are shown in the top row and at $y = 0$ in the bottom row.

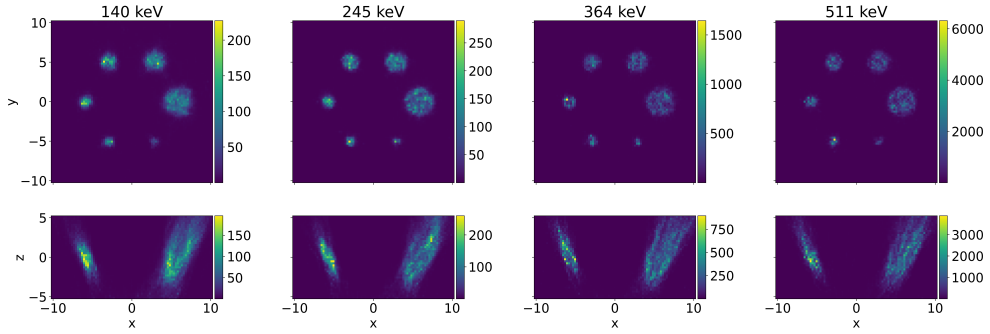


Figure 9: Spectral reconstruction with the angular thickness model at **140 keV**, **245 keV**, **364 keV** and **511 keV** from left to right.

5.3. Illustration of the event simulation module

Figure 11 shows a simulation made with CoReSi for a thin cylindrical source emitting photons with energy **364 keV**. The voxelized source used for the simulation, the emission points and the central slice parallel to the camera from the reconstructed volume are shown on the first row from left to right. On the second row, the left panel shows the distribution of the observed Compton scattering angles (red histogram) and the target distribution computed from the Klein-Nishina differential cross section (blue

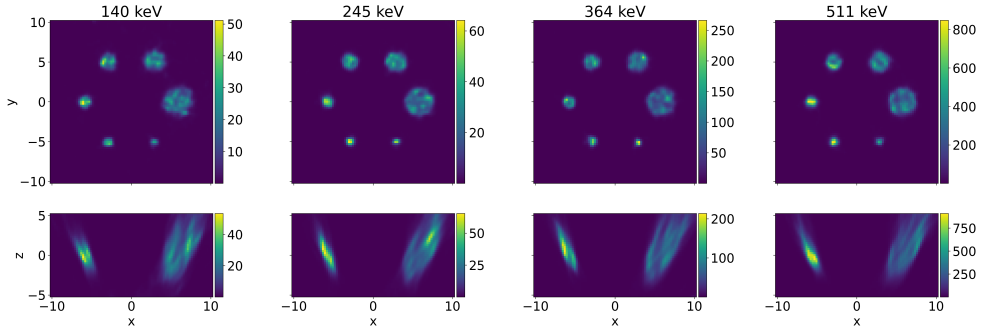


Figure 10: Spectral reconstruction with the parallel thickness model at **140 keV**, **245 keV**, **364 keV** and **511 keV** from left to right.

line). The differences are caused by the geometry of the camera that favours small angles and avoids large ones. The right image is a slice perpendicular to the camera at the centre of the reconstructed volume. The images are reconstructed with the parallel cone thickness model (as the events generated by the simulation are ideal), with 20 LM-MLEM iterations. The simulation generated $\mathbf{2 \times 10^4}$ events and took about 30 minutes to run on a laptop with no GPU and an Intel® Core™ i5-7200U CPU.

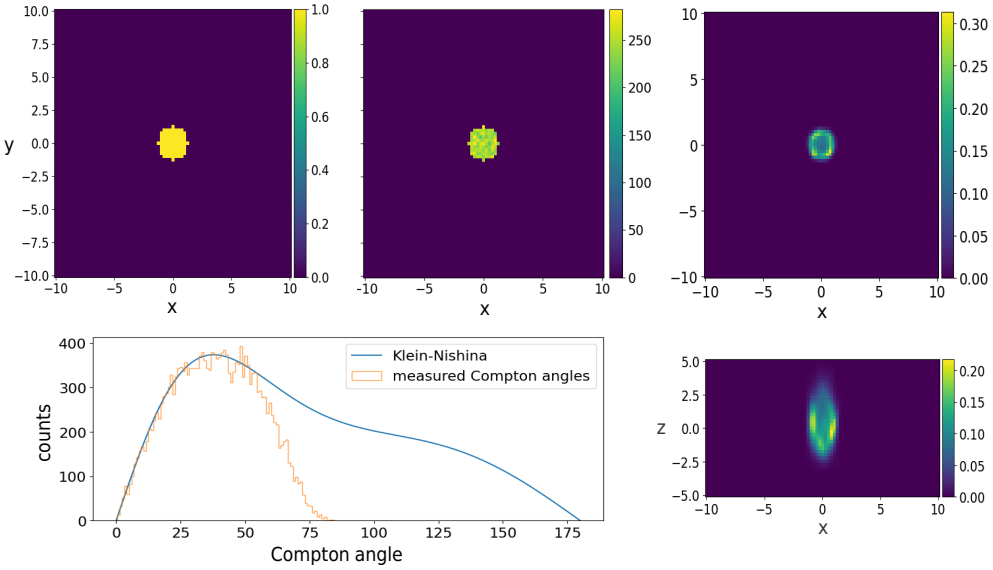


Figure 11: CoReSi simulation and reconstruction chain. From left to right and top to bottom, the voxelized source, the histogram of emission points drawn from the source, the central slice parallel to the camera from the reconstructed volume, the histogram of the Compton scattering angles generated during the simulation along with the Klein-Nishina differential cross section, and the central slice perpendicular to the camera from the reconstructed volume.

5.4. Computing time

Figure 12 shows a comparison between the C++ single-threaded implementation on CPU and without parallel computing capabilities, and the Python version both on the CPU and GPU. Three volume sizes were considered, namely 10^4, 10^5 and 10^6 voxels. Whereas the computing time of the C++ implementation scales linearly with the number of voxels, the Python implementation exhibits nearly consistent computing times across all dimensions, thanks to the multi-threading capabilities and GPU support provided by PyTorch. The values are shown for 5×10^3 events and four LM-MLEM iterations on an NVIDIA V100 GPU.

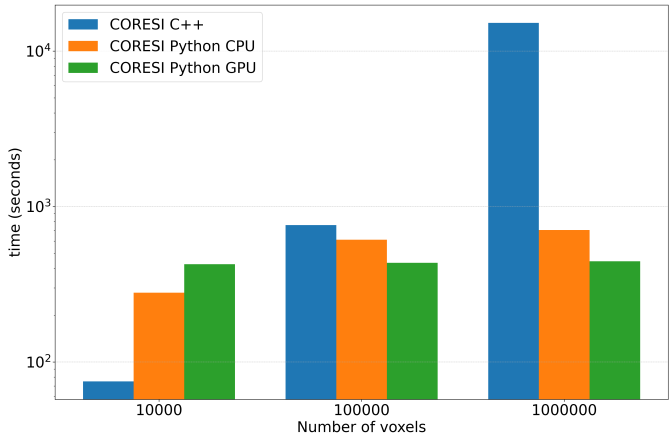


Figure 12: Comparison of computing times for a C++ implementation and a Python implementation using PyTorch, for three volume dimensions. Four LM-MLEM iterations and 5×10^3 events were used for the reconstruction, on an NVIDIA V100 GPU.

More specifically, for a volume of 10^4 voxels, the C++ version took 75 seconds, while the Python version with GPU took 426 seconds. For a volume of 10^5 voxels, the C++ version took 760 seconds, while the Python version with GPU took 435 seconds. For a volume of 10^6 voxels, the C++ version took 15,192 seconds, while the Python version with GPU took 445 seconds. Tests analysing the influence of the number of events showed a linear relationship between the reconstruction time and the number of events for both the C++ and Python versions. Table 3 gives some examples of computing times observed during the reconstruction of the images presented in the article on a Quadro RTX 8000 graphics processing unit with 32 cores. Sensitivity computation with the models CLSA (central layer solid angle) and MLSA (multi-layer solid angle) took a few seconds. For the angular thickness model in the mono-energetic case we activated the option "spectral mode", meaning that all the physical factors were evaluated for a more accurate modelling. This has resulted in longer calculation times compared to the parallel thickness model. In the polyenergetic case, the parallel thickness reconstruction was longer to be computed. The difference can be explained by the number of non-zero elements in the system matrix, larger for the parallel thickness with our choice of parameters.

Table 3: Computing time (single run) for sensitivity and image reconstruction. In the SM-like configuration, the sensitivity is computed by backprojecting 5×10^4 sampled events at each energy. For the reconstructions, 1.1×10^5 events were considered in both the mono and poly-energetic cases. For the sensitivity, SM-like means that the model is the same as for the system matrix.

Method	Duration
Sensitivity SM-like monoenergetic parallel thickness	1 minute
Sensitivity SM-like monoenergetic angular thickness	3 minutes
Sensitivity SM-like polyenergetic parallel thickness	28 minutes
Sensitivity SM-like polyenergetic angular thickness	26 minutes
Reconstruction monoenergetic parallel thickness	38 minutes
Reconstruction monoenergetic angular thickness	2 hours and 10 minutes
Reconstruction polyenergetic parallel thickness	5 hours and 5 minutes
Reconstruction polyenergetic angular thickness	4 hours and 40 minutes

6. Discussion

In this paper, a new Python library called CoReSi is presented, intended as a novel tool for Compton camera SPECT imaging studies. The underlying methods and strategies are detailed, as well as the key classes within the code, and its features. Several of its capabilities are demonstrated using MC-simulated data as well as data generated by the library itself. CoReSi is designed for applications in nuclear medicine, particle therapy, and near-field homeland security.

In a separate experiment not shown here, the results produced by CoReSi have been validated against the C++ code that generated the images published in (Muñoz et al. 2022). Discrepancies between the images therein and the ones from this paper can be explained by the two-dimensional versus three-dimensional reconstruction, by the logarithmic versus linear color scale chosen for visualization, and marginally by a slight modification of the sensitivity computation.

Computing times for image reconstruction were recorded and compared with the previous C++ version. The GPU version of the CoReSi implementation was faster than its CPU or C++ counterparts, with the speed remaining roughly constant as the number of voxels increased, thanks to GPU-based parallel computing. The C++ version was faster than the Python version for small reconstructed volumes, suggesting that parts of the Python code not dealing with volumes are slower than the C++. Only the list-mode MLEM algorithm is currently implemented. The Ordered Subsets Expectation Maximization method (OSEM) could accelerate reconstruction time and should be straight-forward to implement (Hudson & Larkin 1994).

Fundamental to any reconstruction algorithm are system matrix and sensitivity vector computations. These are also the most computationally intensive parts of the algorithm. In tomographic modalities that work with data stored in sinograms, projection and backprojection operators are implemented instead. Since list-mode storage is the only option in Compton imaging, the specific and optimised ray tracing algorithms cannot be used. CoReSi implements the computation of the system matrix row by row and allows to use it directly in the LM-MLEM algorithm. The projection and backprojection operators are built on top of the system matrix computation

function. The system matrix model implemented in the code is relatively accurate, but could be improved by taking into account other sources of uncertainty in CC acquisitions. These models could be difficult to derive mathematically and slow down the reconstruction, which is already much slower than for other tomographic modalities. The improvement in accuracy is at the expense of higher computing costs.

Three sensitivity models are currently implemented. As can be seen from the profiles, they tend to differ from each other. The differences between the CLSA and MLSA sensitivities is due to the large number of scatterer layers. Only the central layer is accounted for in the first, a sum with respect to all the layers is computed for the second. The influence of a layer decreases with the distance to the volume, which explains the more spread-out shape of MLSA. Both of them ignore the shape of the absorber and assume that all scattered photons produce an event with equal probability. This assumption has been removed from the SM-like model, which considers a first hit in the scatterer and a second in the absorber. Forward scattering with relatively small Compton angles are favoured, which may explain the narrowest shape of the SM-like sensitivity. The true shape of the sensitivity, defined as the probability of detection of a photon, strongly depends on the acquisition conditions, for instance measurement uncertainties and rejection strategies. It is likely that the proposed models should be adapted to every single experiment. However, for relatively small extent sources as well as for the small prototype cameras that represent the majority of the published studies, the impact on the reconstructed images should be limited. As it approximates the identity $\boldsymbol{s} = \boldsymbol{T}^T\mathbf{1}$, the SM-like model might be mathematically more sound for solving the reconstruction inverse problem. However, the computing times are significantly increased compared to CLSA and MLSA, and the real impact on the reconstructed images is still to be evaluated. This impact might be influenced by the experimental conditions. SM-like sensitivity profiles shown in this paper present oscillations due to statistical factors. To prevent them from propagating into the reconstructed images, the user could either increase the number of samples or smooth the sensitivity volume.

The code can be applied to both ideal and non-ideal data. For ideal cases, the parallel thickness mode is recommended, whereas for non-ideal cases, the angular thickness mode offers better performance. In comparing the two modes, with the images provided in this paper, the parallel thickness mode produces smoother results. This can be attributed to the use of a Doppler-corrected system matrix in the angular thickness mode, which more accurately represents the underlying data. While increasing cone thickness enhances image smoothness, it also introduces blurring. Another explanation for the difference observed between the two modes is that the cone thickness in the parallel mode was set arbitrarily to the diagonal of a voxel, a value larger than the ones observed for the angular thickness. To achieve smoother images while still preserving sharp edges, total variation regularization can be used within CoReSi.

The simulation tool integrated into CoReSi is primarily designed to assist developers of image reconstruction algorithms and is not intended to compete with software specializing in MC simulations of imaging systems. At present, it only supports monoenergetic sources and ideal data. A potential enhancement would be to enable the simulation of multiple source energies simultaneously, with assigned probabilities for each energy. This upgrade would enhance data generation for spectral reconstruction testing.

Despite significant efforts, certain physical processes, such as energy uncertainties

in detectors or escaping electrons, cannot be accurately modeled. These factors influence the system's point spread function (PSF), resulting in a spatially varying, source-dependent blur (Erlandsson et al. 2012). Estimating the PSF and performing deconvolution are challenging tasks, commonly referred to as resolution recovery when integrated into the reconstruction process (Sureau et al. 2008, Kim et al. 2013). The inclusion of deconvolution and denoising techniques, using either variational or plug-and-play methods, is simplified by the use of a Python-based open-source software with a user-friendly structure.

7. Conclusion

We have developed CoReSi, a Python library dedicated to image reconstruction and data simulation for Compton cameras. This library can be used for three-dimensional iterative image reconstruction either with list-mode MLEM or with other algorithms making use of projection and back-projection operators. The code is open-source and can thus be modified. For fast prototyping purposes, a simplified data simulation module was included. CoReSi is designed for near-field applications, as nuclear medicine, particle therapy, and near-field homeland security. As a 3D reconstruction software, it may not be well-suited for far-field observations in astronomy or environmental measurements. Among various perspectives for future development, one can mention, for instance, the implementation of procedures to account for attenuation in the phantom or patient.

Acknowledgments

The authors thank Xavier Lojacono, Estelle Hilaire, Yuemeng Feng, Enrique Muñoz for contributions to the previous C++ version of the code and Ane Etxebeste for help with data simulation. This work was performed within the framework of the LABEX PRIMES (ANR-11-LABX-0063) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007). This work was performed using HPC resources from GENCI–IDRIS (Grant 2024-AD011011556R3).

References

Alnaaimi M A, Royle G J, Ghoggali W, Banoqitah E, Cullum I & Speller R D 2011 Performance evaluation of a pixellated Ge Compton camera *Physics in Medicine and Biology* **56**(12), 3473–3486.

Aprile E, Aalbers J, Agostini F, Alfonsi M, Althueser L, Amaro F, Antochi V C, Angelino E, Angevaare J, Arneodo F et al. 2020 Energy resolution and linearity of XENON1T in the MeV energy range *The European Physical Journal C* **80**, 1–9.

Azevedo C, Pereira F, Lopes T, Correia P, Silva A, Carramate L, Covita D & Veloso J 2013 A Gaseous Compton Camera using a 2D-sensitive gaseous photomultiplier for Nuclear Medical Imaging *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **732**, 551–555.

Barrientos L, Borja-Lloret M, Etxebeste A, Muñoz E, Oliver J F, Ros A, Roser J, Senra C, Viegas R & Llosá G 2021 Performance evaluation of MACACO II Compton camera *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **1014**, 165702.

Borja-Lloret M, Barrientos L, Bernabéu J, Lacasta C, Muñoz E, Ros A, Roser J, Viegas R & Llosá G 2023 Influence of the background in Compton camera images for proton therapy treatment monitoring *Physics in Medicine & Biology* **68**(14), 144001.

Cree M & Bones P 1994 Towards direct reconstruction from a gamma camera based on Compton scattering *IEEE Transactions on Medical Imaging* **13**(2), 398–407. Conference Name: IEEE Transactions on Medical Imaging.

Dempster A P, Laird N M & Rubin D B 1977 Maximum likelihood from incomplete data via the *EM* algorithm *Journal of the Royal Statistical Society Series B: Statistical Methodology* **39**(1), 1–22.

Draeger E, Peterson S, Mackin D, Chen H, Beddar S & Polf J C 2017 Feasibility studies of a new event selection method to improve spatial resolution of Compton imaging for medical applications *IEEE transactions on radiation and plasma medical sciences* **1**(4), 358–367.

Erlandsson K, Buvat I, Pretorius P H, Thomas B A & Hutton B F 2012 A review of partial volume correction techniques for emission tomography and their applications in neurology, cardiology and oncology *Physics in Medicine & Biology* **57**(21), R119.

Etxebeste A, Dauvergne D, Fontana M, Létang J, Llosá G, Munoz E, Oliver J, Testa É & Sarrut D 2020 CCMod: a GATE module for Compton camera imaging simulation *Physics in Medicine & Biology* **65**(5), 055004.

Feng Y, Etxebeste A, Sarrut D, Létang J M & Maxim V 2019 3-D reconstruction benchmark of a Compton camera against a parallel-hole gamma camera on ideal data *IEEE Transactions on Radiation and Plasma Medical Sciences* **4**(4), 479–488.

Feng Y, Létang J M, Sarrut D & Maxim V 2021 Influence of Doppler broadening model accuracy in Compton camera list-mode MLEM reconstruction *Inverse Problems in Science and Engineering* **29**(13), 3509–3529.

Fontana M, Dauvergne D, Létang J M, Ley J L & Testa E 2017 Compton camera study for high efficiency SPECT and benchmark with Anger system *Physics in Medicine & Biology* . Publisher: IOP Publishing.

Hilaire E, Sarrut D, Peyrin F & Maxim V 2016 Proton therapy monitoring by Compton imaging: influence of the large energy spectrum of the prompt-γ radiation *Physics in Medicine & Biology* **61**(8), 3127. Publisher: IOP Publishing.

Hmissi M Z, Iltis A, Tata C, Zeufack G, Rodrigues L, Mehadji B, Morel C & Snoussi H 2018 First images from a CeBr3 /LYSO:Ce Temporal Imaging portable Compton camera at 1.3 MeV *in* '2018 IEEE Nuclear Science Symposium and Medical Imaging Conference Proceedings (NSS/MIC)' pp. 1–3. ISSN: 2577-0829.

Hudson H & Larkin R 1994 Accelerated image reconstruction using ordered subsets of projection data *IEEE Transactions on Medical Imaging* **13**(4), 601–609. Conference Name: IEEE Transactions on Medical Imaging.

Hunter J D 2007 Matplotlib: A 2D graphics environment *Computing in Science & Engineering* **9**(3), 90–95. Publisher: IEEE COMPUTER SOC.

Kim S M, Lee J S, Lee M N, Lee J H, Lee C S, Kim C H, Lee D S & Lee S J 2007 Two approaches to implementing projector–backprojector pairs for 3d reconstruction from compton scattered data *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **571**(1-2), 255–258.

Kim S M, Seo H, Park J H, Kim C H, Lee C S, Lee S J, Lee D S & Lee J S 2013 Resolution recovery reconstruction for a Compton camera *Physics in Medicine & Biology* **58**(9), 2823.

Krimmer J, Ley J L, Abellan C, Cachemiche J P, Caponetto L, Chen X, Dahoumane M, Dauvergne D, Freud N, Joly B, Lambert D, Lestand L, Létang J M, Magne M, Mathez H, Maxim V, Montarou G, Morel C, Pinto M, Ray C, Reithinger V, Testa E & Zoccarato Y 2015 Development of a Compton camera for medical applications based on silicon strip and scintillation detectors *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **787**, 98–101.

Kurosawa S, Kubo H, Hattori K, Ida C, Iwaki S, Kabuki S, Kubo A, Kunieda E, Miuchi K, Nakahara T, Nishimura H, Parker J D, Takada A, Takahashi M, Tanimori T & Ueno K 2010 Development of an 8×8 array of LaBr3(Ce) scintillator pixels for a gaseous Compton gamma-ray camera *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **623**(1), 249–251.

Lee H, Park J & Lee W 2022 Development of modified scintillator-based single-crystal position-sensitive 4π Compton camera for a portable radiation imaging device *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **1043**, 167485.

Lojacono X, Maxim V & Prost R 2011 Calcul de la matrice des probabilités de transfert en imagerie Compton 3D *in* '23ième Colloque sur le traitement du signal et des images' GRETSI - Groupe de Recherche en Traitement du Signal et des Images Bordeaux pp. p. 1189–1192. Issue: 001-0297.

Maxim V 2014 Filtered backprojection reconstruction and redundancy in Compton camera imaging *IEEE Transactions on Image Processing* **23**(1), 332–341. Conference Name: IEEE Transactions on Image Processing.
URL: *https://ieeexplore.ieee.org/document/6651800*

Maxim V, Feng Y, Banjak H & Bretin E 2023 Tomographic reconstruction from Poisson distributed data: a fast and convergent EM-TV dual approach *Numerical Algorithms* **94**(2), 701–731.

Maxim V, Lojacono X, Hilaire E, Krimmer J, Testa E, Dauvergne D, Magnin I & Prost R 2016 Probabilistic models and numerical calculation of system matrix and sensitivity in list-mode MLEM 3D reconstruction of Compton camera images *Physics in Medicine and Biology* **61**(1), 243–264.

Merlin T, Stute S, Benoit D, Bert J, Carlier T, Comtat C, Filipovic M, Lamare F & Visvikis D 2018 CASToR: a generic data organization and processing code framework for multi-modal and multi-dimensional tomographic reconstruction *Physics in Medicine & Biology* **63**(18), 185005. Publisher: IOP Publishing.

Moon S & Haltmeier M 2017 Analytic inversion of a conical Radon transform arising in application of Compton cameras on the cylinder *SIAM Journal on Imaging Sciences* **10**(2), 535–557.

Muñoz E, Barrientos L, Bernabéu J, Borja-Lloret M, Llosá G, Ros A, Roser J & Oliver J F 2020 A spectral reconstruction algorithm for two-plane Compton cameras *Physics in Medicine & Biology* **65**(2), 025011.

Muñoz E, Barrientos L, Bernabéu J, Borja-Lloret M, Llosá G, Ros A, Roser J & Oliver J F 2020 A spectral reconstruction algorithm for two-plane Compton cameras *Physics in Medicine & Biology* **65**(2), 025011.

Muñoz E, Barrio J, Bernabéu J, Etxebeste A, Lacasta C, Llosá G, Ros A, Roser J & Oliver J F 2018 Study and comparison of different sensitivity models for a two-plane Compton camera *Physics in Medicine & Biology* **63**(13), 135004.

Muñoz E, Etxebeste A, Dauvergne D, Létang J M, Sarrut D, Maxim V & Testa E 2022 Imaging of polychromatic sources through Compton spectral reconstruction *Physics in Medicine & Biology* .

Muñoz E, Ros A, Borja-Lloret M, Barrio J, Dendooven P, Oliver J F, Ozoemelam I, Roser J & Llosá G 2021 Proton range verification with MACACO II Compton camera enhanced by a neural network for event selection *Scientific reports* **11**(1), 9325.

Nakano T, Sakai M, Torikai K, Suzuki Y, Takeda S, Noda S e, Yamaguchi M, Nagao Y, Kikuchi M, Odaka H, Kamiya T, Kawachi N, Watanabe S, Arakawa K & Takahashi T 2020 Imaging of ^{99m} Tc-DMSA and ^{18} F-FDG in humans using a Si/CdTe Compton camera *Physics in Medicine & Biology* **65**(5), 05LT01.

Omata A, Masubuchi M, Koshikawa N, Kataoka J, Kato H, Toyoshima A, Teramoto T, Ooe K, Liu Y, Matsunaga K et al. 2022 Multi-modal 3D imaging of radionuclides using multiple hybrid Compton cameras *Scientific reports* **12**(1), 2546.

Ordonez C E, Bolozdynya A & Chang W 1997 Dependence of angular uncertainties on the energy resolution of Compton cameras *in* '1997 IEEE Nuclear Science Symposium Conference Record' Vol. 2 IEEE pp. 1122–1125.

Parajuli R K, Sakai M, Parajuli R & Tashiro M 2022 Development and applications of Compton camera—a review *Sensors* **22**(19).
URL: *https://www.mdpi.com/1424-8220/22/19/7374*

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L et al. 2019 Pytorch: An imperative style, high-performance deep learning library *Advances in neural information processing systems* **32**.

Polson L, Fedrigo R, Li C, Sabouri M, Dzikunu O, Ahamed S, Karakatsanis N, Kurkowska S, Esquinas P, Sheikhzadeh P & Rahmim A 2024 Pytomography: A python library for medical image reconstruction *Available at SSRN 4865134* .

Rit S, Vila Oliva M, Brousmiche S, Labarbe R, Sarrut D & Sharp G C 2014 The Reconstruction Toolkit (RTK), an open-source cone-beam CT reconstruction toolkit based on the Insight Toolkit (ITK) *Journal of Physics: Conference Series* **489**, 012079.

Sarrut D, Bała M, Bardiès M, Bert J, Chauvin M, Chatzipapas K, Dupont M, Etxebeste A, M Fanchon L, Jan S, Kayal G, S Kirov A, Kowalski P, Krzemien W, Labour J, Lenz M, Loudos G, Mehadji B, Ménard L, Morel C, Papadimitroulas P, Rafecas M, Salvadori J, Seiter D, Stockhoff M, Testa E, Trigila C, Pietrzyk U, Vandenberghe S, Verdier M A, Visvikis D, Ziemons K, Zvolský M & Roncali E 2021 Advanced Monte Carlo simulations of emission tomography imaging systems with GATE *Physics in Medicine and Biology* **66**(10).

Sauve A, Hero A, Rogers W, Wilderman S & Clinthorne N 1999 3D image reconstruction for a Compton SPECT camera model *IEEE Transactions on Nuclear Science* **46**(6), 2075–2084.

Conference Name: IEEE Transactions on Nuclear Science.

Seo Y 2019 Quantitative imaging of alpha-emitting therapeutic radiopharmaceuticals *Nuclear medicine and molecular imaging* **53**(3), 182–188.

Shepp L A & Vardi Y 1982 Maximum likelihood reconstruction for emission tomography *IEEE transactions on medical imaging* **1**(2), 113–122.

Smith B 2005 Reconstruction methods and completeness conditions for two Compton data models *JOSA A* **22**(3), 445–459.

Sullivan J P, Tornga S R & Rawool-Sullivan M W 2009 Extended radiation source imaging with a prototype Compton imager *Applied Radiation and Isotopes* **67**(4), 617–624.

Sureau F C, Reader A J, Comtat C, Leroy C, Ribeiro M J, Buvat I & Trébossen R 2008 Impact of image-space resolution modeling for studies with the high-resolution research tomograph *Journal of Nuclear Medicine* **49**(6), 1000–1008.

Tachella J, Chen D, Hurault S & Terris M 2023 'DeepInverse: A deep learning framework for inverse problems in imaging'.
URL: *https://deepinv.github.io/deepinv/*

Takeda S, Harayama A, Ichinohe Y, Odaka H, Watanabe S, Takahashi T, Tajima H, Genba K, Matsuura D, Ikebuchi H, Kuroda Y & Tomonaka T 2015 A portable Si/CdTe Compton camera and its applications to the visualization of radioactive substances *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* .

Terzioglu F, Kuchment P & Kunyansky L 2018 Compton camera imaging and the cone transform: a brief overview *Inverse Problems* **34**(5), 054002.

Thielemans K, Tsoumpas C, Mustafovic S, Beisel T, Aguiar P, Dikaios N & Jacobson M W 2012 STIR: software for tomographic image reconstruction release 2 *Physics in Medicine & Biology* **57**(4), 867.

Turecek D, Jakubek J, Trojanova E & Sefc L 2020 Single layer Compton camera based on Timepix3 technology *Journal of Instrumentation* **15**(01), C01014–C01014.

Tuy H K 1983 An inversion formula for cone-beam reconstruction *SIAM Journal on Applied Mathematics* **43**(3), 546–552.

van Aarle W, Palenstijn W J, De Beenhouwer J, Altantzis T, Bals S, Batenburg K J & Sijbers J 2015 The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography *Ultramicroscopy* **157**, 35–47.
URL: *https://www.sciencedirect.com/science/article/pii/S0304399115001060*

Watanabe S, Tajima H, Fukazawa Y, Ichinohe Y, Enoto T, Fukuyama T, Furui S, Genba K, Hagino K, Harayama A et al. 2014 The Si/CdTe semiconductor Compton camera of the ASTRO-H soft gamma-ray detector (SGD) *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **765**, 192–201.

Wilderman S, Clinthorne N, Fessler J & Rogers W 1998 List-mode maximum likelihood reconstruction of Compton scatter camera images in nuclear medicine *in* '1998 IEEE Nuclear Science Symposium Conference Record. 1998 IEEE Nuclear Science Symposium and Medical Imaging Conference (Cat. No.98CH36255)' Vol. 3 pp. 1716–1720 vol.3. ISSN: 1082-3654.

Wilderman S J, Rogers W, Knoll G F & Engdahl J C 1998 Fast algorithm for list mode back-projection of Compton scatter camera data *IEEE Transactions on Nuclear Science* **45**(3), 957–962.

Xu D & He Z 2005 Gamma-ray energy-imaging integrated deconvolution *in* 'IEEE Nuclear Science Symposium Conference Record, 2005' Vol. 2 IEEE pp. 882–886.

Yabu G, Yoneda H, Orita T, Takeda S, Caradonna P, Takahashi T, Watanabe S & Moriyama F 2021 Tomographic imaging by a Si/CdTe Compton camera for ^{111}in and ^{131}i radionuclides *IEEE Transactions on Radiation and Plasma Medical Sciences* **6**(5), 592–600.

Zhang J, Xiao X, Chen Y, Zhang B, Ma X, Ai X & Li J 2023 A portable three-layer Compton camera for wide-energy-range gamma-ray imaging: Design, simulation and preliminary testing *Sensors* **23**(21), 8951.

Zoglauer A, Boggs S E, Galloway M, Amman M, Luke P N & Marc Kippen R 2011 Design, implementation, and optimization of MEGAlib's image reconstruction tool Mimrec *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **652**(1), 568–571.