Comparing One-step and Two-step Scatter Correction and Density Reconstruction in X-ray CT

Alexander N. Sietsema[a], Michael T. McCann[b], Marc L. Klasky[b], and Saiprasad Ravishankar[c]

[a]Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA
[b]Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA
[b]Department of Computational Mathematics, Science and Engineering and the Department of Biomedical Engineering, Michigan State University, East Lansing, MI 48824, USA

ABSTRACT

In this work, we compare one-step and two-step approaches for X-ray computed tomography (CT) scatter correction and density reconstruction. X-ray CT is an important imaging technique in medical and industrial applications. In many cases, the presence of scattered X-rays leads to loss of contrast and undesirable artifacts in reconstructed images. Many approaches to computationally removing scatter treat scatter correction as a preprocessing step that is followed by a reconstruction step. Treating scatter correction and reconstruction jointly as a single, more complicated optimization problem is less studied. It is not clear from the existing literature how these two approaches compare in terms of reconstruction accuracy. In this paper, we compare idealized versions of these two approaches with synthetic experiments. Our results show that the one-step approach can offer improved reconstructions over the two-step approach, although the gap between them is highly object-dependent.

Keywords: Computed tomography, computational imaging, density estimation, scatter correction, model-based iterative reconstruction

1. INTRODUCTION

The presence of scattered X-rays presents a challenge for X-ray computed tomography (CT) imaging systems. For example, in the context of medical cone-beam CT, scatter causes a loss in soft-tissue contrast and artifacts such as cupping, streaks, bars, and shadows.[1] The same artifacts appear in nondestructive testing applications of X-ray CT, where they can interfere with subsequent quantification tasks.[2] There is a large body of work on preventing scatter using hardware and on correcting it using software; see[1,3] for a review.

Hardware approaches to scatter correction include collimation (blocking unwanted X-rays at the source, thereby preventing them from contributing to scatter) or increasing the distance between the source and detector, which reduces the amount of scattered radiation reaching the detector. Among software approaches to scatter correction, a key distinction is whether scatter correction happens as a preprocessing step before CT reconstruction or jointly with CT reconstruction. In the former case, which we term *two-step* reconstruction, scatter is typically modeled as a function of the direct (i.e., not scattered) radiograph. This model is used to remove scatter from the measured data and estimate the direct radiograph, which is subsequently used for reconstruction. In the latter case, which we term *one-step* reconstruction, a model of scatter is included in a model-based CT reconstruction algorithm.

In our literature review, we found that the two-step formulation is the more common approach, including among recent work.[4–7] A two-step approach is also implicitly assumed in works where only scatter correction is considered.[8] The popularity of the two-step approach may be because it usually involves solving two simple optimization problems (scatter correction and reconstruction) rather than a challenging joint problem.

The one-step approach appears less well-studied. The model-based reconstruction formulation in Ref. 9 includes a scatter term, but it is assumed to be known. The authors in a recent work[10] iteratively alternate between

---

Note: Alexander N. Sietsema is the corresponding author.

reconstruction and scatter estimation, but do not formulate a joint optimization problem. The review[1] describes a joint scatter correction and reconstruction approach, but only in general terms and without implementing it.

The main goal of this work is to compare one-step and two-step reconstruction approaches to find when, if ever, the added complexity of one-step reconstruction yields better results. To this end, we compare idealized one- and two-step algorithms on synthetic data in our studies. These experiments are intentionally simple: scatter is modelled using a convolution with a Gaussian kernel, noise is Gaussian, and the beam is monoenergetic; we believe this experimental setup includes many of the key features of real descattering and reconstruction while leaving out aspects that may obscure the difference between the one-step and two-step approaches, e.g., inaccuracies in the radiographic forward model and beam hardening.

In the following sections, we formulate the scatter correction and reconstruction problem, describe our one- and two-step algorithms, and present our experiments, results, and conclusions.

2. SCATTER CORRECTION AND RECONSTRUCTION

We focus on a model of 2D monoenergetic X-ray tomography that includes scatter. Given a (vectorized) density profile $\boldsymbol{\rho} \in \mathbb{R}^{N_1 N_2}$, our model of the total transmission $\boldsymbol{t}$ is

$$\boldsymbol{t} = \boldsymbol{s} + \boldsymbol{d}, \quad \boldsymbol{s} = \boldsymbol{K}\boldsymbol{d}, \quad \boldsymbol{d} = \exp(-\xi \boldsymbol{A}\boldsymbol{\rho}), \tag{1}$$

where $\boldsymbol{s}$ is the scattered signal, $\boldsymbol{d}$ is the direct (i.e., scatter-free) signal, $\boldsymbol{A} \in \mathbb{R}^{M \times N_1 N_2}$ is the X-ray transform, ξ represents the mass attenuation coefficient for a given material, $\exp(\cdot)$ is applied element-wise, and $\boldsymbol{K}$ represents linear convolution by a kernel $\boldsymbol{k}$, used to approximate scatter.

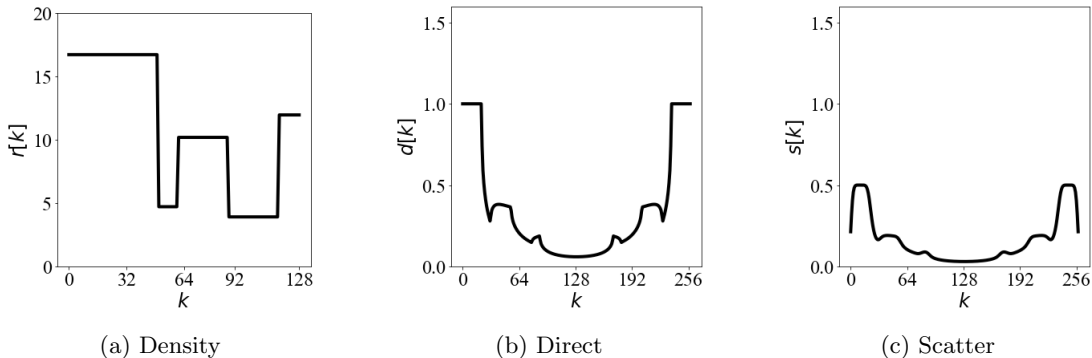(a) Density (b) Direct (c) Scatter

Figure 1: Density, direct, and scatter lineouts for profile 4 in our data set.

The choice to model scatter as a kernel convolved with the direct is common in the scatter correction literature.[7, 11–14] This provides a fast scatter model that is at least representative of models used in practice. As our main goal is to bring out the differences between the one-step and two-step formulations, we leave more complicated models for future investigation.

2.1 One-step vs. Two-step Scatter Correction and Reconstruction

The two-step method involves first solving a scatter correction problem that is followed by solving a density reconstruction problem. We formulate the first step, i.e. scatter correction, as

$$\boldsymbol{d}^* = \underset{\boldsymbol{d}}{\operatorname{argmin}} \|\boldsymbol{t} - (\boldsymbol{K}\boldsymbol{d} + \boldsymbol{d})\|_2^2, \tag{2}$$

where we minimize an ℓ_2 fit between the measured transmission and the model for it (i.e., direct + scatter) to account for noisy data. This step would correct for scatter in all the measured (one or multiple) CT views.

Following scatter correction, we invert the nonlinear part of (1) with the elementwise operation, $\boldsymbol{\rho}_A^*[m, n] = -\log(\boldsymbol{d}^*[m, n])/\xi$. Note that we would need to set any values where $\boldsymbol{d}^*[m, n]$ was less than or equal to zero to

zero, as the logarithm would be invalid there. We use the symbol $\boldsymbol{\rho}_A^*$ here because this quantity represents the areal density.[7]

After performing this scatter correction, we solve the following optimization problem to reconstruct the underlying density:

$$\boldsymbol{\rho}^* = \underset{\boldsymbol{\rho}}{\operatorname{argmin}} \|\boldsymbol{\rho}_A^* - \boldsymbol{A}\boldsymbol{\rho}\|_2^2 + \alpha R(\boldsymbol{\rho}), \tag{3}$$

where R is a regularization functional and α is a nonnegative parameter. In essence, this approach first estimates the direct $\boldsymbol{d}^*$ by removing scatter from the transmission $\boldsymbol{t}$, and then takes that estimate and uses it to reconstruct an estimate of the object density $\boldsymbol{\rho}^*$.

The main advantage of the two-step method is its simplicity: both the first and second step are well-studied formulations of linear inverse problems that can be readily solved with standard algorithms. The scatter correction step involves linear least squares (with possible constraints) for which there are many good algorithms, e.g., a fixed-point method such as Jacobi iteration,[15] the conjugate gradient method[15] or ADMM.[16] The reconstruction step can be solved efficiently using ADMM.

One-step scatter correction and reconstruction involves jointly optimizing over the entire forward model rather than first optimizing for the direct radiograph. This amounts to solving the following optimization problem:

$$\boldsymbol{\rho}^* = \underset{\boldsymbol{\rho}}{\operatorname{argmin}} \|\boldsymbol{t} - (\boldsymbol{K} + \boldsymbol{I})\exp(-\xi\boldsymbol{A}\boldsymbol{\rho})\|_2^2 + \alpha R(\boldsymbol{\rho}), \tag{4}$$

where R is again a regularization functional. Intuitively, the one-step method has the benefit of not relying on the estimation of the direct: in the two-step method, our overall estimate of the density is limited by the estimate of $\boldsymbol{d}^*$ we obtain from solving (2). However, the one-step optimization could be more challenging depending on the complexity of the entire forward model.

2.2 Implementation of One-step and Two-step Methods

In our implementation of the one-step and two-step methods described above, we consider imaging a spherically symmetric, single-material object, parameterized by its radial profile $\boldsymbol{\rho} \in \mathbb{R}^N$. This simple model captures the important elements of X-ray reconstruction with scatter, while remaining fast to optimize; similar models find application in nondestructive testing[17] and have been used for developing scatter correction methods.[7] As a result, our operator $\boldsymbol{A}$ is the forward Abel transform and is followed by spinning the 1D signal into a 2D image on which the convolution (for scatter) is applied. For an example of data generated with this model, see Figure 1.

For regularization, (R in (3) and (4)) we use total variation on the profile, i.e., $R(\boldsymbol{\rho}) = \sum_{k=1}^{n} |\boldsymbol{\rho}[k] - \boldsymbol{\rho}[k-1]|$.

In both approaches, we solve the underlying optimization problems using the LBFGS algorithm in PyTorch.[18] Note that PyTorch automatically handles non-differentiability issues. Additionally, due to the poor conditioning of the Abel matrix $\boldsymbol{A}$, we use the separable quadratic surrogate (SQS) preconditioning as follows:

$$\boldsymbol{P} = \operatorname{diag}\left(\boldsymbol{A}^\top \boldsymbol{A}\mathbf{1}\right)^{-1}, \tag{5}$$

where $\mathbf{1} \in \mathbb{R}^N$ denotes a vector of ones. We normalize $\boldsymbol{A}^\top \boldsymbol{A}\mathbf{1}$ prior to computing (5). This preconditioner is used for performing the one-step optimization and the second step of the two-step optimization. We apply this preconditioning by premultiplying $\boldsymbol{\rho}$ by $\boldsymbol{P}$ prior to optimizing. For the one-step method, this then becomes the preconditioned optimization formulation

$$\boldsymbol{\rho}' = \underset{\boldsymbol{\rho}}{\operatorname{argmin}} \|\boldsymbol{t} - (\boldsymbol{K} + \boldsymbol{I})\exp(-\xi\boldsymbol{A}\boldsymbol{P}\boldsymbol{\rho})\|_2^2 + \alpha \sum_{k=1}^{n} |\boldsymbol{P}\boldsymbol{\rho}[k] - \boldsymbol{P}\boldsymbol{\rho}[k-1]|, \tag{6}$$

where the solution is recovered via $\boldsymbol{\rho}^* = \boldsymbol{P}\boldsymbol{\rho}'$; a similar formulation is used for the two-step problem.

3. EXPERIMENTS AND RESULTS

We compare the one-step and two-step algorithms by scatter correcting and reconstructing ten synthetically generated transmissions. In all experiments, we quantify performance by taking the root mean square error (RMSE) between the ground truth density and the reconstructed density for each algorithm.
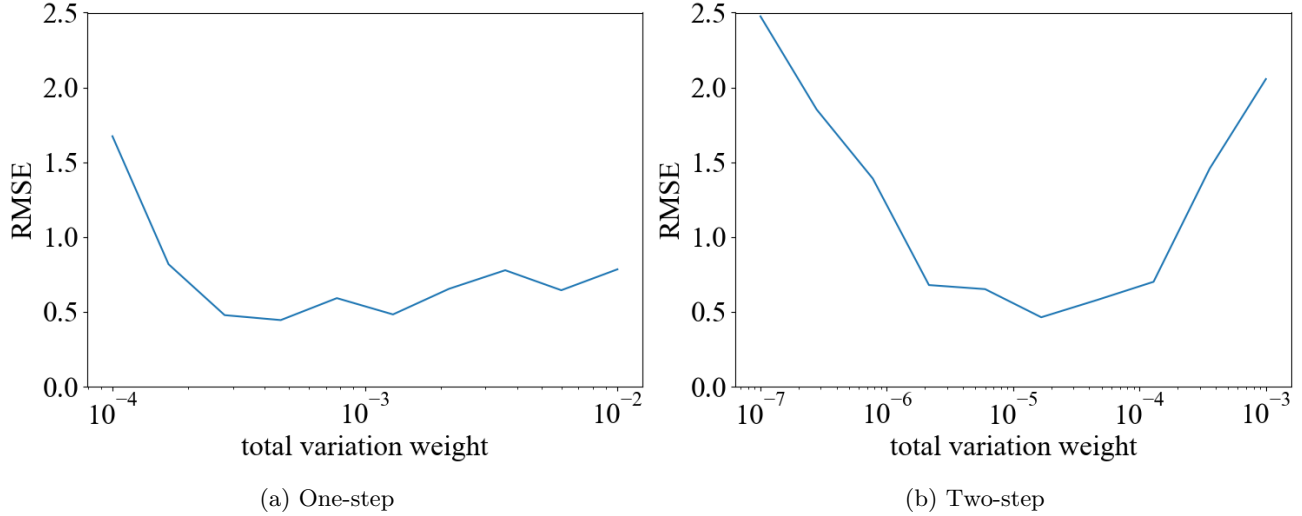
| (a) One-step | (b) Two-step |

Figure 2: Reconstruction accuracy (RMSE) as a function of the total variation weight (α in (3) and (4)). The optimal learning rate was tuned independently for each weight value. The minima achieved are used in later fitting.

3.1 Data Generation

To generate test objects, we first randomly selected indices in 1D at which shells start and end, assuming a maximum radius of $N = 129$ pixels of the object. We converted these indices into a piecewise-constant profile with steps at each shell boundary. Finally, we picked shell densities (in $\mathbb{R}$) uniformly in the range (0, 20), and assigned each shell a density. These 1D radial profiles were spun to create 2D images (representing slice of 3D volume) as part of the forward projection process.[19] The spinning process slightly cropped the images to avoid edge effects.

In order to generate transmissions, we applied the forward model in (1), choosing $\xi = 1 \times 10^{-3}$ and fixing a Gaussian scatter kernel. In our implementation, this kernel is a three-fold convolution with a 7×7 Gaussian blur with standard deviation 1.5 pixels. Spinning occurs after $\boldsymbol{A\rho}$ is computed. Finally, Gaussian noise is added in the end with $\mu = 0, \sigma = 3 \times 10^{-2}$, and negative values in the transmission are set to zero (non-physical). See Figure 1 for an example of the density, direct, and scatter profiles.

3.2 Implementation Details

All experiments were performed in Python using simulated data, and optimization is performed using the PyTorch package. The one-step algorithm was run with a learning rate (step size of Wolfe line search) of 3×10^{-2} and a total variation weight of 1×10^{-3} with 20 iterations. The two-step algorithm used a learning rate of 1.0 and no total variation with 10 iterations for the first step, and a learning rate of 1×10^{-2} and a total variation weight of 7×10^{-4} with 20 iterations for the second step. All parameters were optimized by performing a grid search over possible combinations of learning rate and total variation parameters. See Figure 2 for the results of the grid search. The forward Abel transform was the Hansenlaw method[20] from the PyAbel Python package.[19]

3.3 Results

Results for each of the ten profiles are summarized in Table 1. Overall, the one-step algorithm outperforms the two-step algorithm, with a median RMS error of 1.548, compared to 2.575 for the latter. Two example reconstructions are shown in Figure 3. Qualitatively, the two-step method can be quite noisy, especially near the center of the reconstruction, despite the regularization. However, the two-step model may reconstruct thin shells better (Figure 3b), although this benefit is on the whole negligible.

Both methods tend to struggle with profiles with high-density (densities above 8) shells near the center. This is likely due to the conditioning effects of the Abel matrix, which the SQS preconditioning is only partially able

Table 1: One-step vs. two-step reconstruction performance.

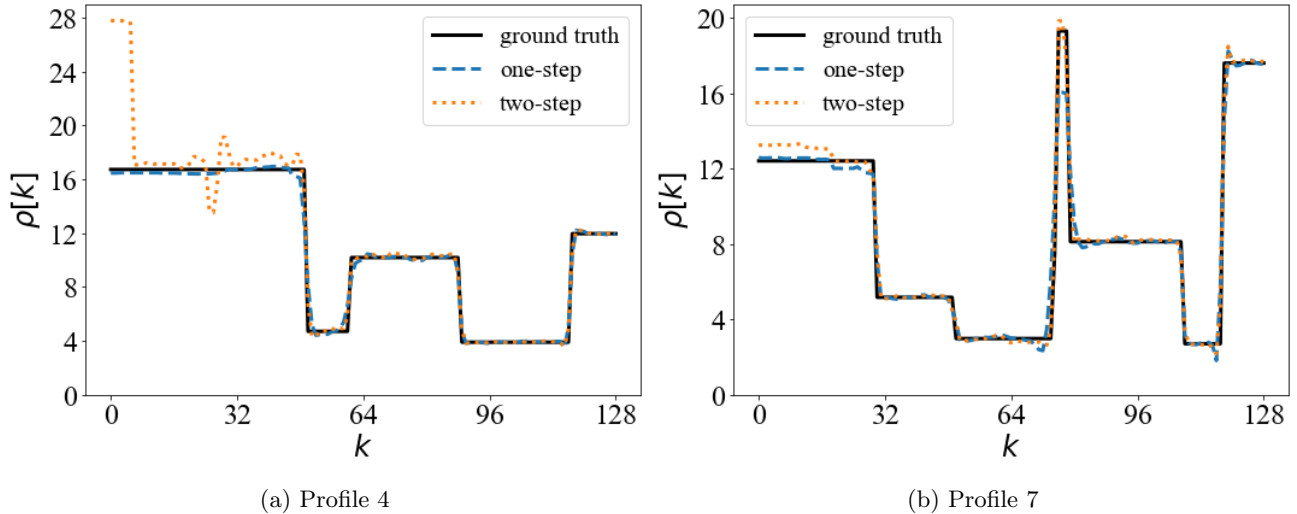| Profile	One-step RMSE	Two-step RMSE
1	**0.827**	1.445
2	**2.077**	9.242
3	**2.021**	9.552
4	**0.568**	2.468
5	**1.210**	2.683
6	**2.488**	3.071
7	0.853	**0.552**
8	**0.422**	0.623
9	**1.887**	2.397
10	**2.305**	3.491

(a) Profile 4   (b) Profile 7

Figure 3: Comparison of one-step and two-step reconstruction results on two profiles.

to solve. Uniformly high-density profiles can also cause some problems in both methods (e.g., profiles 2 and 3). This may be an artifact of our noise model, as denser objects produce smaller transmissions, which results in a low signal-to-noise-ratio when noise with a constant variance is added.

Finally, we summarize convergence results, where we plot the data-fidelity terms in (3) and (4) over the LBFGS iterations. From Figure 4, we see that both algorithms converge quickly. We omitted the first step for the two-step method here since it converges nearly instantly and we plotted for only the one-step method and the second step of the two-step method.

4. CONCLUSIONS

In this paper, we performed an empirical comparison of one-step and two-step X-ray CT scatter correction and reconstruction. Our experiments showed that the one-step approach can demonstrate significant improvements over the common two-step approach. While certainly not exhaustive, these experiments suggest that the added complexity of the one-step method may be worth it. Our future work in this area would extend these experiments to more complicated regimes, including using more complicated scatter estimation models,[7] polyenergetic spectra, and multiple materials. The same comparisons could be also run on more realistic synthetic data (e.g., generated from particle transport simulations as in recent work[7]) to validate the efficacy of one-step descattering in a setting where the scatter model used during reconstruction does not perfectly match the scatter in the data. Finally, we aim to run similar comparisons on real, experimental data.

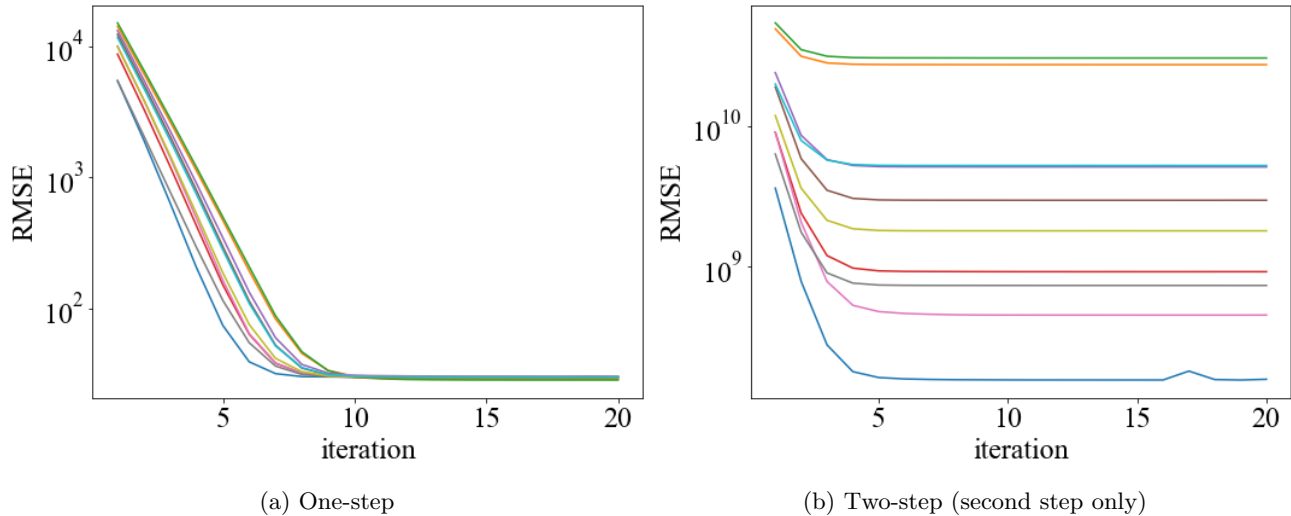(a) One-step   (b) Two-step (second step only)

Figure 4: Comparison of one-step and two-step (second step only) convergence rates for each of the ten profiles. Satisfactory convergence is demonstrated in all cases. Note that the two-step loss is calculated in the areal density space, so its values are much higher than the one-step method's.

REFERENCES

[1] Rührnschopf, E.-P. and Klingenbeck, K., "A general framework and review of scatter correction methods in X-ray cone-beam computerized tomography. part 1: Scatter compensation approaches," *Med. Phys.* **38**, 4296–4311 (June 2011).

[2] Lifton, J. J., Malcolm, A. A., and McBride, J. W., "An experimental study on the influence of scatter and beam hardening in X-ray CT for dimensional metrology," *Meas. Sci. Technol.* **27**, 015007 (Dec. 2015).

[3] Rührnschopf, E.-P. and Klingenbeck, K., "A general framework and review of scatter correction methods in cone beam CT. part 2: Scatter estimation approaches," *Med. Phys.* **38**, 5186–5199 (Aug. 2011).

[4] Bhatia, N., Tisseur, D., and Létang, J. M., "Convolution-based scatter correction using kernels combining measurements and monte carlo simulations," *J Xray Sci Technol.* **25**, 613–628 (2017).

[5] Maier, J., Sawall, S., Knaup, M., and Kachelrieß, M., "Deep scatter estimation (DSE): Accurate real-time scatter estimation for X-ray CT using a deep convolutional neural network," *J. Nondestruct. Eval.* **37** (July 2018).

[6] Nomura, Y., Xu, Q., Shirato, H., Shimizu, S., and Xing, L., "Projection-domain scatter correction for cone beam computed tomography using a residual convolutional neural network," *Med Phys* **46**, 3142–3155 (jun 2019).

[7] McCann, M. T., Klasky, M. L., Schei, J. L., and Ravishankar, S., "Local models for scatter estimation and descattering in polyenergetic X-ray tomography," *Opt. Express* **29**(18), 29423–29438 (2021).

[8] Tisseur, D., Bhatia, N., Estre, N., Berge, L., Eck, D., and Payan, E., "Evaluation of a scattering correction method for high energy tomography," *EPJ Web Conf.* **170**, 06006 (2018).

[9] Nuyts, J., Man, B., Fessler, J. A., Zbijewski, W., and Beekman, F. J., "Modelling the physics in the iterative reconstruction for transmission computed tomography," *Phys. Med. Biol.* **58**, R63–R96 (June 2013).

[10] Mainegra-Hing, E. and Kawrakow, I., "Fast monte carlo calculation of scatter corrections for CBCT images," *J. Phys.: Conf. Ser* **102**, 012017 (Feb. 2008).

[11] Sun, M. and Star-Lack, J. M., "Improved scatter correction using adaptive scatter kernel superposition," *Phys. Med. Biol* **55**, 6695–6720 (Oct. 2010).

[12] Suri, R. E., Virshup, G., Zurkirchen, L., and Kaissl, W., "Comparison of scatter correction methods for CBCT," in [*SPIE Med. Imaging*], **6142**, 614238, International Society for Optics and Photonics (2006).

[13] Ohnesorge, B., Flohr, T., and Klingenbeck-Regn, K., "Efficient object scatter correction algorithm for third and fourth generation CT scanners," *Eur. Radiol* **9**(3), 563–569 (1999).

[14] Love, L. A. and Kruger, R. A., "Scatter estimation for a digital radiographic system using convolution filtering," *Med. Phys* **14**(2), 178–185 (1987).

[15] Shewchuk, J. R., "An introduction to the conjugate gradient method without the agonizing pain," tech. rep., Carnegie Mellon University (1994).

[16] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.* **3**, 1–122 (Jan. 2011).

[17] Asaki, T. J., Campbell, P. R., Chartrand, R., Powell, C. E., Vixie, K. R., and Wohlberg, B. E., "Abel inversion using total variation regularization: applications," *Inverse Probl. Sci. Eng.* **14**, 873–885 (Dec. 2006).

[18] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S., "Pytorch: An imperative style, high-performance deep learning library," in [*Advances in Neural Information Processing Systems*], Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., eds., **32**, Curran Associates, Inc. (2019).

[19] Gibson, S., Hickstein, D. D., Yurchak, R., Ryazanov, M., Das, D., and Shih, G., "Pyabel/pyabel: v0.8.4," (2021).

[20] Hansen, E. W. and Law, P., "Recursive methods for computing the Abel transform and its inverse," *J. Opt. Soc. Am. A: Opt. Image Sci. Vis.* **2**, 510 (Apr. 1985).