Towards robust quantitative photoacoustic tomography via learned iterative methods

Anssi Manninen, Janek Gröhl, Felix Lucka, and Andreas Hauptmann *Senior Member, IEEE*

***Abstract*—Photoacoustic tomography (PAT) is a medical imaging modality that can provide high-resolution tissue images based on the optical absorption. Classical reconstruction methods for quantifying the absorption coefficients rely on sufficient prior information to overcome noisy and imperfect measurements. As these methods utilize computationally expensive forward models, the computation becomes slow, limiting their potential for time-critical applications. As an alternative approach, deep learning-based reconstruction methods have been established for faster and more accurate reconstructions. However, most of these methods rely on having a large amount of training data, which is not the case in practice. In this work, we adopt the model-based learned iterative approach for the use in Quantitative PAT (QPAT), in which additional information from the model is iteratively provided to the updating networks, allowing better generalizability with scarce training data. We compare the performance of different learned updates based on gradient descent, Gauss-Newton, and Quasi-Newton methods. The learning tasks are formulated as greedy, requiring iterate-wise optimality, as well as end-to-end, where all networks are trained jointly. The implemented methods are tested with ideal simulated data as well as against a digital twin dataset that emulates scarce training data and high modeling error.**

***Index Terms*—Quantitative photoacoustic tomography, nonlinear inverse problems, deep Learning, learned iterative methods, convolutional neural networks, digital twin.**

I. Introduction

During recent decades, quantitative photoacoustic tomography (QPAT) has received increasing interest as a new, non-ionizing, *in vivo* imaging modality due to its ability to produce quantitative images of optical parameters with higher resolution than purely optical modalities by combining the propagation of non-scattering ultrasound and optical contrast [1]–[3]. In a typical QPAT setup, short pulses of light are emitted to a target region, where the light scatters and is absorbed, causing the emergence of ultrasound waves, which are measured at the boundary. The measurements carry information on chromophores that absorbed the light and can be used to form an inverse problem to reconstruct quantitative chromophore concentrations [4], [5], revealing functional tissue properties such as blood oxygenation.

The inverse problem of QPAT is usually formulated as two separate problems: the acoustical and optical inversion. In the acoustic inverse problem, the objective is to recover the initial pressure field, which primarily provides qualitative structural information that can already carry diagnostic meaning in clinical applications, see, e.g. [6], [7]. Various established methods exist to solve it [1], [8], and we assume it to be solved in this work. While the acoustic waves propagate nearly scattering-free, scattering of photons cannot be neglected when solving the optical inverse problem. Therefore, the measured data is modeled to be nonlinearly dependent on the unknown optical parameters. For biological tissues, the commonly accepted model for light transport is the radiative transfer equation (RTE) [9], which often has to be approximated for computational efficiency. The most popular of these is the diffusion approximation (DA) [10], [11], which is most accurate in highly diffusive regions [10]–[12]. Alternatively, a popular choice is to stochastically estimate the photon transport with Monte Carlo methods [13].

The solution of the optical nonlinear inverse problem with a light propagation model can be formulated as an optimization problem. This nonlinear optimization problem is popularly solved by using information from second-order derivatives iteratively [14]. However, computing or storing the Hessian is often impractical for a large number of unknowns and hence it is often approximated with quasi-Newton or Newton type methods [14], [15].

The reconstruction of the weakly data-dependent and highly ill-posed scattering is the biggest challenge. To overcome it, one could assume constant scattering values and only estimate the mildly ill-posed absorption [16]. This can lead to poor absorption estimates if the assumed scattering is inaccurate. But even simultaneous estimation of absorption and scattering from a single illumination is non-unique. To guarantee uniqueness, multiple sources with different illumination patterns or multiple wavelengths need to be utilized [10], [17].

A. Deep learning-based techniques for QPAT

Solving the nonlinear optical problem with classical model-based techniques comes with difficulties [8], [18] that we illustrate in Fig. 1). Recently, data-driven methods have been studied to overcome these. However, in contrast to the acoustic

AM and AH were supported in part by the Research council of Finland (Project No. 353093, Finnish Centre of Excellence in Inverse Modeling and Imaging; and Project No. 359186, Flagship of Advanced Mathematics for Sensing Imaging and Modelling; Projects No. 338408, 359915, Academy Research Fellow)

Codes published at: https://github.com/AnsManni/Learned_iterative_QPAT

A. Manninen is with the Research Unit of Mathematical Sciences, University of Oulu, Oulu, Finland.

J. Gröhl is with ENI-G, a Joint Initiative of the University Medical Center Göttingen and the Max Planck Institute for Multidisciplinary Sciences, Göttingen, Germany

F. Lucka is with the Centrum Wiskunde & Informatica, Amsterdam, The Netherlands.

A. Hauptmann is with the Research Unit of Mathematical Sciences, University of Oulu, Oulu, Finland and with the Department of Computer Science, University College London, London, United Kingdom.

problem, where a wide range of methods have been studied [19], [20], approaches for the optical problem have been more limited due to the computational burden of including the imaging model.

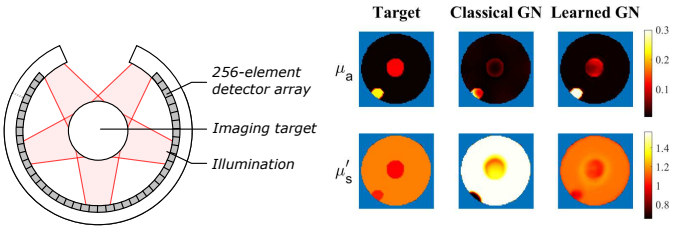

Fig. 1. Left: Setup of the photoacoustic hardware. Right: Absorption (μ_a) and reduced scattering (μ_s') reconstructions of a digital twin sample (left) after the optical inversion. Due to the huge modeling error, the classical Gauss-Newton solver (mid) is not able to reconstruct the targets robustly, whereas its deep-learned assisted variant (right) is.

Deep learning techniques developed for the optical problem of QPAT have focused primarily on learning a direct reconstruction, e.g., to fully learn a single-step reconstruction operator that takes multiple initial pressure images as input and directly and computationally efficiently estimates absorption coefficients [21], chromophore concentrations [22], [23] or blood oxygen saturation [24], [25]. A downside of fully learned operators is that they do not incorporate an explicit imaging model, and the learning task is highly complex. They thus require a large quantity of training samples, and generalization is limited [26]. For clinical applications, limited sample acquisition and missing ground truth for training remain a problem. Consequently, with only a handful of experimental samples the fully learned reconstruction methods are not likely to generalize well.

The idea of model-based learned methods is to incorporate an imaging model to reduce the complexity of the learning task. However, in contrast to the fully learned approach, modeling inaccuracies can now affect the result. Some recent model-based work has concentrated on accelerating classical iterative solvers by replacing the hand-crafted light transport model by a learned Fourier neural operator [27], [28]. In the popular model-based learned iterative reconstruction methods, the classically computed model information and the neural networks updating the unknown values are iteratively repeated K times, analogous to classical iterative optimization solvers [29]. While the repeated model evaluation leads to longer run times, they have been shown to outperform single-step reconstructions in quality and generalizability with smaller network sizes [30]–[32]. However, their application to nonlinear inverse problems is severely limited due to the additional model complexity. Specifically, the approach has not been used for QPAT and is limited to related modalities such as diffuse optical tomography [33] and electrical impedance tomography [34]–[36], often using a cheaper *greedy* training regime of training each updating network (iterate) separately instead of the preferred *end-to-end* training of all iterates usually employed in linear inverse problems [29].

B. Contributions of this work

The scientific literature on model-based iterative learned methods for nonlinear inverse problems remains scarce and several questions remain:

1) Influence of the step directions, i.e., of the underlying iterative optimization method.
2) Influence of the number of the learned iteratively updating networks on the reconstructions
3) Performance under large modeling errors
4) Training regimes: Greedy vs. end-to-end

We provide insights into the above by investigating the performance of the model-based learned iterative method by prototyping it for the QPAT problem. QPAT is ideal for our purposes due to its nonlinear nature, and because there is a general need for fast and robust methods. Importantly, the optical problem involves solving the mildly ill-posed absorption and severely ill-posed scattering, demonstrating the performance on both types of unknowns. As the model-based iterative method requires repeated use of the light propagation model, the DA is used for training the networks, providing a possibility to investigate the model error compensation capabilities.

To gain insight into how reconstruction accuracy and computational times differ between network realizations, we test gradient descent, Gauss-Newton, and quasi-Newton rank-1-based learned updates with an increasing number of learned updating blocks. An additional focus is to compare performances between greedily and end-to-end trained networks. While all previous work on nonlinear inverse problems has utilized computationally cheaper greedy training, we also implemented end-to-end training for the first time and will examine if there is a significant performance difference between the two schemes. To compare each implemented method, we first consider a simulated 2D study with a large number of samples and low modeling error. To investigate the performances closer to experimental conditions, we utilize a 2D digital twin [26], [37] with a scarce amount of training data, and a high modeling error when using the DA.

II. Quantitative Photoacoustic Tomography

In QPAT, the acoustic inversion is followed by the optical inversion, for which the data, absorbed energy density $h(r)$ for region $r \in \Omega \subset \mathbb{R}^n$ ($n = 2$ or 3), is acquired from the acoustical problem via the relation $h(r) = \frac{p_0(r)}{\hat{\Gamma}(r)}$, where p_0 is the initial pressure and $\hat{\Gamma}$ is Grüneisen parameter. In this paper, only the optical inverse problem is covered. Therefore, we assume that the initial pressure field is already solved and divided by the known Grüneisen parameter, yielding an estimate of the absorbed optical energy density h.

A. Forward model of the optical problem

For the optical inversion, we consider a discretization of the domain with P non-overlapping finite elements and N grid coordinates. The optical parameters in Ω are then represented in the basis

$$\mu_a(r) \approx \sum_{t=1}^{N} \mu_{a_t} \varphi_t(r), \quad \mu_s(r) \approx \sum_{t=1}^{N} \mu_{s_t} \varphi_t(r), \qquad (1)$$

where $\mu_a(r)$ is the absorption field, $\mu_s(r)$ is the scattering field and $\varphi_k(r)$ is the finite element basis function. The absorbed optical energy density is linked to the scattering and absorption coefficients via the relation

$$h(r) = \mu_a(r)\Phi(\mu_a(r), \mu_s(r)), \tag{2}$$

where $\Phi(r)$ is the fluence in $r \in \Omega$. The fluence depends on the absorption $\mu_a(r)$ and scattering $\mu_s(r)$ distributions in Ω in a nonlinear way. For the fluence, the same basis is used as for the optical coefficients in (1). To solve the dependency between the optical parameters and the fluence in (2), a light propagation model is introduced. The commonly accepted model for biological medium is the RTE, originating from Maxwell's equations [38]. For QPAT, the time-independent RTE can be found in the A section of the supplementary file.

Due to the high computational cost of solving the RTE, a diffusion approximation (DA) has been popularly applied in biomedical applications. As in this work (see Sec. III), we will numerically test the model-based learned iterative methods, which evaluate the light propagation model multiple times in each training step. Therefore, it is practical to consider the DA for light propagation. However, the DA can only predict the photon fluence accurately if the domain is approximately diffusive, that is, the scattering needs to be significantly larger than the absorption, and the domain size should be sufficiently large compared to the wavelength used [9].

By choosing Henyey–Greenstein scattering function [39] and assuming a diffusive region Ω, the diffusion approximation is given by (see e.g. [9])

$$-\nabla \cdot \kappa(r)\nabla\Phi(r) + \mu_a(r)\Phi(r) = q_0(r), \quad r \in \Omega,$$

where q_0 is the light source in Ω, the parameter $\kappa(r) = (n(\mu_a(r) + \mu_s'(r)))^{-1}$ is the diffusion coefficient where $\mu_s' = (1-g)\mu_s$ is the reduced scattering coefficient with anisotropy parameter $-1 < g < 1$. A boundary condition for the DA can be formed by assuming that the total inward-directed photon current is zero at the boundary. Moreover, including the possible refraction mismatch between Ω and its surrounding medium, the boundary condition for DA can be derived to be [9]

$$\Phi(r) + \frac{1}{2\gamma}\kappa(r)A\frac{\partial\Phi(r)}{\partial\hat{n}} = \begin{cases} \frac{I(r)}{\gamma}, & r \in \rho \\ 0, & r \in \partial\Omega\backslash\rho, \end{cases},$$

where $\hat{n}$ is the outward unit normal vector, $I(r)$ is the boundary photon flux at the source positions $\rho \in \partial\Omega, \gamma$ is a dimension-dependent constant with values $1/\pi$ in $\mathbb{R}^2$ and 1/4 in $\mathbb{R}^3$. The parameter A accounts for the internal reflection at the boundary $\partial\Omega$. If no boundary refraction occurs, A corresponds to 1.

To numerically solve the DA, the finite element approximation (for details, see [40], [41]) is applied. Applying the FE approximation in the chosen basis φ_t yields a matrix form for the DA from which the fluence can be computed as

$$\Phi = (M + C + R)^{-1}Q. \tag{3}$$

The exact forms of matrices M, C, R, and vector Q can be found in the B section of the supplementary file. In the case of RTE, a similar FE approximation can be derived (see, e.g. [41]), but requiring the discretization of the angular directions.

B. Variational methods for solving the optical inversion

To define the inverse problem for I illuminations, let us consider a discrete observation model of the form

$$h = F(x) + e, \tag{4}$$

where $h \in \mathbb{R}^{N\times I}$ is the absorbed optical energy density at locations r_t, $t = 1, \ldots, N$, $x \in \mathbb{R}^l$ contains the unknown scattering and absorption parameters, $F : \mathbb{R}^l \to \mathbb{R}^{N\times I}$ is a nonlinear forward operator mapping the optical parameters x to the data space, and $e \in \mathbb{R}^{N\times I}$ is the additive measurement noise. In this work, the operator F corresponds to the RHS of the equation (2), where the fluence Φ is obtained from the finite element solution of the DA in the form of (3). The inverse problem is then to recover the set of unknown optical parameters x from the measured noisy data h. The number l of unknowns x depends on the type of the measurement setup. For multiple-source illumination, the optical values are the same for each illumination ($l = 2N$), whereas for the multi-wavelength setup, scattering and absorption are simultaneously solved for every used wavelength $\lambda \in \mathbb{R}$. However, to guarantee that a set of optical values uniquely describes data h, it is necessary to approximate the wavelength dependency of the scattering as $\mu_s(r, \lambda) \approx f(\lambda)a(r)$. Therefore, while multiple absorption reconstructions are solved, only $a \in \mathbb{R}^N$ is solved and used to compute scattering for each wavelength via $f(\lambda)$. The wavelength dependency in biological tissue has been approximated with an exponentially decaying function such as $f(\lambda) = \lambda^{-b}$, where $b \in \mathbb{R}$ is determined from experiments [42].

A well-established framework to solve the inverse problem of the nonlinear observation model (4) is the variational approach. In the variational approach for QPAT, the optical parameters x are solved by minimizing the functional

$$\mathcal{E}(x, h) = \frac{1}{2}\|L_e(h - F(x))\|_2^2 + \alpha\Psi(x), \tag{5}$$

where L_e is the weighting matrix, α is the regularization parameter, adjusting the importance of the regularizer Ψ, which encodes the prior information about x. By using the regularizer $\Psi(x)$, the solution space of x can be restricted such that smooth or sparse solutions are promoted.

A popular approach to minimize the functional (5) is to use an iterative optimization method of the form

$$x^{(k+1)} = x^{(k)} + \beta_k p_k \left(x^{(k)}, h\right),$$

where p_k is the update direction and β_k is the step length. For a differentiable functional $\mathcal{E}$, the update direction can utilize information from the first-order derivatives in the form of the gradient descent (GD) update

$$x^{(k+1)} = x^{(k)} + \beta_k \nabla\mathcal{E}\left(x^{(k)}, h\right). \tag{6}$$

or second-order derivatives, in the form of a Newton update

$$x^{(k+1)} = x^{(k)} + \beta_k H_k^{-1}\nabla\mathcal{E}\left(x^{(k)}, h\right), \tag{7}$$

where H_k is the Hessian of $\mathcal{E}$ at $x^{(k)}$. The initial optical values $x^{(0)}$ need to be set according to prior knowledge from experiments.

When the regularizer is assumed to be differentiable, the gradient of (5) can be written as

$$\nabla\mathcal{E}(x^{(k)}, h) = J_k^T W_e(h - F(x^{(k)})) + \alpha\nabla\Psi(x^{(k)}),$$

where J_k is Jacobian of the F at $x^{(k)}$ and $W_e = L_e^T L_e$. Usually, it is not practical to compute the exact second derivatives of F, and instead, an approximation $J_k^{\mathrm{T}} W J_k$ is used for the Hessian. The Newton update (7) then becomes the Gauss-Newton (GN) update

$$x^{(k+1)} = x^{(k)} + \beta_k(J_k^{\mathrm{T}} W_e J_k + H_\Psi)^{-1}\nabla\mathcal{E}\left(x^{(k)}, h\right), \quad (8)$$

where H_Ψ is the Hessian of the regularizer at $x^{(k)}$. Alternatively, the inverse of the Hessian H_k^{-1} can be approximated with quasi-Newton methods [14]. In these methods, the approximations of H_k^{-1} try to encapsulate the curvature information of the exact Hessian on $x^{(k)}$ based on the gradients.

As the used FE approximation of the DA and hence $F(x)$ is differentiable, the differentiability of the whole functional $\mathcal{E}$ depends on the type of regularizer$\Psi(x)$ used. A popular, differentiable regularizer is the weighted ℓ^2-norm

$$\Psi(x) = \frac{1}{2}\|L_x(x - x_\mu)\|_2^2, \quad (9)$$

with weighting matrix $L_x \in \mathbb{R}^{n\times n}$ and mean $x_\mu \in \mathbb{R}^l$. The gradient $\nabla\Psi(x^{(k)})$ is then simply $L_x^T L_x(x - x_\mu)$. The weighting matrix L_x can be used to introduce spatial correlations between the nodes, for example, by using the Ornstein-Uhlenbeck process

$$(\Gamma_x)_{p,t} = \sigma^2 \exp\left(-\frac{\|r_p - r_t\|}{\ell}\right), \quad (10)$$

where r_p and r_t are the locations of the pth and tth nodes, σ^2 is the variance, and ℓ is the characteristic length scale controlling the spatial decay of the correlation. The weighting matrix L_x is then the Cholesky factor of the inverse Γ_x^{-1}. An example of a non-smooth regularizers is given by the (discretized) total variation (TV),

$$TV(x) = \sum_{e=1}^{E} d_k \left|\boldsymbol{x}_{p(e)} - \boldsymbol{x}_{t(e)}\right|,$$

where d_k is the length of the kth edge in the mesh used, between nodes x_p and x_t, and E is the total number of edges in the mesh. The gradients for the update rules (6)-(8) are now not defined, and to minimize (5), more complex iterative methods need to be used.

For a practical use of the updates (6)-(8), the optical parameters need to be strictly positive between the iterations. To impose positivity, one can consider a logarithmically scaled solution space $\widetilde{x} = \log(x)$ and computing the respective Jacobian as $\widetilde{J}_k = J_k\mathrm{diag}(x^{(k)})$.

The convergence of the introduced classical iterative methods depends on the selection heuristics of the step length β_k. As the optimal step length can be challenging and computationally slow to find for a nonlinear problem (5), it is common to implement an inexact line search. Also, note that, for a nonlinear minimization problem, such as (5), there are no general methods to choose a suitable regularization parameter α, and it is often determined by sieving for feasible value.

In some cases, the absorbed energy density data can have a very large range of values, possibly slowing the convergence of the iterative methods. To overcome this, the log-scaled data space has been used to improve the convergence. For log-scaled data space, the Jacobian of $\log(F(x^{(k)}))$ is calculated as $\mathrm{diag}(1/F(x^{(k)}))J_k$, where J_k was the Jacobian of $F(x^{(k)})$.

C. Previous learned reconstructions for the optical inversion

Let us shortly summarize previous deep learning approaches to solve the quantitative problem. Majority if approaches have considered a fully learned approach, that means a neural network is trained to predict optical values, or often blood oxygenation (sO_2), directly from either acoustic measurement data or an intermediate reconstruction, such as reconstructed initial pressure p_0. More precisely, let $[p_0]_\lambda$ be a set of a multi-wavelength reconstructions, and denote by sO_2 a spatially resolved sO_2-map. Then the a neural network Λ_θ with parameters θ is trained to estimate $\Lambda_\theta([p_0]_\lambda) \approx \mathrm{sO}_2$, see for instance [20], [24].

Alternatively, the network can be trained to estimate the optical values instead of blood oxygenation. Here, we will consider as baseline for our proposed approach a network that predicts the optical values $[\mu_a, \mu_s']$ from the absorbed energy density, that is

$$\Lambda_\theta(h) \approx [\mu_a, \mu_s']. \quad (11)$$

While such a fully learned approach works well for in distribution data, it often generalizes insufficiently well for out-of-distribution data and especially experimental measurement data.

III. Model-based learned iterative methods for the optical problem

Classical variational methods to compute solutions of (5) often heavily depend on the prior information encoded in the regularizer Ψ and may need a large number of iterations to reach sufficient accuracy and hence limit time-critical applications. In the following, we will learn a reconstruction operator that encapsulates the prior information and learn effective update steps to improve reconstruction quality and speed.

For our application, in analogy to the classical updates (6)-(8), we can write the deep-learned iterations of the model-based learned iterative method for K iterations as

$$x^{(k+1)} = \Lambda_{\theta_k}\left(x^{(k)}, p_k\left(x^{(k)}, h\right)\right), \quad (12)$$

where $k = 0, \ldots, K-1$, and each iteration has an updating network Λ_{θ_k} with weights θ_k. The reconstruction operator is then defined as the result after K updates

$$\mathcal{R}_\theta(h) := x^{(K)}, \quad (13)$$

where $\theta = \{\theta_0, \ldots, \theta_{K-1}\}$. The commonly used architecture for the network Λ_{θ_k} is based on the residual network

ResNet [43], which consists of several multi-channel convolutional layers with activation functions and the last single-channel layer giving the update to the previous reconstruction $x^{(k)}$. As the updating networks Λ_{θ_k} in (12) repeatedly receive information based on the model, the network sizes can now be kept relatively small. A detailed description of architecture used for Λ_{θ_k} is given in Section III-C.

A. Choices for the update direction

In our work, we will consider the model-based learned iterative solver (12), where the directions p_k are chosen as GD from (6), i.e.,

$$p_k\left(x^{(k)}, h\right) = J_k^T W_e(h - F(x^{(k)})) \tag{14}$$

as well as GN (7) by

$$p_k\left(x^{(k)}, h\right) = (J_k^{\mathrm{T}} W_e J_k + H_\Psi)^{-1} \nabla \mathcal{E}\left(x^{(k)}, h\right). \tag{15}$$

Since the learned updates (12) harness the spatial information of the optical parameters, the classical regularizer $\Psi(x)$ is in practice only needed to stabilize the inversion in (7). For this purpose, any differentiable regularizer, such as the weighted l^2-norm (9) is suitable that allows using the update rule (7). In case of a non-smooth regularizer such as TV, the learned minimization problem (5) can be formulated as a learned primal-dual problem [30]. For the learned GD step (14), no regularizer was used to minimize the number of manually-set parameters.

The gradient direction provides less information for the networks, but is significantly cheaper to compute and store than the GN step. During network training, the step direction is computed thousands of times, and using the GN step can cause a crucial slowdown. However, it is not clear if the gradients provide enough information for the networks to achieve feasible convergence within a small number K of update layers Λ_{θ_k}. Therefore, in hopes of reducing the computational burden while maintaining fast convergence, we introduce the learned quasi-Newton iteration scheme with

$$p_k\left(x^{(k)}, h\right) = \bar{H}_k \nabla \mathcal{E}\left(x^{(k)}, h\right), \tag{16}$$

where $\bar{H}_k$ approximates the inverse of Hessian H_k^{-1}. The considered quasi-Newton approximation here is the symmetric-rank-1 (SR1) update

$$\bar{H}_{k+1} = \bar{H}_k + \frac{\left(s_k - \bar{H}_k y_k\right)\left(s_k - \bar{H}_k y_k\right)^T}{\left(s_k - \bar{H}_k y_k\right)^T y_k}, \tag{17}$$

where $s_k = x^{(k+1)} - x^{(k)}$ and $y_k = \nabla \mathcal{E}\left(x^{(k+1)}\right) - \nabla \mathcal{E}\left(x^{(k)}\right)$. The initial approximation $\bar{H}_0$ is usually chosen to be a scaled identity matrix, leading to an initial step direction equal to the GD step. To avoid too small denominator values in (17), a safeguard criteria such as [14]

$$\left|y_k^T\left(s_k - H_k y_k\right)\right| \geq \omega \left\|y_k\right\| \left\|s_k - H_k y_k\right\|,$$

with a small $\omega > 0$ can be applied to determine if the update needs to be skipped.

The main reason to choose the SR1 update over the popular BFGS update is its ability to produce stable Hessian approximations without an exact line-search [14], which the learned iterative solvers cannot impose.

B. Training schemes

Each of the learned iterative schemes (14)-(16) consists of K separate networks with respective weights θ_k. Given supervised training data pairs $\{x_i, h_i\}_{i=1}^{\mathcal{I}}$ satisfying (4), we aim to train the reconstruction operator $\mathcal{R}_\theta$ in (13) defined by the updates (12) and where $\theta = \{\theta_0, \ldots, \theta_{K-1}\}$. The common ideal approach to train the reconstruction $\mathcal{R}_\theta$ operator is end-to-end, that means we train all iterates simultaneously by minimizing the following empirical loss

$$\theta^* = \arg\min_\theta \sum_{i=1}^{\mathcal{I}} \left\|\mathcal{R}_\theta(h_i) - x_i\right\|_2^2.$$

In order to compute the backpropagation, all of the K networks are unfolded. Therefore, each training iteration requires computing K update directions and backpropagation through all intermediate steps. Evidently, training the networks in this end-to-end manner quickly becomes computationally infeasible if a large number of learned iterations is used or the dimensionality of the optical parameters is large. Additionally, the computation of the update directions needs to support either backpropagation or a have a defined derivative.

To avoid the need for backpropagation through the forward model and computations of the step direction, a greedy (iterate-wise) version of the optimization problem (III-B) can be considered. In the greedy training scheme each of the networks in (12) is trained separately and sequentially, forming a chain of K training problems of the form

$$\theta_k^* = \arg\min_\theta \sum_{i=1}^{\mathcal{I}} \left\|\Lambda_{\theta_k}\left(x_i^{(k)}, p_k\left(x_i^{(k)}, h_i\right)\right) - x_i\right\|_2^2, \tag{18}$$

$$\begin{cases} x_i^{(k)} & = \Lambda_{\theta_{k-1}^*}\left(x_i^{(k-1)}, p_{k-1}\left(x_i^{(k-1)}, h\right)\right), \text{ for } k > 0, \\ x_i^{(0)} & = [c_{\mu_a,i}, c_{\mu_s',i}]. \end{cases}$$

In this work, we assume that for each sample there exists rough estimates $c_{\mu_a,t} \in \mathbb{R}$ for the magnitude of absorption and $c_{\mu_s',t} \in \mathbb{R}$ reduced scattering for each sample and set uniform initial values $x_t^{(0)} = \left(\mathbf{1}c_{\mu_a,i}, \mathbf{1}c_{\mu_s',i}\right)$, where $\mathbf{1} \in \mathbb{R}^N$ is vector of ones.

In the optical problem, we estimate reduced scattering and absorption simultaneously. The updating network Λ_{θ_k} then takes, absorption μ_a^k, reduced scattering $\mu_s'^{(k)}$ and their respective step directions $p_k^{\mu_a}$ and $p_k^{\mu_s'}$. However, if the scattering, absorption, and step directions are not correlated, forcing the networks to learn the correlations can hinder the performance. In this work, we consider networks $\Lambda_{\theta_k}^{\mu_s'}$ and $\Lambda_{\theta_k}^{\mu_a}$ that separately update the absorption and reduced scattering based on their respective step directions.

In the end-to-end training (III-B), the step directions p_k are changing during the training and computed separately for each training iteration. Training the network End-to-End

is expensive, as all intermediate steps of computing step directions are stored and backpropagated through. Whereas, in the greedy training (18), training a single network Λ_{θ_k} requires computing the step directions only once for each sample, and the backpropagating is done merely through the network Λ_{θ_k}.

We emphasize that in this work, the update steps (14)-(16) are computed with respect to the DA, but could be similarly implemented with other light propagation models.

C. Network architectures

The used network architecture Λ_{θ_k} for the model-based learned iterative methods followed the structure of the ResNet as earlier utilized in [33]. A single updating network block can be seen in Fig. 2. The scattering and absorption were updated using separate networks, i.e, the two-channel input of one network was the current absorption $\mu_a^{(k)}$ and respective step direction, while the other network was given the current reduced scattering $\mu_s'^{(k)}$ and respective step direction. For both networks, the input was followed by three 32-channel convolutional layers with 3×3 kernel, bias, group norm of 8 groups, and ReLU activation. The fourth convolutional layer reduces the 32 channels to a single channel, which updates the current optical values. The last layer is not followed by ReLU, to allow negative optical parameter changes. To strictly restrict the positivity of the optical parameters, a ReLU activation function with a small additive bias of 10^{-5} was used at the end of each network Λ_{θ_k}.

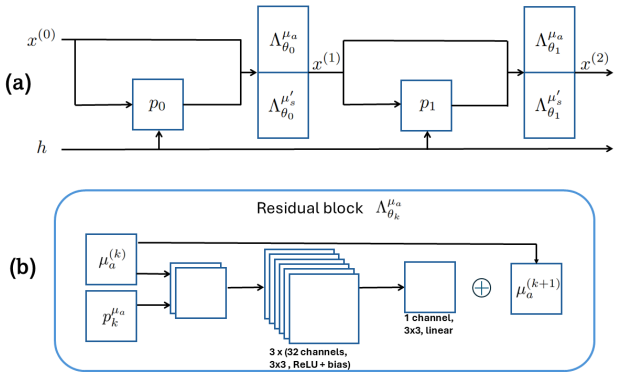

Fig. 2. Used network architecture for the learned model-based iterative solvers. **a)** unenrolled first two iterations of the end-to-end trained iterative network that takes the initial optical values and the data h as the input as in (12). **b)** architecture of a single updating network $\Lambda^{\mu_a}_{\theta_k}$. The network takes the current absorption values $\mu_a^{(k)}$ and the respective step direction $p_k^{\mu_a}$ as two-channel input and uses 3 convolutional layers of 32 channels with ReLU, and a final single-channel layer with a final linear layer to produce the updated absorption values $\mu_a^{(k+1)}$. The networks $\Lambda^{\mu'_s}_{\theta_k}$ in **a)** have the same architecture as shown in **b)**, but take the current reduced scattering $\mu_s'^{(k)}$ and respective step direction $p_k^{\mu'_s}$ as the inputs.

To compare the model-based approach to a fully learned inversion scheme, a residual U-Net based realization was implemented as in (11). The input for the U-Net was the data, absorbed energy density, from which it directly produced the absorbtion and scattering. Again, two separate networks were used to produce absorption and scattering independently. The used residual U-Net follows a standard architecture with three scales (32, 64, and 128 channels), i.e., two max-pooling layers, and two convolutional layers on each scale for the encoding and decoding path.

IV. Experimental setups: data generation and implementation

A. Ideal simulated data

In the first experiment, an ideal setup was constructed by simulating a comprehensive amount of samples, in a highly diffusive region where the overall modeling error from using DA was negligible. This setup is ideal for comparing the convergence and reconstruction accuracy of the learned iterative methods with different step directions and training schemes.

For this experiment, we consider a rectangular 2-D domain (20 mm $\times$ 25 mm) for the optical parameters. The training samples were chosen to consist of randomly located and sized ellipsoidal inclusions, which were also allowed to overlap. The background optical values (mm^{-1}) of the samples were drawn to be uniformly between 0.0085 and 0.0115 for absorption and between 1.7 and 2.6 for reduced scattering. The contrast of absorption and scattering inclusions was uniformly drawn to be between 0.2 and 3.5 times the background optical values of each sample. The nodes were additionally corrupted by Gaussian noise with a standard deviation of 2% of the amplitude. Finally, the samples were slightly blurred with a Gaussian filter to make the inclusion edges smoother, which would likely be the case with practical targets. These generated samples with ellipses can be interpreted to mimic, for instance, a cross-section of veins.

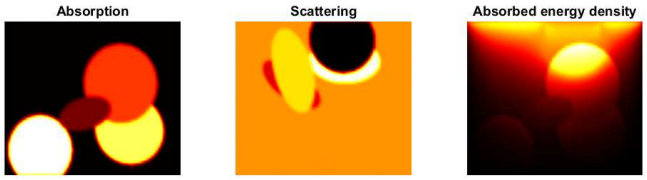

Fig. 3. *Ideal problem*: Example of drawn absorption (left) and scattering (mid) coefficients and respective simulated absorbed energy density (right), when the top side was illuminated.

To simulate the absorbed energy densities of each sample, the ValoMC [13] open Monte Carlo software package for Matlab was used. ValoMC utilizes the photon package method to simulate the fluence in the given geometry and light sources. The absorbed energy densities are then obtained by multiplying the fluence by the absorption as in (2). To guarantee a unique solution when estimating scattering and absorption simultaneously, two separate limited-angle illuminations were performed, forming two sets of absorbed energy densities for each sample. The illuminations were simulated from the top and right side of the rectangle using 10^8 photons uniformly entering from the sides. By default, the ValoMC scales the total strength of the light source to correspond to 1 W. Example of drawn absorption and scattering coefficients and respective simulated absorbed energy density (top side illuminated) are shown in Fig. 3. The simulated absorbed energy density was corrupted by noise drawn from a Gaussian distribution with a standard deviation corresponding to 1% of the absorbed

energy density values. The simulations were repeated with 1250 samples.

B. Digital twin phantoms

We refer to the second numerical example as the digital twin problem. It is much closer to experimental conditions compared to the ideal problem. The problem now has limited training data, samples with varying locations of the tubes (region of interest), a large range of possible optical values, and a large modeling error (see Fig. 4).

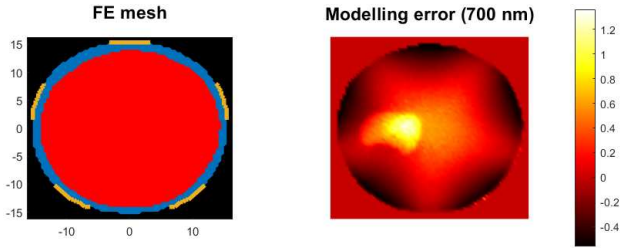

Fig. 4. *Digital twin problem*: Used finite element mesh for the digital twin problem (left). The blue regions represent the water nodes, red the estimated nodes, and yellow the light source (700 nm) locations. The right figure shows nodewise relative difference between predicted absorbed energy density from diffusion approximation and simulated measurements for a single *twin* sample.

We emphasize that since the data is statistically simulated based on physical absorption and scattering phantoms, this corresponds to a case where the acoustical problem is perfectly solved, and possible noise from the optical inversion is not apparent.

The digital twin problem re-used data from physical tissue-mimicking phantoms with piecewise-constant material distributions [26], [44], enabling optical characterization via an in-house double integrating sphere system [45], [46] to determine wavelength-dependent absorption (μ_a) and reduced scattering (μ_s') coefficients between 600 and $950\,\mathrm{nm}$. The phantoms were immersed in a water bath and imaged using a pre-clinical PAI system (MSOT inVision-256TF, iThera Medical GmbH, Munich, Germany). It has 256 transducer elements in a circular array with an angular coverage of 270°, which have a 5-MHz centre frequency and 60% bandwidth. We imaged in the wavelength range from 700 nm to 900 nm in 10 nm steps using 10-frame averaging.

We constructed digital twins of the physical phantoms using manual segmentation masks derived with MITK [47] based on images obtained with filtered backprojection. We assigned our reference optical measurements and reference acoustic characterisation from the literature (density: $1000\,\mathrm{g/cm^3}$; sound speed: $1468\,\mathrm{m/s}$ for phantoms, $1489\,\mathrm{m/s}$ for coupling medium) to each piecewise-constant material region.

Our simulation pipeline computes device-specific initial pressure distributions $p_0(r)$ and measurement data $p(t, r)$ as follows:

1) Light fluence ϕ was simulated using MCX [48], yielding $p_0(r) = \Gamma \cdot \mu_a(r) \cdot \phi(r)$ with constant Grüneisen parameter Γ.
2) Acoustic wave propagation was modeled using k-Wave's k-space pseudospectral method [49] to generate $p(t, r)$.

A digital twin of the MSOT InVision-256TF implemented in SIMPA [50] incorporated vendor-provided hardware geometry. Custom MCX illumination and k-Wave transducer arrays were validated against 15 calibration phantoms. Optimized Gaussian illumination profiles achieved experimental agreement in radiant exposure measurements. SIMPA orchestrated all simulations through integrated MCX and k-Wave adapters. More detailed information on the digital twin construction can be found in [37], details on the phantoms in [45], and details on the imaging system in [26].

The total number of actual digital twin samples for training was only 14. As such, we supplemented the digital twins with additional phantom simulations, where the optical properties are not in correspondence with physical phantoms but instead determined the following way: (1) Draw two random material characterizations from the physical phantom data, M_1, M_2; (2) Determine a random mixing factor $c = [0, 1[$; (3) Calculate the new material property $M_3 = c \cdot M_1 + (1 - c) \cdot M_2$. The resulting distribution can be found in the supplementary file Fig. S1. Each phantom was drawn to have between 1 and 3 inclusions and a radius of 14.2 mm. We simulated at six wavelengths: 700, 740, 780, 820, 860, and 900 nm. In total, we supplemented the training data with 41 of these phantom simulations.

The digital twin problem introduces several sources of modeling error. Firstly, the simulations were conducted in 3D, but we approximated the light distribution in 2D, where the photons need to travel a much shorter path to reach the center. Secondly, the optical values of the samples contained relatively high absorption coefficients, violating the $\mu_a << \mu_s'$ assumption of the DA, hence causing it to be inaccurate.

C. Implementing the forward models

The FE solution of the diffusion approximation was implemented according to the equations in Sec. II-A using PyTorch tensors and built-in functions. A piecewise constant basis was used for the optical parameters. In the case of the ideal simulated data, the whole 2-D square domain was discretized with 4636 nodes, forming 9000 evenly sized triangular elements. For digital twin discretization, most of the water around the tubes was left out of the FE discretization. However, as the same FE mesh was used for all of the samples and the location of the tubes slightly varied, a small layer of water remains in the discretization region, as demonstrated in Fig. 4. For the convolutional networks, the discretization was embedded back to an evenly spaced grid of 72×72 nodes. The FE discretized part of the domain had evenly spaced 4172 nodes and 8148 evenly sized triangular elements. The location of the tubes in the water was assumed to be known, and only the non-water nodes were estimated.

For the ideal simulated data, DA was implemented with light sources that uniformly illuminated the top and right sides, similar to the ValoMC simulation. Whereas, for the digital twin experiments, the five 6.12 mm wide light sources were approximated to have a Gaussian intensity profile with a standard deviation of 3.

D. Backpropagation and computational considerations

For the End-to-End training (III-B), computation of the update steps (14)-(16) is implemented inside the networks. This includes computing the FEM matrices (3), the corresponding fluence (3), and solving the chosen step direction (14)-(16). As these computations come with a large number of operations for which gradients are needed with respect to the weights of the networks, it is convenient to utilise the automatized gradient computation provided by PyTorch. PyTorch currently supports GPU versions for all the required matrix operations to compute the fluence and step directions. The detailed steps to compute the Jacobian of (5) with respect to scattering and absorption can be found, for instance, in [11].

For each of the step directions, the weighting matrix L_e in (5) needed to be determined. The noise profile for the ideal simulations, was known to be Gaussian $e \sim \mathcal{N}(0, \Gamma_e)$ (with 1% standard deviation of the data amplitude). Therefore, the weight matrix L_e was chosen to be the Cholesky factor of the inverse of the covariance $\Gamma_e^{-1} = L_e^T L_e$. The noise profile of the digital twin was unknown, and the weighting matrix L_e was set as the identity matrix.

In the digital twin simulations, the samples had an extensive range of data values, and weighting each node equivalently led to poor convergence of certain regions. To overcome this, a log-scaled data space was used, substantially improving the performance of both classical and the learned solvers when used for the digital twin problem. Using log-scaled space for the ideal problem that had a narrow range of data values did not show a noticeable difference in performance.

Solving the digital twin problem uniquely required assuming the wavelength dependency, for the scattering. The training samples were fit to approximately follow $\mu_s'(r, \lambda) \approx a(r)e^{-b\lambda}$, dependency with sample-specific constant $b > 0$. However, the constants b were varying only slightly, and using the average value 1.6×10^{-3} was observed to work sufficiently for the training and test samples.

E. Training procedures and computational requirements

The training-validation-test splits were done as follows. For ideal simulations, a total of 1250 samples were drawn, with a training-test-validation split of 80:10:10. For digital twin experiments, the samples were roughly labeled as follows: *i)* The generated complementary *circular* samples that had absorption and scattering drawn as described in section IV-B, *ii)* digital *twin* targets that represent the experimental phantoms similar to *circular* samples, *iii) outlier* samples that represent the experimental phantoms with much larger inclusions. The training set consisted of 8 *twin* samples and 41 *circular* samples. Since the illumination patterns were approximately symmetric with respect to the vertical axis (Fig. 4), the training set was augmented by flipping the vertical coordinates of the samples, doubling the size of the training set. The validation set consisted of 4 *circular* targets, and the test set 6 digital *twin* samples, 6 complementary *circular* samples, and 5 *outlier* samples. Each sample was solved with six different wavelengths, 700, 740, 780, 820, 860, 900 (nm), yielding a total of 6×98 training samples and 6×17 test samples. However, the optical values and locations of the inclusion were highly correlated and only marginally different for the six wavelengths used.

For all training tasks, the ℓ^2 loss function was used. Due to reduced scattering being significantly larger than absorption, in the loss function, the scattering of digital twin samples was weighted by a factor of 1/10, and by a factor of 1/100 for ideal samples, balancing the loss contribution. The model-based iterative learned solvers were trained using 35000 (ideal) and 40000 (digital twin) training steps. The learning rate followed cosine annealing scheduling with initial learning rates of 10^{-4} (ideal) and 4×10^{-5} (digital twin). Due to the volatile behaviour of the optical values inside untrained end-to-end networks Λ_{θ_k}, a few thousand smaller initial learning steps were used to avoid numerical errors and unstable inversion.

The memory consumption of the greedy training was primarily contributed by the stored Jacobian and Hessian, yielding less than 2 GB of peak GPU memory usage for all greedily trained solvers. For end-to-end training, all of the intermediate steps of computing the fluence and step direction needed to be stored for efficient backpropagation. This produced a linear increase in the memory consumption with respect to the number of used updating networks Λ_{θ_k}. The memory consumption increases of a single updating network Λ_{θ_k} within the ideal problem are shown in Table I.

TABLE I
MEMORY CONSUMPTION AND RUN TIMES

Solver	t_{forw} (s)		t_{back} (s)		Memory (GB)	
	Ideal	Twin	Ideal	Twin	Ideal	Twin
GD	0.32	0.34	0.51	0.23	2.7	1.8
GN	0.62	0.56	0.82	0.36	3.4	3.1
SR1	0.33	X	0.52	X	3.4	X
U-Net	0.01	<0.01	0.03	0.01	0.1	0.1

In the greedy training, the majority of the training time was spent on solving the fluences and step directions for each sample. The time to compute forward pass and backpropagation through only an updating network Λ_{θ_k} took less than 0.01 s. Therefore, the total training time for the greedy training tasks was approximately $S \times t_{forw} \times K$, where S is the number of training samples, K is the number of updating networks Λ_{θ_k}, and t_{forw} is the time of computing a single step direction. The times t_{forw} to compute the different step directions are shown in Table I. For instance, five updating networks of GN step greedily trained with 35000 iterations using 1000 samples, took approximately one hour.

In the end-to-end scheme, evaluating the forward pass took roughly the same amount t_{forw} as in the greedy scheme, but now the backpropagation was done through all of the intermediate steps of computing the step directions. Also, since the step directions (except the initial step directions) were changing during the training, computing the K step directions and back propagation through them was repeated for each training iteration T. Therefore, the total training time for the end-to-end tasks was approximately $T \times (t_{forw} + t_{back}) \times (K - 1)$, where t_{back} is the time taken to backpropagate through a single updating network and the intermediate steps of a single step direction computation. The average forward pass and

backpropagation evaluation times for the end-to-end solvers are shown in Table I. For instance, within an ideal problem, five end-to-end trained updating networks with GN step and 35000 training iterations took roughly 62 hours to train.

The residual U-net used for the fully learned approach took approximately 30 minutes to train with 80000 training iterations. The reconstruction (forward pass) was evaluated in less than 0.01 s. All of the computations were performed using an RTX A5000 GPU equipped with 24GB of memory. All of the used codes can be found on GitHub (https://github.com/AnsManni/Learned_iterative_QPAT)

V. Results

A. Ideal problem

In the first study, we tested greedy and end-to-end trained GD, GN, and SR1 update-based learned iterative methods on the ideal simulated problem. To compare the performance of using varying numbers of updating networks Λ_{θ_k}, multiple learned iterative solvers were trained, using between 1 and 9 updating networks. As a comparison, a residual U-Net-based fully learned reconstruction method was tested. Further, a single test sample was reconstructed using the classical GN updates with ℓ^2 regularizer utilizing local spatial correlations via the Ornstein-Uhlenbeck process 10 with correlation length of 1 mm and standard deviation of 0.005 for absorption and 0.035 for scattering. The learned GN used ℓ^2-regularizer with regularization parameters of 10000 for absorption and 1.8 for scattering. All initial values for the classical and learned solvers were set to correspond to the average absorption (0.01) and scattering (2) values over the training set, which were also used as the mean of the ℓ^2-regularizer (9).

For the classical GN updates, we found a bisection style inexact line-search [51] to be efficient. For the learned GN updates, we used an uncorrelated ℓ^2-regularizer (9) with regularization parameters of 12.5 for absorption and 0.17 for scattering.

In general, the performance of the classical solver was heavily affected by the chosen regularizer and regularization parameters. In our experiments, we found that including local spatial correlations in the form of the Ornstein-Uhlenbeck process produced stable reconstructions across the simulated samples. On the other hand, the performance of the learned GN and SR1 was consistent over a wide set of regularization parameters and only started to decrease once the regularization parameters were drastically increased.

Fig. 5 shows the average relative error of absorption and reduced scattering over the test set of 125 samples for GD, SR1, and GN based learned iterative methods up to 9 updating networks Λ_{θ_k}. After 9 iterations, the difference in average relative errors was observed to be marginal. The relative error of the fully learned (single-step) reconstructions is shown as the vertical dashed line in Fig. 5. Reconstructions of a single test sample are shown in Fig. 6 for the classical GN solver and the residual U-Net and in Fig. 7 for each of the greedily and end-to-end trained methods.

Relative errors in Fig. 5 reveal the learned GN solver having significantly better performance compared to the GD and SR1. The learned GN based solver is also the only one to noticeably surpass the accuracy of the residual U-Net. For the learned GN updates, there was no distinguishable difference between the end-to-end and greedily trained networks. On the other hand, the learned GD and SR1 yielded relatively accurate absorption and scattering reconstructions, but only when trained end-to-end. The generally poor performance of the greedily learned iterative solvers can be seen in Fig. 7. Only the GN updates provided sufficient information to the networks making the greedy training produce feasible absorption and scattering reconstructions (Fig. 7).

The residual U-Net was also able to produce accurate

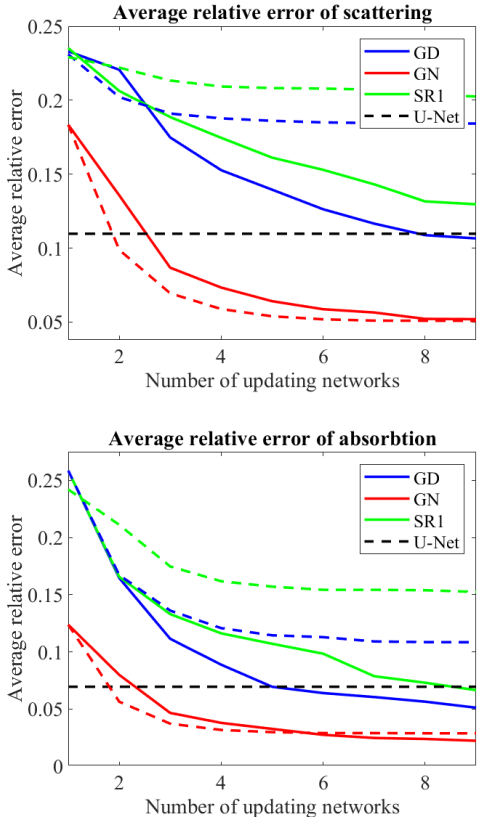

Fig. 5. *Ideal problem*: Average relative errors of absorption (μ_a) and reduced scattering (μ_s') over 125 test samples. The solid lines show the relative errors of end-to-end trained learned iterative methods up to 9 updating networks Λ_{θ_k} and the dashed lines the corresponding greedily trained values. The black dashed line shows the average relative error from the residual U-Net.

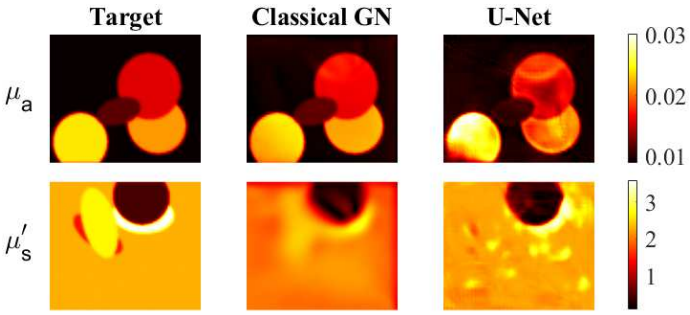

Fig. 6. *Ideal problem*: absorption (μ_a) and reduced scattering (μ_s') reconstructions of a test sample (left) using classical GN solver (14 iterations) and residual U-Net.

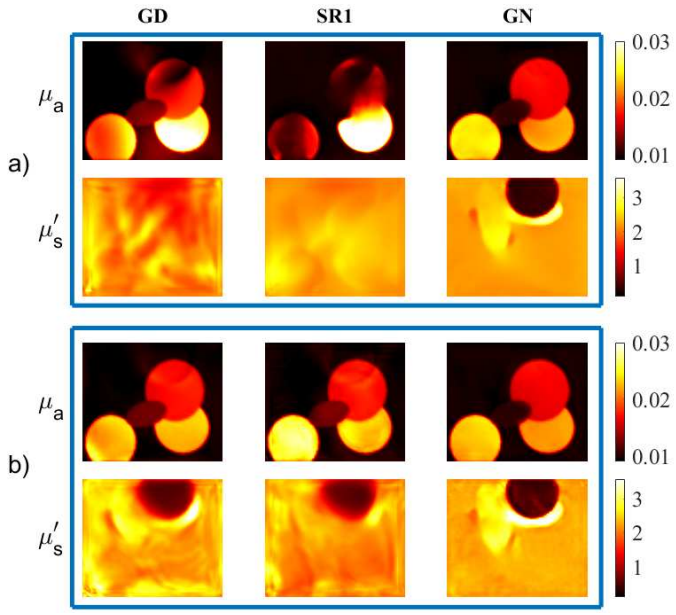

Fig. 7. *Ideal problem*: Absorption (μ_a) and reduced scattering (μ_s') reconstructions of a single test sample using **a)** greedily and **b)** end-to-end trained learned iterative SR1, GD, and GN with 9 updating networks Λ_{θ_k}.

reconstructions with an average relative error of 7% for absorption and 9% for reduced scattering. As can be seen from Fig. 6, despite low relative scattering error, the U-Net tends to produce many hallucinated artifacts in the scattering reconstructions. The learned iterative methods did not in general, produce hallucinated scattering artifacts (Fig. 7).

The reconstructions with the classical GN in Fig. 6 demonstrate that with relatively low modeling error, even the classical solver, fine-tuned with proper regularization parameters, can accurately reconstruct both optical parameters. However, compared to the learned solvers, the classical solver required more GN iterations, each requiring multiple forward model evaluations to determine a suitable step length.

B. Digital twin problem

In the digital twin study, we reconstructed a set of digital phantoms with approximately the same shapes and optical parameters as existing experimental phantoms. We used a finite element approximation of the DA light propagation model based on the real measurement setup. The reconstruction was done using end-to-end GD and GN based learned iterative methods, a fully learned residual U-Net, and a classical GN solver with the total variation regularizer, ideal for reconstructing the piecewise constant inclusions. Up to five end-to-end trained GD/GN updating networks were used, as no notable improvement was achieved with further networks. The greedily trained layers were not observed to improve reconstruction accuracy after the initial iteration. The learned SR1 was excluded from this study due to its generally poor performance that was observed outside of the reported results.

The initial values $x^{(0)}$ were drawn to be uniformly 0-30% off from the ground truth background optical values of each sample. These initial values were also used as the mean x_μ of the ℓ^2 regularizer (9) for the learned GN. The classical total variation regularized problem was solved by using the primal-dual interior point (PD-IPM) method [52]. The total variation regularizer was used with a smoothness parameter of 10^{-4} and a regularization parameter of 0.1.

Fig. 8 shows the relative errors of each reconstructed test sample type using residual U-Net, learned iterative GD, and GN, both with 5 updating networks. The relative errors show that all of the learned methods performed well with the *circular* test samples. With absorption, the fully learned U-Net performed slightly better than the iterative learned solver, but with scattering, the iterative solvers achieved better accuracy. With the *twin* samples, the overall reconstruction quality was still feasible, especially for the learned GN, which reconstructed both scattering and absorption quite accurately. The most challenging *outlier* samples were, in general, hard to reconstruct for all of the learned methods.

For U-Net, the amount of training samples was evidently too low, as the accuracy difference between the training and test reconstructions was significant. The learned iterative solvers generalized well with the *circular* test samples, but less well with the *twin* and *outlier* samples.

Fig. 9 shows example reconstructions with classical solver and the learned methods, for all sample types. For the classical solver, the modeling error was overwhelming, causing a very poor reconstruction quality for both scattering and absorption as seen from Fig. 9. We tested the classical solver with different regularization and smoothing parameters, but none of the changes generally improved the reconstruction quality.

VI. Discussion

A. Ideal problem

In the first study, we investigated the performance of different learned iterative model-based solvers when used to solve a nonlinear inverse problem under ideal conditions. The results showed significant differences in reconstruction accuracy when the GD, GN, and SR1 step directions were used as the information for the networks. Increasing the number of trained networks increased performance, analogous to the behavior of classical iterative solvers. However, all the learned iterative solvers except the greedily trained SR1, were drastically more robust than the classical GN solver. The performance of the learned solvers was not sensitive to the chosen prior values, whereas finding a plausible prior value for the classical solver was not trivial.

In general, the learned GN achieved superior reconstruction accuracy compared to the other methods. However, the whole Hessian was computed for GN updates, leading to larger memory consumption and longer training times. While the aim of utilizing the learned SR1 update was to match the convergence speed and accuracy of the GN updates, our study showed that the SR1 step direction seems to provide information similar to the GD step. However, the expensive matrix inversion needed for the GN step could also be approximated by using existing matrix-free iterative techniques.

For a much larger dimensional problem, the training procedure (scheduling, initial weights, etc.) should be optimized as

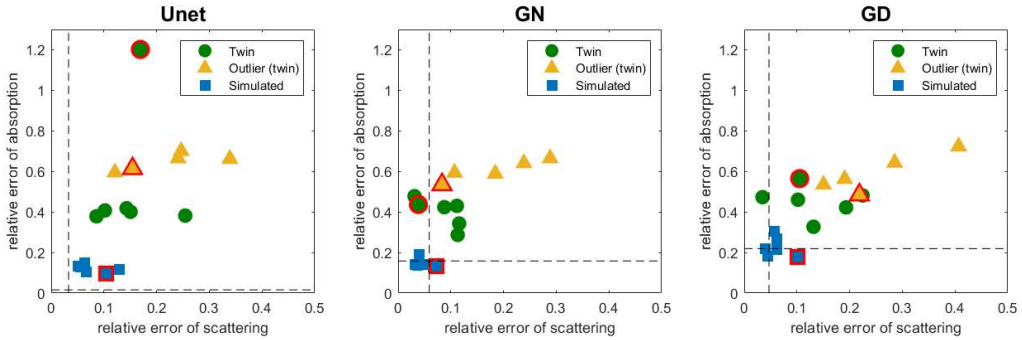

Fig. 8. *Digital twin problem*: relative errors of absorption (μ_a) and reduced scattering (μ_s') reconstruction from U-Net (left), learned iterative Gauss-Newton (mid), and Gradient descent (right) for *circular*, *twin*, and *outlier* samples. The dashed black line shows the average relative error of training samples. The relative error of each plotted sample is averaged over the six solved wavelengths. The learned gradient descent method was using two updating networks Λ_{θ_k} and the learned Gauss-Newton five updating networks. The highlighted samples correspond to the shown samples in Fig. 9.

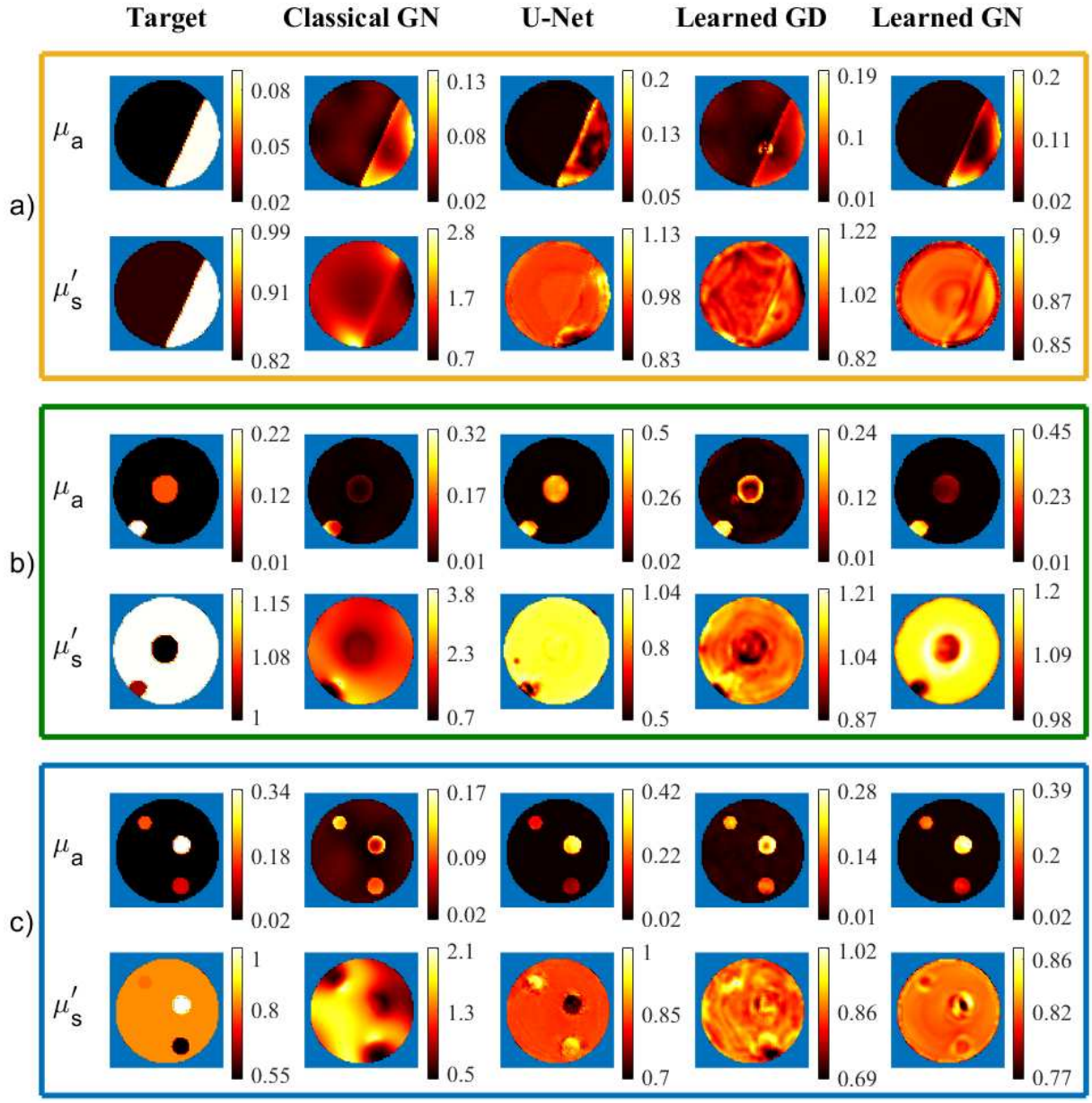

Fig. 9. *Digital twin problem*: absorption (μ_a) and reduced scattering (μ_s') reconstructions of **a)** *outlier*, **b)** *twin*, and **c)** *circular* test samples (leftmost column). The methods used for the reconstructions starting from the second leftmost column are classical Gauss-Newton with total variation regularizer after around 30 iterations, fully learned U-Net, learned gradient descent with $K = 4$, and learned Gauss-Newton with $K = 5$.

much as possible to reduce training time. Also, the resolution of the first few networks could be coarser and refined in later iterations to reveal finer targets. It could also be beneficial to set the initial optical values more accurately, based on, for instance, a few greedy iterations or the output of the U-Net.

B. Digital twin

In the second study, we investigated the performance of the implemented solvers in a more challenging setup, incorporating only a small training set and a large modeling error. In contrast to the ideal problem, the classical GN solver was now generally unable to reconstruct the targets due to the modeling error. The single-step U-Net-based solver performed relatively well. As the only input for the U-Net was the almost noiseless simulated data, there was no apparent modeling error or noise, making the data ideal input for the U-Net. Therefore, for the U-Net, the principal challenge was the scarce training set with samples different from the test set.

One question was whether the augmented *circular* training samples were good enough for the U-Net and iterative learned networks to generalize to the *twin* and *outlier* samples. The relative errors in Fig. 8 demonstrate that in fact, the iterative networks did generalize quite well for the *twin* samples, but not for the *outlier*. The U-Net achieved the highest reconstruction accuracy on training samples, but failed to generalize well to the test samples.

While the learned solvers steadily improved the reconstruction accuracy with respect to the number of updating networks in the ideal study, they were now not able to achieve similar convergence. The main obstacle to the performance of the iterative learned solvers was the modeling error, which was not directly compensated. The end-to-end networks indirectly compensate for the modeling error as the information of the gradients flows through the networks, whereas the greedy training computes erroneous step directions and is unable to improve after the first network. The step directions inside the end-to-end network are computed according to the term $\|h - F(x)\|_2^2$, corrupted by the modeling error. To make the gradients more informative, a more accurate light transport model could be considered.

Using more accurate models, such as the RTE, comes with a heavy computational cost. However, combined with greedy training, it can still be faster than using less accurate models with end-to-end training. The question would be whether a greedily trained solver with a more accurate model performs better than an end-to-end trained solver with an inaccurate model.

Instead of using more expensive models, a modeling error compensation technique could be considered, too. However, as traditional error compensation methods, such as Bayesian approximation error modeling, utilize a large number of samples to estimate error statistics, it is not clear if these methods can provide a remedy if only a low number of training samples is available. As the behavior of the modeling error depends on the optical values and their relative spatial distances to each other, using an additional convolution-based correction layer could allow for a rough estimation of the modeling error.

VII. Conclusions

In this work, we described, implemented and examined learned model-based iterative solvers for the nonlinear inverse problem of QPAT using different GD, GN, and SR1-based step directions. Two different training schemes were investigated: (1) the end-to-end scheme, leading to an optimal solution after K trained iterations; and (2) the computationally cheaper greedy scheme, where the iterations were trained separately. The first numerical study revealed the differences between the step directions and training schemes under ideal conditions: In general, the more informative GN step direction performed well with both greedy and end-to-end schemes, whereas the GD and SR1 performed robustly only within the end-to-end training scheme.

In the second study, the digital twin problem was introduced, providing a closer to real-life setup, with a scarce amount of training data and modeling inaccuracies. The results showed that the end-to-end trained GN and GD networks were partly able to compensate for the modeling error. However, the learned iterative solvers could only reach an accuracy similar to the fully learned method. To justify the further usage of the learned iterative solvers, future work should investigate model error compensation techniques or employ more accurate light transport models.

While we studied the problem of QPAT in this paper, we expect that the presented insights on step directions, training regimes, and model errors are relevant for nonlinear inverse problems in general.

References

[1] P. Beard, "Biomedical photoacoustic imaging," *Interface Focus*, vol. 1, 2011.

[2] L. V. Wang, *Photoacoustic imaging and spectroscopy*. CRC press, 2017.

[3] C. Li and L. V. Wang, "Photoacoustic tomography and sensing in biomedicine," *Physics in Medicine & Biology*, vol. 54, no. 19, p. R59, 2009.

[4] B. Cox, J. G. Laufer, S. R. Arridge, and P. C. Beard, "Quantitative spectroscopic photoacoustic imaging: a review," *Journal of biomedical optics*, vol. 17, no. 6, pp. 061 202–061 202, 2012.

[5] L. Li and L. V. Wang, "Recent advances in photoacoustic tomography," *BME frontiers*, 2021.

[6] J. Xia and L. V. Wang, "Small-animal whole-body photoacoustic tomography: a review," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 5, pp. 1380–1389, 2013.

[7] L. V. Wang and J. Yao, "A practical guide to photoacoustic tomography in the life sciences," *Nature methods*, vol. 13, no. 8, pp. 627–638, 2016.

[8] A. Hauptmann and T. Tarvainen, "Model-based reconstructions for quantitative imaging in photoacoustic tomography," in *Biomedical Photoacoustics: Technology and Applications*. Springer, 2024, pp. 133–153.

[9] S. R. Arridge, "Optical tomography in medical imaging," *Inverse problems*, vol. 15, no. 2, p. R41, 1999.

[10] B. Cox, T. Tarvainen, and S. Arridge, "Multiple illumination quantitative photoacoustic tomography using transport and diffusion models," in *Tomography and Inverse Transport Theory*, 2011, vol. 559, pp. 1–12.

[11] T. Tarvainen, A. Pulkkinen, B. T. Cox, J. P. Kaipio, and S. R. Arridge, "Image reconstruction in quantitative photoacoustic tomography using the radiative transfer equation and the diffusion approximation," in *European Conference on Biomedical Optics*. Optica Publishing Group, 2013, p. 880006.

[12] M. Firbank, S. R. Arridge, M. Schweiger, and D. T. Delpy, "An investigation of light transport through scattering bodies with non-scattering regions," *Physics in Medicine & Biology*, vol. 41, no. 4, p. 767, 1996.

[13] A. A. Leino, A. Pulkkinen, and T. Tarvainen, "Valomc: a monte carlo software and matlab toolbox for simulating light transport in biological tissue," *Osa Continuum*, vol. 2, no. 3, pp. 957–972, 2019.

[14] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.

[15] T. Saratoon, T. Tarvainen, B. Cox, and S. Arridge, "A gradient-based method for quantitative photoacoustic tomography using the radiative transfer equation," *Inverse Problems*, vol. 29, no. 7, p. 075006, 2013.

[16] B. T. Cox, S. Arridge, K. Kostli, and P. C. Beard, "Quantitative photoacoustic imaging: fitting a model of light transport to the initial pressure distribution," in *Photons Plus Ultrasound: Imaging and Sensing 2005*, vol. 5697. SPIE, 2005, pp. 49–55.

[17] B. T. Cox, S. R. Arridge, and P. C. Beard, "Estimating chromophore distributions from multiwavelength photoacoustic images," *JOSA A*, vol. 26, no. 2, pp. 443–455, 2009.

[18] T. Tarvainen and B. Cox, "Quantitative photoacoustic tomography: modeling and inverse problems," *Journal of Biomedical Optics*, vol. 29, no. S1, pp. S11 509–S11 509, 2024.

[19] A. Hauptmann and B. Cox, "Deep learning in photoacoustic tomography: current approaches and future directions," *Journal of Biomedical Optics*, vol. 25, no. 11, pp. 112 903–112 903, 2020.

[20] J. Gröhl, M. Schellenberg, K. Dreher, and L. Maier-Hein, "Deep learning for biomedical photoacoustic imaging: A review," *Photoacoustics*, vol. 22, p. 100241, 2021.

[21] T. Chen, T. Lu, S. Song, S. Miao, F. Gao, and J. Li, "A deep learning method based on u-net for quantitative photoacoustic imaging," in *Photons Plus Ultrasound: Imaging and Sensing 2020*, vol. 11240. SPIE, 2020, pp. 216–223.

[22] C. Cai, K. Deng, C. Ma, and J. Luo, "End-to-end deep neural network for optical inversion in quantitative photoacoustic imaging," *Optics letters*, vol. 43, no. 12, pp. 2752–2755, 2018.

[23] C. Yang, H. Lan, H. Zhong, and F. Gao, "Quantitative photoacoustic blood oxygenation imaging using deep residual and recurrent neural network," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE, 2019, pp. 741–744.

[24] C. Bench, A. Hauptmann, and B. Cox, "Toward accurate quantitative photoacoustic imaging: learning vascular blood oxygen saturation in three dimensions," *Journal of Biomedical Optics*, vol. 25, no. 8, pp. 085 003–085 003, 2020.

[25] J. Gröhl, K. Yeung, K. Gu, T. R. Else, M. Golinska, E. V. Bunce, L. Hacker, and S. E. Bohndiek, "Distribution-informed and wavelength-flexible data-driven photoacoustic oximetry," *Journal of Biomedical Optics*, vol. 29, no. S3, pp. S33 303–S33 303, 2024.

[26] J. Gröhl, T. R. Else, L. Hacker, E. V. Bunce, P. W. Sweeney, and S. E. Bohndiek, "Moving beyond simulation: data-driven quantitative photoacoustic imaging using tissue-mimicking phantoms," *IEEE Trans. Med. Imag.*, vol. 43, no. 3, pp. 1214–1224, 2023.

[27] Z. Liang, S. Zhang, Z. Liang, Z. Mo, X. Zhang, Y. Zhong, W. Chen, and L. Qi, "Deep learning acceleration of iterative model-based light fluence correction for photoacoustic tomography," *Photoacoustics*, vol. 37, p. 100601, 2024.

[28] Z. Liang, Z. Mo, S. Zhang, L. Chen, D. Wang, C. Hu, and L. Qi, "Self-supervised light fluence correction network for photoacoustic tomography based on diffusion equation," *Photoacoustics*, vol. 42, p. 100684, 2025.

[29] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, "Solving inverse problems using data-driven models," *Acta Numerica*, vol. 28, pp. 1–174, 2019.

[30] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1322–1332, 2018.

[31] A. Hauptmann, F. Lucka, M. Betcke, N. Huynh, J. Adler, B. Cox, P. Beard, S. Ourselin, and S. Arridge, "Model-based learning for accelerated, limited-view 3-d photoacoustic tomography," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1382–1393, 2018.

[32] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a variational network for reconstruction of accelerated mri data," *Magnetic resonance in medicine*, vol. 79, no. 6, pp. 3055–3071, 2018.

[33] M. Mozumder, A. Hauptmann, I. Nissilä, S. R. Arridge, and T. Tarvainen, "A model-based iterative learning approach for diffuse optical tomography," *IEEE Trans. Med. Imag.*, vol. 41, pp. 1289–1299, 2021.

[34] W. Herzberg, D. B. Rowe, A. Hauptmann, and S. J. Hamilton, "Graph convolutional networks for model-based learning in nonlinear inverse problems," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1341–1353, 2021.

[35] F. Colibazzi, D. Lazzaro, S. Morigi, and A. Samoré, "Deep-plug-and-play proximal gauss-newton method with applications to nonlinear, ill-posed inverse problems," *Inverse Problems and Imaging*, vol. 17, no. 6, pp. 1226–1248, 2023.

[36] G. S. Alberti, D. Lazzaro, S. Morigi, L. Ratti, and M. Santacesaria, "Deep unfolding network for nonlinear multi-frequency electrical impedance tomography," *arXiv preprint arXiv:2507.16678*, 2025.

[37] J. Gröhl, L. Kunyansky, J. Poimala, T. R. Else, F. Di Cecio, S. E. Bohndiek, B. T. Cox, and A. Hauptmann, "Digital twins enable full-reference quality assessment of photoacoustic image reconstructions," *The Journal of the Acoustical Society of America*, vol. 158, no. 1, pp. 590–601, 07 2025.

[38] A. Ishimaru *et al.*, *Wave propagation and scattering in random media*. Academic press New York, 1978, vol. 2.

[39] L. G. Henyey and J. L. Greenstein, "Diffuse radiation in the galaxy," *Astrophysical Journal, vol. 93, p. 70-83 (1941).*, vol. 93, pp. 70–83, 1941.

[40] E. Malone, S. Powell, B. T. Cox, and S. Arridge, "Reconstruction-classification method for quantitative photoacoustic tomography," *Journal of Biomedical Optics*, vol. 20, no. 12, pp. 126 004–126 004, 2015.

[41] T. Tarvainen, B. T. Cox, J. Kaipio, and S. R. Arridge, "Reconstructing absorption and scattering distributions in quantitative photoacoustic tomography," *Inverse Problems*, vol. 28, no. 8, p. 084009, 2012.

[42] S. L. Jacques, "Optical properties of biological tissues: a review," *Physics in Medicine & Biology*, vol. 58, no. 11, p. R37, 2013.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[44] J. Gröhl, T. Else, L. Hacker, E. Bunce, P. Sweeney, and S. Bohndiek, "Dataset for: Moving beyond simulation: data-driven quantitative photoacoustic imaging using tissue-mimicking phantoms," *Apollo - University of Cambridge Repository*, 2023. [Online]. Available: https://www.repository.cam.ac.uk/handle/1810/359968

[45] L. Hacker, J. Joseph, A. M. Ivory, M. O. Saed, B. Zeqiri, S. Rajagopal, and S. E. Bohndiek, "A copolymer-in-oil tissue-mimicking material with tuneable acoustic and optical characteristics for photoacoustic imaging phantoms," *IEEE Trans. Med. Imag.*, vol. 40, no. 12, pp. 3593–3603, 2021.

[46] J. W. Pickering, S. A. Prahl, N. Van Wieringen, J. F. Beek, H. J. Sterenborg, and M. J. Van Gemert, "Double-integrating-sphere system for measuring the optical properties of tissue," *Applied optics*, vol. 32, no. 4, pp. 399–410, 1993.

[47] M. Nolden, S. Zelzer, A. Seitel, D. Wald, M. Müller, A. M. Franz, D. Maleike, M. Fangerau, M. Baumhauer, L. Maier-Hein *et al.*, "The medical imaging interaction toolkit: challenges and advances: 10 years of open-source development," *International journal of computer assisted radiology and surgery*, vol. 8, pp. 607–620, 2013.

[48] Q. Fang and D. A. Boas, "Monte carlo simulation of photon migration in 3d turbid media accelerated by graphics processing units," *Optics express*, vol. 17, no. 22, pp. 20 178–20 190, 2009.

[49] B. E. Treeby and B. T. Cox, "k-wave: Matlab toolbox for the simulation and reconstruction of photoacoustic wave fields," *Journal of biomedical optics*, vol. 15, no. 2, pp. 021 314–021 314, 2010.

[50] J. Gröhl, K. K. Dreher, M. Schellenberg, T. Rix, N. Holzwarth, P. Vieten, L. Ayala, S. E. Bohndiek, A. Seitel, and L. Maier-Hein, "Simpa: an open-source toolkit for simulation and image processing for photonics and acoustics," *Journal of biomedical optics*, vol. 27, no. 8, pp. 083 010–083 010, 2022.

[51] M. Schweiger, S. R. Arridge, and I. Nissilä, "Gauss–Newton method for image reconstruction in diffuse optical tomography," *Physics in Medicine & Biology*, vol. 50, no. 10, p. 2365, 2005.

[52] K. D. Andersen, E. Christiansen, A. R. Conn, and M. L. Overton, "An efficient primal-dual interior-point method for minimizing a sum of euclidean norms," *SIAM Journal on Scientific Computing*, vol. 22, no. 1, pp. 243–262, 2000.

SUPPLEMENTARY MATERIAL

A. Radiative transform equation for QPAT

For QPAT, the time-independent RTE in $r \in \Omega \subset \mathbb{R}^n$ ($n = 2$ or 3) with given scattering μ_s and absorption μ_a coefficients can be written as

$$\begin{aligned}&\hat{s} \cdot \nabla \phi(r, \hat{s}) + (\mu_s + \mu_a)\, \phi(r, \hat{s}) \\ &= \mu_s \int_{S^{n-1}} \Theta\left(\hat{s} \cdot \hat{s}'\right) \phi\left(r, \hat{s}'\right) \mathrm{d}\hat{s}' + q(r, \hat{s}), \quad r \in \Omega,\end{aligned}$$

where $\Theta\left(\hat{s} \cdot \hat{s}'\right)$ is the scattering phase function and $q(r, \hat{s})$ is the source in Ω, $\partial\Omega$ is the domains boundary, and $\hat{s} \in S^{n-1}$ denotes a unit vector in the direction of interest. With the radiance $\phi(r, \hat{s})$, we can compute the photon fluence as $\Phi(r) = \int_{S^{n-1}} \phi(r, \hat{s}) \mathrm{d}\hat{s}$

For modeling biological tissues, the scattering phase function is commonly chosen to be the Henyey–Greenstein scattering function. For QPAT, the boundary condition of RTE is often formed by assuming that at the boundary $\partial\Omega$, photons travel in an inward direction only at the source positions.

B. Finite element matrices of diffuse approximation

By choosing Henyey–Greenstein scattering function and assuming a diffusive region $r \in \Omega \subset \mathbb{R}^n$ ($n = 2$ or 3) with absorption μ_a and scattering μ_s coefficients, the diffusion approximation is given by

$$-\nabla \cdot \kappa(r) \nabla \Phi(r) + \mu_a(r) \Phi(r) = q_0(r), \quad r \in \Omega,$$

where q_0 is the light source in Ω, the parameter $\kappa(r) = (n(\mu_a(r) + \mu_s'(r)))^{-1}$ is the diffusion coefficient where $\mu_s' = (1-g)\mu_s$ is the reduced scattering coefficient with anisotropy parameter $-1 < g < 1$. A boundary condition for the DA can be formed by assuming that the total inward-directed photon current is zero at the boundary. Moreover, including the possible refraction mismatch between Ω and its surrounding medium, the boundary condition for DA can be derived to be

$$\Phi(r) + \frac{1}{2\gamma} \kappa(r) A \frac{\partial \Phi(r)}{\partial \hat{n}} = \begin{cases} \frac{I(r)}{\gamma}, & r \in \rho \\ 0, & r \in \partial\Omega \backslash \rho, \end{cases},$$

where $\hat{n}$ is the outward unit normal vector, $I(r)$ is the boundary photon flux at the source positions $\rho \in \partial\Omega, \gamma$ is a dimension-dependent constant with values $1/\pi$ in $\mathbb{R}^2$ and 1/4 in $\mathbb{R}^3$. The parameter A accounts for the internal reflection at the boundary $\partial\Omega$. If no boundary refraction occurs, A corresponds to 1.

Assume that the domain is separated into P non-overlapping elements with N grid coordinates, and that the optical parameters in Ω are represented as

$$\mu_a(r) \approx \sum_{t=1}^{N} \mu_{a_t} \varphi_t(r), \quad \mu_s(r) \approx \sum_{t=1}^{N} \mu_{s_t} \varphi_t(r), \qquad (1)$$

with the chosen finite element basis φ_t. The finite element approximation for the diffuse approximation can then be written as

$$(M + C + R)\Phi = Q,$$

where

$$\begin{aligned}M(i,j) &= \sum_{t=1}^{N} \frac{1}{n(\mu_{a_t} + \mu_{s_t}')} \int_{\Omega} \varphi_t(r) \nabla \varphi_i(r) \cdot \nabla \varphi_j(r) \mathrm{d}r \\ C(i,j) &= \sum_{t=1}^{N} \mu_{a_t} \int_{\Omega} \varphi_t(r) \varphi_i(r) \varphi_j(r) \mathrm{d}r \\ R(i,j) &= \int_{\partial\Omega} \frac{2\gamma}{A} \varphi_i(r) \varphi_j(r) \mathrm{d}S \\ Q(i) &= \int_{\partial\Omega} \frac{2I(r)}{A} \varphi_i(r) \mathrm{d}S.\end{aligned}$$

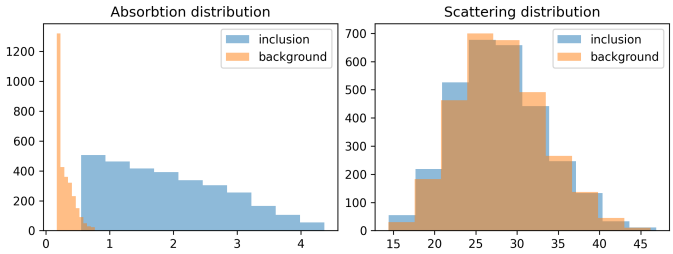

Fig. S1. *Digital twin problem*: Distribution of the simulated absorption coefficient (left) and scattering coefficient (right) values for both the background material (orange) and the inclusion materials (blue). While there is a distinct difference between the background and inclusion materials on the absorption values, there is no difference in the scattering distribution. The y-axes shows the bin frequencies when drawing 10,000 samples. The x-axes show the absorption and scattering coefficient in units of cm^{-1}.