Monotonicity Prior for Cloud Tomography
Supplementary Material

Tamar Loeub[1], Aviad Levis[2], Vadim Holodovsky[1], and Yoav Y. Schechner[1]

[1] Viterbi Faculty of Electrical Engineering,
Technion - Israel Institute of Technology, Haifa, Israel
[2] Computing and Mathematical Sciences Department,
California Institute of Technology, Pasadena, CA, USA
tamarloeub@gmail.com, aviad.levis@gmail.com,
vholod@ef.technion.ac.il, yoav@ee.technion.ac.il

Abstract. This is a supplementary document to the main manuscript. Here we provide the mathematical derivation of the gradient of the image formation model and additional details about run times.

1 Outline

This supplementary material contains two parts. The first part (Secs. 2 and 3) details the mathematical derivation of the gradient of the image fitting term, and the prior. The second part, Sec. 4, provides additional information regarding the simulations, i.e., run time, figures, and framework, and configuration details. Moreover, we present an additional test and application: rendering a new viewpoint from the reconstructed medium.

2 Gradient of the Image Fitting Term

We now describe the calculation of the gradient of the image model. Recall Eq. (14) from the main manuscript, which is the analytical formulation of a clear signal at detector d. It is the expectation of a photon to reach d, multiplied by the number of photons ($N_{\rm phot}$) generated at the source,

$$i_d(\boldsymbol{\beta}) = N_{\rm phot} \int_{L_d} P(\mathcal{L}) I_B(\mathcal{L}) d\mathcal{L}. \tag{1}$$

Here $P(\mathcal{L})$ is the probability density of a general path $\mathcal{L} \in L_d$. Index a photon by p, and an interaction point by b. The Monte-Carlo (MC) estimation of the signal at detector d is

$$i_d(\boldsymbol{\beta}) \approx \sum_{p=1}^{N_{\rm phot}} \sum_{b=1}^{B} I_b^{\rm le} \left[\mathcal{L}(d,p)\right] \ , \tag{2}$$

where $I_b^{\rm le} \left[\mathcal{L}(d,p)\right]$ is given in Eq. (17) of the main manuscript.

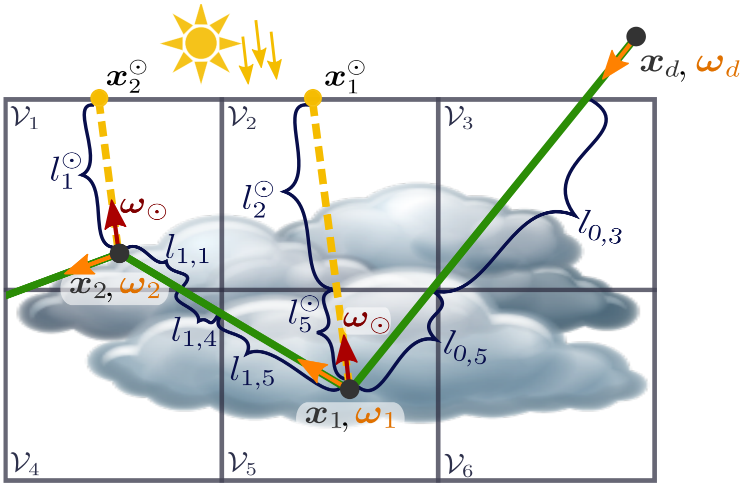

Fig. 1: Illustration of a photon path from a detector through the medium. The gradient is calculated according to the process described in Sec. 2 of this supplementary document, and specifically, Eq. (19) herein.

Note that while $P(\mathcal{L})$ is not needed explicitly for rendering, it is useful to associate a probability to a given path. The usefulness is clear when approaching the inverse problem. Following the definition of the transmittance function $a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right)$, in Eq. (16) of the main manuscript, associate with a scattering event at $\boldsymbol{x}_b$ a probability $a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right) f(\theta_{b-1,b})$. Here, $f(\theta_{b-1,b})$ is the phase function of the scattering angle $\theta_{b-1,b}$. Each scattering event and each extinction sampling is independent. Hence, inspired by [1], the probability P associated with a general path $\mathcal{L}$ is

$$P(\mathcal{L}) = \prod_{b=1}^{B-1} a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right) \beta_b f(\theta_{b-1,b}) \ . \tag{3}$$

Note the different forms of Eq. (1) and Eq. (2) above:

- The probability $P(\mathcal{L})$ in the integral is eliminated during summation, because path probability dictates the MC sampling in the sum.
- Point $\boldsymbol{x}_B$ is not expected to reach the sun, when sampling a small number of photons. Hence the signal in the summation relies on local estimation from all nodes, in lieu of $I(B)$.

We summarize the approximation of MC radiative transfer (RT) using the following operations:

$$I_B \underset{\text{becomes}}{\Longrightarrow} \sum_{b=1}^{B} I_b^{\text{le}}\left[\mathcal{L}(d,p)\right] \quad ; \quad N_{\text{phot}} \int_{L_d} P(\mathcal{L})(\cdot) d\mathcal{L} \underset{\text{becomes}}{\Longrightarrow} \sum_{p=1}^{N_{\text{phot}}} (\cdot) \ . \tag{4}$$

The optimization problem described in the paper is solved iteratively using stochastic gradient descent (SGD). This requires, specifically, estimation of the gradient of $F(\boldsymbol{\beta})$. A component of the gradient corresponding to voxel v is given by

$$\frac{\partial F(\boldsymbol{\beta})}{\partial \beta_v^{\mathrm{c}}} = \sum_{d=1}^{N_{\mathrm{detectors}}} \left[i_d(\boldsymbol{\beta}) - i_d^{\mathrm{measured}}\right] \frac{\partial i_d(\boldsymbol{\beta})}{\partial \beta_v^{\mathrm{c}}} \ . \tag{5}$$

We now assess $\partial i_d(\boldsymbol{\beta})/\partial \beta_v^{\mathrm{c}}$, i.e., how image pixels change in response to an infinitesimal perturbation of the medium $\boldsymbol{\beta}$. Changes in the medium generally lead to new paths being sampled as described in the paper. However, it is computationally demanding to sample new paths per degree of freedom of $\boldsymbol{\beta}$. To derive the gradient using existing paths, we use Eq. (1) for differentiation:

$$\frac{\partial i_d(\boldsymbol{\beta})}{\partial \beta_v^{\mathrm{c}}} = N_{\mathrm{phot}} \int_{L_d} \frac{\partial [P(\mathcal{L}) I_B(\mathcal{L})]}{\partial \beta_v^{\mathrm{c}}} d\mathcal{L} \ \ . \tag{6}$$

The term inside the integral of Eq. (6) above can be written

$$\frac{\partial [P(\mathcal{L}) I_B(\mathcal{L})]}{\partial \beta_v^{\mathrm{c}}} = \left[\frac{I_B(\mathcal{L})}{P(\mathcal{L})} \frac{\partial P(\mathcal{L})}{\partial \beta_v^{\mathrm{c}}} + \frac{\partial I_B(\mathcal{L})}{\partial \beta_v^{\mathrm{c}}} \right] P(\mathcal{L}) \ . \tag{7}$$

Then, applying the transformations of Eq. (4) above, the sampled gradient is assessed by

$$\frac{\partial i_d}{\partial \beta_v^{\mathrm{c}}} \approx \sum_{p=1}^{N_{\mathrm{phot}}} \sum_{b=1}^{B} \frac{\partial i_d(p,b)}{\partial \beta_v^{\mathrm{c}}} \ , \tag{8}$$

where

$$\frac{\partial i_d(p,b)}{\partial \beta_v^{\mathrm{c}}} = \frac{I_b^{\mathrm{le}}\left[\mathcal{L}(d,p)\right]}{P(\mathcal{L})} \frac{\partial P(\mathcal{L})}{\partial \beta_v^{\mathrm{c}}} + \frac{\partial I_b^{\mathrm{le}}\left[\mathcal{L}(d,p)\right]}{\partial \beta_v^{\mathrm{c}}} \ . \tag{9}$$

Therefore,

$$\frac{\partial i_d(p,b)}{\partial \beta_v^{\mathrm{c}}} = I_b^{\mathrm{le}}\left[\mathcal{L}(d,p)\right] \frac{\partial \log[P(\mathcal{L})]}{\partial \beta_v^{\mathrm{c}}} + \frac{\partial I_b^{\mathrm{le}}\left[\mathcal{L}(d,p)\right]}{\partial \beta_v^{\mathrm{c}}} \ . \tag{10}$$

We now derive the terms of Eq. (10) above. In calculating I_b^{le}, the only term explicitly dependent on β_v^{c} is $a\left(\overline{\boldsymbol{x}_b \boldsymbol{x}_b^{\odot}}\right)$. Generally, the line segment $\overline{\boldsymbol{x}_b \boldsymbol{x}_b^{\odot}}$ traverses several voxels, as illustrated in Fig. 1 above. Let $\mathcal{V}_v$ be the domain of voxel v. The intersection of voxel v with $\overline{\boldsymbol{x}_b \boldsymbol{x}_b^{\odot}}$ is of length

$$l_v^{\odot} = \left| \overline{\boldsymbol{x}_b \boldsymbol{x}_b^{\odot}} \cap \mathcal{V}_v \right| \ . \tag{11}$$

Notice that if $\overline{\boldsymbol{x}_b, \boldsymbol{x}_b^{\odot}} \cap \mathcal{V}_v = \varnothing$, then $l_v^{\odot} = 0$. Using Eq. (11) above and Eq. (16) of the main manuscript, the transmittance a is

$$a\left(\overline{\boldsymbol{x}_b \boldsymbol{x}_b^{\odot}}\right) = \prod_v \exp\left[-l_v^{\odot}\left(\beta_v^{\mathrm{a}} + \beta_v^{\mathrm{c}}\right)\right] \ . \tag{12}$$

Following Eqs. (10,12) above,

$$\frac{\partial I_b^{\mathrm{le}}\left[\mathcal{L}\left(d,p\right)\right]}{\partial \beta_v^{\mathrm{c}}} = -l_v^{\odot} I_b^{\mathrm{le}}\left[\mathcal{L}\left(d,p\right)\right]. \tag{13}$$

Using Eq. (3) above,

$$\frac{\partial \log\left[P\left(\mathcal{L}\right)\right]}{\partial \beta_v^{\mathrm{c}}} = \sum_{b=1}^{B-1}\left[\frac{\partial \log\left[a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right)\right]}{\partial \beta_v^{\mathrm{c}}} + \frac{\partial \log(\beta_b)}{\partial \beta_v^{\mathrm{c}}}\right]. \tag{14}$$

Here we used the fact that given a scattering particle (air or cloud droplet) at $\boldsymbol{x}_b$, the phase function $f(\theta_{b-1,b})$ is independent of β_v^{c}.

Generally, the line segment denoted $\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}$ traverses several voxels, as illustrated in Fig. 1 above. The length of the intersection of a voxel v with the set $\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}$ is

$$l_{b-1,v} = \left|\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b} \cap \mathcal{V}_v\right| \ . \tag{15}$$

Similarly to Eq.(11) above, $l_{b-1,v} = 0$ if $\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b} \cap \mathcal{V}_v = \varnothing$. Using Eq. (15) above and Eq. (16) of the main manuscript, the transmittance is

$$a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right) = \prod_v \exp\left[-l_{b-1,v}\left(\beta_v^{\mathrm{a}} + \beta_v^{\mathrm{c}}\right)\right] \ , \tag{16}$$

hence

$$\frac{\partial \log\left[a\left(\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}\right)\right]}{\partial \beta_v^{\mathrm{c}}} = -l_{b-1,v} \ . \tag{17}$$

Most line segments $\overline{\boldsymbol{x}_{b-1}\boldsymbol{x}_b}$ cross relatively few voxels. Thus, the output of Eq. (17) above is sparse over the domain.

The term $\partial\log(\beta_b)/\partial\beta_v^{\mathrm{c}}$ in Eq. (14) above is relevant only to the particular voxel at which scattering occurs. Moreover, air density is known irrespective of β_v^{c}. Scattering by an air molecule yields a null value of $\partial\log(\beta_b)/\partial\beta_v^{\mathrm{c}}$. Consequently,

$$\frac{\partial \log(\beta_b)}{\partial \beta_v^{\mathrm{c}}} = \begin{cases} 1/\beta_v^{\mathrm{c}} & \boldsymbol{x}_b \in \mathcal{V}_v \ \ \& \ \ \text{scatter is by a cloud droplet} \\ 0 & \text{else} \end{cases} . \tag{18}$$

Compounding Eqs. (10,13,14,17,18) above,

$$\frac{\partial i_d(\boldsymbol{x})}{\partial \beta_v^{\mathrm{c}}} \approx -\sum_{p=1}^{N_{\mathrm{phot}}}\sum_{b=1}^{B} I_b^{\mathrm{le}}\left[\mathcal{L}(d,p)\right] \cdot \left(l_v^{\odot} + l_{b-1,v} - \begin{cases} \frac{1}{\beta_v^{\mathrm{c}}} & \boldsymbol{x}_b \in \mathcal{V}_v \ \ \& \ \ \text{scattered by a cloud} \\ 0 & \text{else} \end{cases}\right). \tag{19}$$

3 Gradient of The Prior

Recall the formulation of the prior,

$$R(\boldsymbol{\beta}^{\mathrm{c}}) = \frac{1}{2}\mu_1 \|\mathbf{A}\boldsymbol{\beta}^{\mathrm{c}}\|_2^2 + \mu_2 R_{\mathrm{mon}}(\boldsymbol{\beta}^{\mathrm{c}}) \quad , \tag{20}$$

where

$$R_{\mathrm{mon}}(\boldsymbol{\beta}^{\mathrm{c}}) = -\mathbf{1}^{\intercal} \tanh\left(c_1 \mathbf{D}_z \boldsymbol{\beta}^{\mathrm{c}}\right) \tanh\left(c_2 \boldsymbol{\beta}^{\mathrm{c}}\right) \mathbb{1}\left\{\boldsymbol{\beta}^{\mathrm{c}} > 0\right\}. \tag{21}$$

The gradient of the regularization term is

$$\frac{\partial R(\boldsymbol{\beta}^{\mathrm{c}})}{\partial \boldsymbol{\beta}^{\mathrm{c}}} = \mu_1 \mathbf{A}^{\intercal}\mathbf{A}\boldsymbol{\beta}^{\mathrm{c}} + \mu_2 \frac{\partial R_{\mathrm{mon}}(\boldsymbol{\beta}^{\mathrm{c}})}{\partial \boldsymbol{\beta}^{\mathrm{c}}} \ . \tag{22}$$

The corresponding gradient of the monotonicity term is

$$\begin{aligned} \frac{\partial R_{\mathrm{mon}}(\boldsymbol{\beta}^{\mathrm{c}})}{\partial \boldsymbol{\beta}^{\mathrm{c}}} = -\Bigg[& c_1 \mathbf{D}_z \cosh^{-2}\left(c_1 \mathbf{D}_z \boldsymbol{\beta}^{\mathrm{c}}\right) \tanh\left(c_2 \boldsymbol{\beta}^{\mathrm{c}}\right) \\ & + c_2 \tanh\left(c_1 \mathbf{D}_z \boldsymbol{\beta}^{\mathrm{c}}\right) \cosh^{-2}\left(c_2 \boldsymbol{\beta}^{\mathrm{c}}\right) \Bigg] \mathbb{1}\left\{\boldsymbol{\beta}^{\mathrm{c}} > 0\right\}. \end{aligned} \tag{23}$$

Here $\cosh^{-2}(\mathbf{q})$ is a diagonal matrix. The diagonal is defined by the squared-reciprocal of the hyperbolic-cosine of each element of the argument vector. Recall that we use SGD to solve the optimization problem in Eq. (5) of the main manuscript,

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta} \geq 0} \left[F(\boldsymbol{\beta}) + R(\boldsymbol{\beta})\right] \ . \tag{24}$$

After each SGD iteration, we clip negative values of $\boldsymbol{\beta}^{\mathrm{c}}$.

4 Additional Details about the Simulations

As described in Sec. 6 of the main manuscript, estimation is done in coarse-to-fine stages. We used six stages in scene `F`, and eight stages in scene `C`. The number of voxels and pixels used in each stage of scene `F`, for example, are plotted in Fig. 2 below.

The cost function in Eq. (24) above is a combination of a data fitting term $F(\boldsymbol{\beta})$, and a regularization term $R(\boldsymbol{\beta})$. Moreover, recall the local reconstruction error measure,

$$\epsilon = \frac{\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{\mathrm{true}}\|_1}{\|\boldsymbol{\beta}^{\mathrm{true}}\|_1} \ . \tag{25}$$

This measure ϵ compares the iteratively-estimated model to the ground truth.

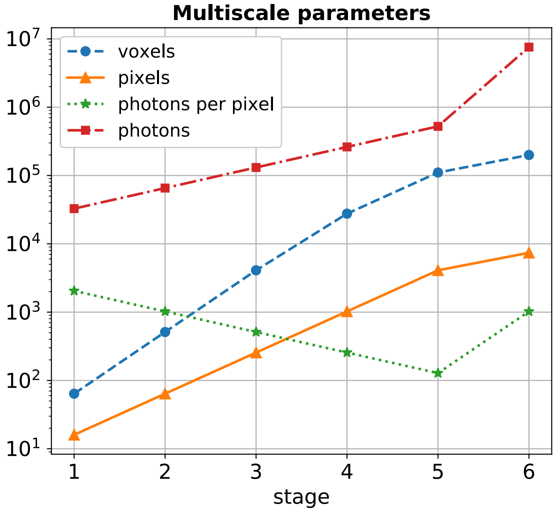

Fig. 2: The coarse-to-fine parameters as a function of the stages of the algorithm. This figure displays the increasing number of voxels, pixels, photons per pixel, and the overall number of photons.

We plot two criteria:

- The cost function from Eq. (24) above, as it evolves during SGD.
- The error measure Eq. (25) above, evolving during SGD.

Both are plotted as a function of time, during runs on our server. Scene `F` ran on a two CPUs Intel® Xeon® Platinum 8175M 2.40GHz with 24 cores each. And scene `C` ran on a two CPUs Intel® Xeon® Processor E5-2680 v4 2.40GHz with 14 cores each.

Overall, the criteria are plotted in Fig. 3 and Fig. 4 herein. Termination of each stage is color-marked in the plots here using a vertical colored line. Run time is displayed in log-scale. Fig. 3 herein corresponds to scene `F`. Fig. 4 herein corresponds to scene `C`.

Fig. 5 herein presents the effect of the monotonicity prior, beyond a smoothness prior (scene `F`). The left plot presents how the cost function Eq. (24) above evolves during the whole SGD process. The right figure presents a zoom-in of the error measure Eq. (25) above in the last stage of the coarse-to-fine solution. The orange lines present the cost Eq. (24) above, and error Eq. (25) above without the monotonicity prior, i.e., $R(\boldsymbol{\beta}^{\rm c}) = \frac{1}{2}\mu_1\|\mathbf{A}\boldsymbol{\beta}^{\rm c}\|_2^2$. The blue lines present these respective costs when solving the full optimization problem, as formulated in the main manuscript, including the monotonicity prior. Here the regularization term is $R(\boldsymbol{\beta}^{\rm c}) = \frac{1}{2}\mu_1\|\mathbf{A}\boldsymbol{\beta}^{\rm c}\|_2^2 + \mu_2 R_{\rm mon}(\boldsymbol{\beta}^{\rm c})$. As stated in the main manuscript, the monotonicity prior improves the quality and convergence.

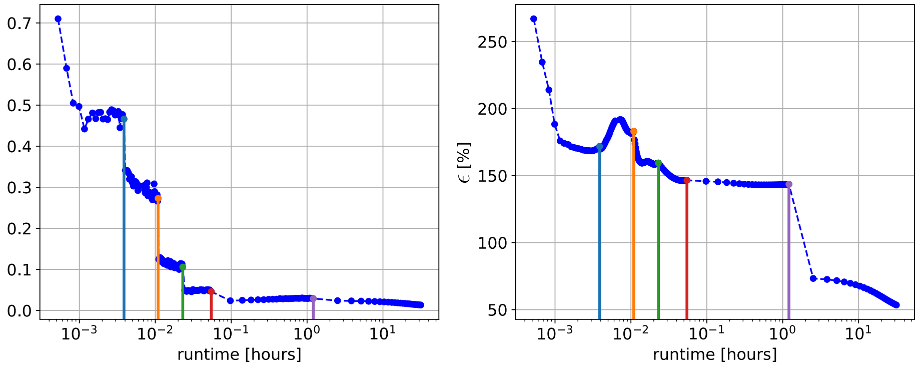

Fig. 3: Recovery of scene `F`. These plots present [Left] the cost function Eq. (24) above, and [Right] the error function ϵ Eq. (25) above, as functions of run time.

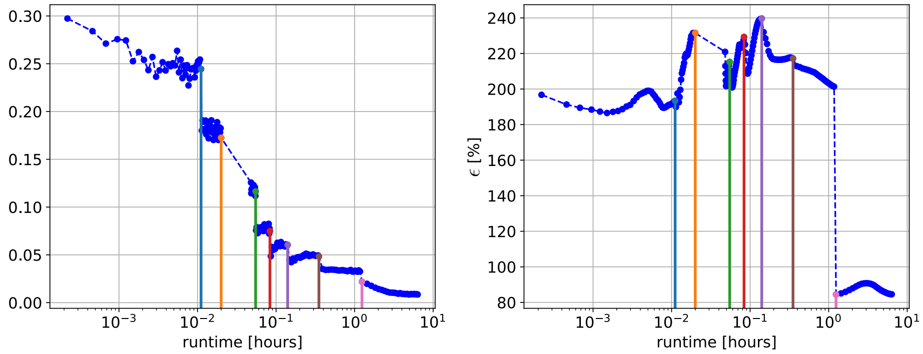

Fig. 4: Recovery of scene `C`. These plots present [Left] the cost function Eq. (24) above, and [Right] the error function ϵ Eq. (25) above, as functions of run time. The hierarchical solution enables a faster convergence to a better solution.

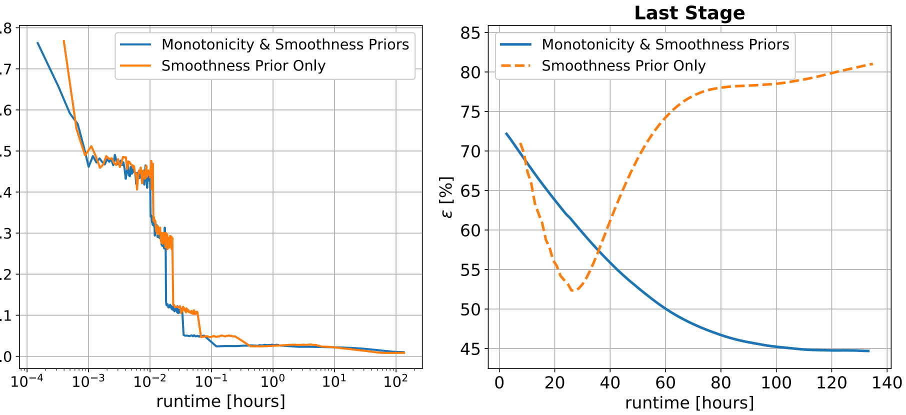

Fig. 5: This figure illustrates the recovery of scene `F`, using exclusively the smoothness prior (orange line), or using monotonicity compounded with smoothness (blue line), as formulated in Eq. (24) above. [Left] The cost function evaluation, as a function of run time. [Right] The error ϵ during the last stage, as a function of run time on our server. The error ϵ significantly improves using the monotonicity prior.

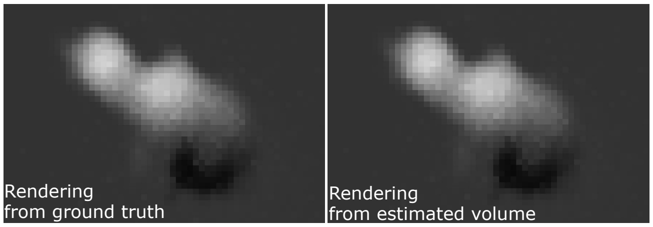

Fig. 6: Rendering a new viewpoint, 50.4° off-nadir. This viewpoint was not used during the reconstruction of scene `C`. This image was not an input to tomographic recovery. [Left] Image rendered from the ground-truth medium. [Right] Image rendered from the estimated medium.

Finally, Fig. 6 presents an application of scattering tomography. Here we render a new viewpoint, that was not included in the input measurements of scene `C`. For comparison, we render an image using the ground truth medium. Then we render the same viewpoint using the estimated medium.

References

1. Gkioulekas, I., Zhao, S., Bala, K., Zickler, T., Levin, A.: Inverse volume rendering with material dictionaries. ACM Trans. Graph. **32**(6), 162 (2013)