3DeepCT: Learning Volumetric Scattering Tomography of Clouds

Yael Sde-Chen, Yoav Y. Schechner, Vadim Holodovsky
Viterbi Faculty of Electrical and Computer Engineering,
Technion - Israel Institute of Technology, Haifa, Israel
yael.sde.chen@gmail.com, yoav@ee.technion.ac.il, vholod@ef.technion.ac.il

Eshkol Eytan
Department of Earth and Planetary Science,
The Weizmann Institute of Science, Rehovot, Israel
eshkol.eytan@weizmann.ac.il

Abstract

We present 3DeepCT, a deep neural network for computed tomography, which performs 3D reconstruction of scattering volumes from multi-view images. The architecture is dictated by the stationary nature of atmospheric cloud fields. The task of volumetric scattering tomography aims at recovering a volume from its 2D projections. This problem has been approached by diverse inverse methods based on signal processing and physics models. However, such techniques are typically iterative, exhibiting a high computational load and a long convergence time. We show that 3DeepCT outperforms physics-based inverse scattering methods, in accuracy, as well as offering orders of magnitude improvement in computational run-time. We further introduce a hybrid model that combines 3DeepCT and physics-based analysis. The resultant hybrid technique enjoys fast inference time and improved recovery performance.

1. Introduction

There is increasing effort to develop deep neural networks (DNNs) for reconstructing three dimensional (3D) shapes or projections of opaque objects. These DNNs use either explicit outer-shell or volumetric representations [7, 11, 12, 13, 19, 24, 28, 35, 53, 58, 59, 67, 78, 79, 80, 82].

However, there is still a significant gap in advancing reconstruction of 3D *heterogeneous volumetric translucent* fields, such as the atmosphere (Fig. 1). 3D heterogeneous translucent objects are reconstructed, essentially, using tomographic data. Radiation that propagates through the medium yields a set of multi-view two dimensional (2D) radiometric images. Analysis retrieves from the data the volumetric spatial distribution of material density in 3D.

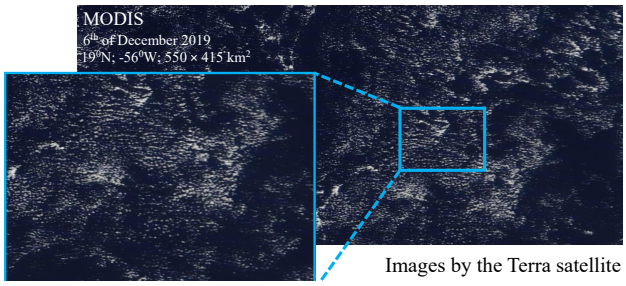


Figure 1. Shallow cumulus cloud fields [76].

This is computed tomography (CT). It is used extensively in biomedical imaging and earth sciences [3, 5, 23, 37, 57, 66]

DNNs advance medical CT [29, 34, 65, 72, 77]. Nevertheless, most CT modalities are based on a linear image formation model. Linear models can be solved well using established signal processing methods, including optimization of a convex functional, without requiring learning. It may be argued that 3D tomography does not lend itself easily to current DNNs. The reason is that DNNs require big training data, but it is extremely difficult to obtain sufficient ground-truth data of volumetric heterogeneous translucent objects.

Reconstructing such objects poses a serious challenge which is worth tackling by DNNs. We believe this challenge and opportunity occur when these conditions are met. **(1)** The tomographic model is very complex: nonlinear, not unimodal. Then, methods of linear-CT analysis cannot apply. Moreover, optimization-based estimation is very slow, unscalable and too dependent on initialization. **(2)** Scalability is critically needed to analyze huge 3D fields. **(3)** While the imaging model is nonlinear, it is continuous: an infinitesimal change of the medium continuously affects the image data, and vice versa. **(4)** There is a physics-based way to generate a large and diverse database. Under these

conditions, a DNN can realistically learn to express the richness of large translucent fields, and the physical processes that generate both 3D translucent objects and their images. Moreover, the inference speed offered by a trained DNN can significantly overtake explicit physics-based optimization.

We pose a problem that should greatly benefit from a DNN for CT. The problem is imaging of a very large random 3D spatially heterogeneous *scattering* medium [2, 10, 25, 46, 55, 68, 69, 74]: the atmosphere. In computer vision, imaging through a scattering medium has usually been related to dehazing [33, 51], defogging [43], underwater descattering [2, 71], or recovering properties of a medium, assuming its spatial uniformity [10]. Here, however the focus is on imaging of *clouds* [50, 84]. Clouds have interactions with the global climate system which are not well understood. This leads to major uncertainties in climate predictions [4, 6, 8]. This is a major motivation to properly sense these volumetric translucent objects internally.

Clouds are usually highly heterogeneous. Furthermore, multi-view images of clouds are governed by 3D radiative transfer (RT) [20, 31, 64]: a nonlinear, recursive forward model, which expresses arbitrary multiple scattering in 3D. Inverting this model is highly complex. Common methods in remote sensing try to bypass this complexity by imposing a model [60, 61] where clouds are horizontally uniform, infinitely broad, and RT is roughly vertical (one dimensional). This is inconsistent with nature, particularly when clouds are small. Recent work in computer vision introduced 3D *scattering tomography* [1, 16, 25, 26, 32, 40, 45, 46, 47] and proposed it as a viable path to study clouds. However, it is still slow and has not been scaled.

Ref. [22] shows that thick clouds have a veiled core, to which images are insensitive. Relying only on images may lead to major errors in a veiled core: the data term is ill-conditioned in thick clouds. Regularization should be valuable. A learning-based system may address this: Using examples as it trains, the system implicitly learns a prior of the nature and structure of clouds. 3DeepCT trains on physically realistic clouds, expressing both RT (fidelity) and the nature of clouds (regularization). We thus believe that a neural network (NN) is the way to better condition cloud tomography.

Our proposed learning-based system, *3DeepCT* infers 3D scattering-CT. While its results have quality which is comparable to explicit physics-based methods, it appears to run *five orders of magnitude faster*. Moreover, after training, it can potentially be scalable to broad cloud fields, exploiting GPU parallelism. We show how natural properties of clouds in cloud fields lead to the architecture of 3DeepCT. This includes a convolutional neural network (CNN) based on 2D convolutions, the size of its receptive field and layer-depth, and avoidance of dimensionality reduction. Furthermore, we provide an approach to train this system using rigorous physics-based generation of simulated fields and images.

2. Theoretical Background

We seek to recover the 3D volumetric optical parameters of a medium, particularly 3D clouds. We now provide background on inverse problems and focus on this domain.

2.1. 3D Radiative Transfer

Let $\boldsymbol{x}$ denote a 3D location and $\boldsymbol{\omega}$ denote a 3D direction unit vector. This vector expresses the propagation direction of radiation. The direction changes from $\boldsymbol{\omega}'$ to $\boldsymbol{\omega}$ by scattering. This change is set by the dimensionless *scattering phase function* at $\boldsymbol{x}$, denoted $p(\boldsymbol{x}, \boldsymbol{\omega} \cdot \boldsymbol{\omega}')$. The phase function is normalized, as it is equivalent to the probability density of scattering between directions $\boldsymbol{\omega}'$ and $\boldsymbol{\omega}$.

When radiation interacts with a particle, the radiation can be either scattered or absorbed. Scattering and absorption have, respectively, relative probabilities ϖ and $1-\varpi$, where $0 \leq \varpi \leq 1$ is the *single scattering albedo* of the particle. In visible light, for both air molecules and cloud water droplets, $\varpi \sim 1$, i.e., absorption is negligible.

The *extinction coefficient* around location $\boldsymbol{x}$ is $\beta(\boldsymbol{x})$. There, interaction of any kind along an infinitesimal distance $\mathrm{d}\boldsymbol{x}$ has probability $\beta(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$. Transmittance on a straight line between two points $\boldsymbol{x}_1, \boldsymbol{x}_2$ is

$$T(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left[-\int_{\boldsymbol{x}_1}^{\boldsymbol{x}_2} \beta(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right]. \tag{1}$$

The richness of multiple scattering, including various paths to interaction, possible absorption or scattering events, and scattering to different angles, embodies RT. It is expressed [9] by a set of recursively coupled equations:

$$\begin{aligned} I(\boldsymbol{x}, \boldsymbol{\omega}) =& I(\boldsymbol{x}_0, \boldsymbol{\omega}) T(\boldsymbol{x}_0, \boldsymbol{x}) \\ &+ \int_{\boldsymbol{x}_0}^{\boldsymbol{x}} J(\boldsymbol{x}', \boldsymbol{\omega}) \beta(\boldsymbol{x}') T(\boldsymbol{x}', \boldsymbol{x})\, \mathrm{d}\boldsymbol{x}', \\ J(\boldsymbol{x}, \boldsymbol{\omega}) =& \frac{\varpi(\boldsymbol{x})}{4\pi} \int_{4\pi} p(\boldsymbol{x}, \boldsymbol{\omega}\cdot\boldsymbol{\omega}')\, I(\boldsymbol{x}, \boldsymbol{\omega}')\, \mathrm{d}\boldsymbol{\omega}'. \end{aligned} \tag{2}$$

Here $I(\boldsymbol{x}, \boldsymbol{\omega})$ is the radiance field at each location and direction and $I(\boldsymbol{x}_0, \boldsymbol{\omega})$ is input radiance to the medium in direction $\boldsymbol{\omega}$ at boundary point $\boldsymbol{x}_0$ (boundary condition). The field $J(\boldsymbol{x}, \boldsymbol{\omega})$ is termed the *source function*.

Eq. (2) provides the radiance anywhere. Imaging samples this radiance field: then, $\boldsymbol{x}$ is the camera's center of projection, while $\boldsymbol{\omega}$ corresponds to a line of sight (LOS) that projects to a specific pixel in the camera at $\boldsymbol{x}$. Hence, the *forward model* $\mathcal{F}$ comprises two consecutive steps: (a) Run Eq. (2), which depends on the medium $\boldsymbol{\beta}$, and (b) sample the radiance at the camera locations and LOS that project to pixels in these cameras. This sampling is independent of $\boldsymbol{\beta}$.

For a given medium, Eq. (2) is computed by established methods, such as SHDOM [1, 18, 20] and Monte Carlo [15, 16, 32, 54, 56]. Moreover, in recent years, approximations to the Jacobian $\partial \mathcal{F}(\boldsymbol{\beta})/\partial \boldsymbol{\beta}$ have been derived by the computer vision and graphics communities [25, 26, 42, 45, 52, 83]. Here, $\boldsymbol{\beta}$ concatenates the extinction coefficients across all voxels to a vector. The vector length equals the number of voxels in the domain.

In clouds, Eqs. (1,2) are highly nonlinear in $\boldsymbol{\beta}$. In contrast, in medical (traditional) CT, the signal is dominated by linear dependency on $\boldsymbol{\beta}$ and $\approx 1/2$ the photons do not scatter [23], making SART viable, at least for initialization [23]. In clouds, the optical depth is $\mathcal{O}(10 \dots 100)$. Thus direct transmission of ground radiance and single-scattering events are negligible, and cannot be a realistic source of initialization, eg., using SART.

2.2. Inverse Problem

The forward model $\mathcal{F}(\boldsymbol{\beta})$ renders images given $\boldsymbol{\beta}$. The measured radiance image data are expressed in a vector $\boldsymbol{y}$. Its length is the product of the number of camera image pixels, viewpoints, spectral bands and possibly polarization channels. Estimation of $\boldsymbol{\beta}$ is an inverse problem. Often, it is formulated as an optimization problem:

$$\boldsymbol{\beta}^* = \underset{\boldsymbol{\beta}}{\arg\min}\, \mathcal{E}(\boldsymbol{\beta}) \,, \tag{3}$$

where $\mathcal{E}$ is a cost function. Specifically,

$$\mathcal{E}(\boldsymbol{\beta}) = ||\boldsymbol{y} - \mathcal{F}(\boldsymbol{\beta})||_2^2. \tag{4}$$

In scattering media, $\mathcal{F}$ is nonlinear in $\boldsymbol{\beta}$. Hence, Eq. (3) cannot be solved using linear-algebra tools of matrix inversion. Nevertheless, thanks to the approximate Jacobian mentioned in Sec. 2.1, an approximate gradient $\partial \mathcal{E}(\boldsymbol{\beta})/\partial \boldsymbol{\beta}$ is derived. This enables a practical solution to (3) using gradient-based methods.

Gradient-based optimization has two main problems. First, the solution significantly depends on an initial guess, because $\mathcal{F}(\boldsymbol{\beta})$ is non-linear in $\boldsymbol{\beta}$ and $\mathcal{E}(\boldsymbol{\beta})$ is not unimodal. Second, the problem is very difficult to scale [52], and it runs typically on small domains. Computation of $\mathcal{F}(\boldsymbol{\beta})$ and $\partial \mathcal{E}(\boldsymbol{\beta})/\partial \boldsymbol{\beta}$ is complex. It requires recursive evaluation of the fields I and J, relying on Eq. (2) as well as rendering of modeled projected images, during iterated optimization. We seek to mitigate this load, by designing a learning-based system for volumetric scattering CT.

2.3. Simulated Data and Noise

Learning-based systems for analysis of high-dimensional data and unknowns require a large training database. It is very difficult to obtain large databases for real-world, large heterogeneous volumetric objects. For example, in clouds, which are dynamic, a real-world ground-truth training database would require, for each real-world cloud, in-situ measurements of cloud droplets, simultaneously in $\mathcal{O}(10^5 - 10^6)$ voxels, using this number of airborne cloud-droplet sensors. It is doubtful if such a large distributed ground-truth in-situ system would ever exist. Even if it will exist, it would need to be scaled to sample a large database of clouds, in varying atmospheric and illumination conditions.

To overcome the practical absence of real-world labeled data, we train the tomographic analysis system using meticulous simulations. We explain the simulations that relate to clouds in Sec. 4. Our simulations of a perspective camera have a noise model based on the CMV4000 sensor [14]. The pixel size is $5.5 \times 5.5 \mathrm{micron}^2$, readout noise standard deviation is 13 electrons, the sensor dark current is $125 \mathrm{electrons/sec}$ at $25^{\mathrm{o}}\mathrm{C}$, full well of a pixel is 13,500 electrons. It uses 10bit quantization, thus the number of photo-electrons required to change a unit gray level is $g^{\mathrm{e}} = 13,500/2^{10}$. The exposure time is adjustable according to the radiance that reaches the camera such that the camera sensor reaches the full well. Additional information about the noise model are in the supplementary material.

2.4. Clouds in a Field

We test learned-tomography on atmospheric clouds. This medium has characteristics that affect our system.

Significance of warm clouds. Warm clouds are made of liquid water droplets. Clouds account for two thirds of Earth's albedo while warm shallow clouds constitute a major part of it. In visible light, this albedo forms the most significant part of clouds' radiative effect on Earth's energy balance. Warm clouds over the ocean have a particularly dramatic effect. Consider Fig. 1. The ocean is dark (highly absorbing) while clouds are white (highly scattering and reflecting). Thus, the difference between cloudy and non-cloudy regions has an extreme effect on Earth's energy balance. Due to their significance both in the optical signal of scattering and on climate, the *focus of our work here is on warm clouds over the ocean.*

Vertically thin, horizontally wide domain. There are marked differences between vertical and horizontal coordinates. Atmospheric domains of interest, and cloud fields in particular, can be thousands of kilometers wide. On the other hand, the vertical dimension is thin. Atmospheric pressure (approximately the density) falls exponentially with altitude, dropping to half sea-level pressure at $\approx 5 - 6$ km. Warm clouds have tops that are typically under ≈ 2 km. Hence, there is a difference of *orders of magnitude* between horizontal and vertical lengths. Moreover, clouds are created by vertical air currents driven by gravity and buoyancy of air parcels due to temperature gradients. In contrast, horizontal wind is driven by a subtle horizontal

pressure gradient, which may be null. For this reason, our *learning-based system is anisotropic, treating vertical and horizontal variations very differently*.

Horizontal Stationarity. As seen in Fig. 1, cloud fields maintain rather stationary statistics over long ranges [75]. This is particularly true for clouds on the ocean, far from land. Because the statistics are approximately *space invariant* (in the horizontal coordinates), our analysis system is space invariant as well: this naturally leads to a *convolutional architecture* when treating horizontal coordinates. The vertical coordinate is separate and not space invariant.

3. 3DeepCT

Here we present a system, 3DeepCT, for learning 3D scattering CT. Based on a DNN, 3DeepCT learns and then infers volumetric domains. *3DeepCT is faster by orders of magnitude*, relative to pure physics-based scene analysis. Its input is multi-view images of the scene, and its output is an estimated 3D heterogeneous $\hat{\boldsymbol{\beta}}$. As described in Sec. 2.3, it is challenging, to say the least, to empirically acquire a large training database for large heterogeneous volumetric media, which are typically dynamic. Hence, training is based on rigorous physics-based models of the true scenes $\{\boldsymbol{\beta}^{\text{true}}\}$, images and realistic noise. Using such a simulated database, learning is performed by minimizing a loss. The loss we use is the mean square error (MSE):

$$\text{Loss}_{\text{MSE}}(\boldsymbol{\beta}^{\text{true}}, \hat{\boldsymbol{\beta}}) = ||\boldsymbol{\beta}^{\text{true}} - \hat{\boldsymbol{\beta}}||_2^2 \tag{5}$$

3DeepCT is based on physics-based training data, hence in inference of test data, physics plays an implicit role. However, for scientific needs, it benefits that the final word is given to an explicit physics-based solution to the inverse-problem. A **hybrid system** can do this. Given a new data of an unknown scene, 3DeepCT can provide a solution which is comparable to explicit iterated physics-based inversion, yet doing so very fast. The solution obtained by 3DeepCT is then used as initialization for the much slower physics-based inversion. Only a few iterations would then be needed for explicit physics-based inversion.

These principles can apply to any type of computed tomography and any media. However, it benefits an expert system to be tailored to an imaging modality and the media being observed. Therefore, from this point on, we describe how the architecture and implementation of 3DeepCT is tailored to scattering CT of warm clouds.

3.1. Architecture Tailored to Cloud Fields

These architectural principles follow the natural traits of clouds, some of which are described in Sec. 2.4:

$\{1\}$ Due to the horizontal stationarity of cloud fields, both as volumetric objects and in images, 3DeepCT is a convolutional neural network (CNN). Convolution kernels are operated in the horizontal domain.

$\{2\}$ Data is in the form of multi-angular images, i.e., the input has *angular channels* corresponding to different viewpoints (view angles). To capture 3D information from multiview images, all angular channels participate. There is no preference to angular-neighborhood, contrary to a convolutional operation. Moreover, output clouds have a vertical structure expressed by *vertical channels*. As written in Sec. 2.4, cloud vertical structure is not stationary. Therefore in all layers of the DNN, convolution kernels operate *only* in 2D. The channels on the complementary coordinate, specifically angular channels in the input and vertical channels in the output, are *completely intertwined, per NN layer*.

$\{3\}$ Due to the effective thinness of the atmospheric domain (Sec. 2.4) and typically small number of viewpoints, having a completely intertwined architecture in the vertical or angular direction is not a computational burden. On the other hand, due to the very wide extent of cloud fields and high resolution of images, a horizontal CNN makes analysis scalable. Principles $\{1, 2\}$ are addressed directly by using the *Conv2D* operation at the entry stage of any layer of the 3DeepCT neural network.

$\{4\}$ A small warm cloud cell typically has size of hundreds of meters to a kilometer. Thus an output neuron needs to have some statistical dependency to the output of other neurons in that range. This dictates the minimum *receptive field* range R of the overall DNN. This determines the DNN *depth*, i.e, its number of layers. Let the camera horizontal resolution be r meters/pixel. In principle $\{3\}$, operations per layer are based on *Conv2D*, which uses kernels having a small 2D support, of length h pixels. The number of layers needs to be $L \sim R/(rh)$.

$\{5\}$ Usually, clouds are created by chaotic turbulent flow. Thus they are not highly structured objects as the human body. Consequently, clouds are not amenable to significant dimensionality reduction. Moreover, tomography tries to resolve voxels, whose number is comparable to the data size. Therefore, contrary to auto-encoder networks, we do not include dimensionality reduction (pooling) operations.

3.2. Architecture Simplicity

3DeepCT has a simple structure. Yet, it outperformed many alternatives that we tried and tested. Specifically, we tested deep unfolding by formulating an iterative proximal-gradient and unrolling it to construct a deep feed forward-NN (resembling the RNN of [49]). We further examined ResNet, U-Net, CNN with 3D convolution layers and elements from [53]. Inspired by [10], we incorporated physics-based rendering in training using loss of both 3D object and rendered images, but rendering all objects in each epoch takes too long to be practical. In all cases, the results proved to be inadequate. Performance results of U-Net and ResNet are shown in the supplementary material. Eventually, our study led us to the 3DeepCT structure described here. This

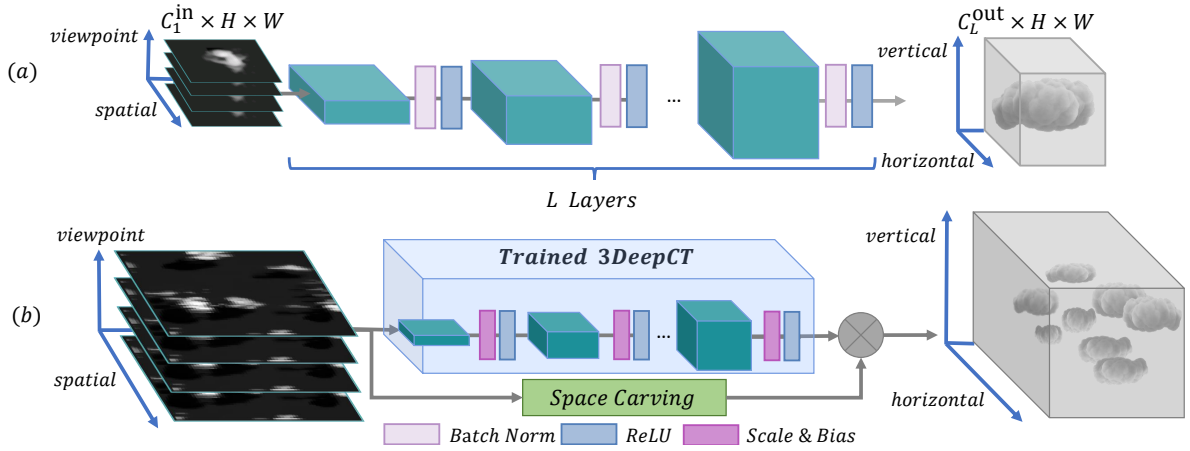


Figure 2. **3DeepCT neural network architecture:** A convolutional neural network receives multi-view satellite-images of a cloud as input and reconstructs the 3D extinction field of the cloud. (a) The architecture we use to train the network. The images are of single isolated clouds. (b) Illustration of the architecture for inference, where the input can be large images that capture a large cloud field. A space-carving mask is incorporated. A batch normalization layer degenerates to linear operations of scaling and bias.

may not be a coincidence: as we explained, the nature of our problem may suit this structure.

3.3. Implementation Details

The architecture is illustrated in Fig. 2. 3DeepCT has L layers, indexed $l = 1 \dots L$. In each layer, there are, respectively C_l^{in} and C_l^{out} input and output channels, where $C_l^{\text{in}} = C_{l-1}^{\text{out}}$ for $l \neq 1$. The number of viewpoints sets C_1^{in}, while C_L^{out} is the number of resolved vertical altitudes in the volume. Each input image (data per input channel) is of size $H \times W$ pixels. Since we have no dimensionality reduction, each layer in 3DeepCT maintains a 2D size of $H \times W$, and this is the horizontal size of the estimated volume domain. Moreover, $C_l^{\text{out}} \geq C_l^{\text{in}}$, gradually increasing or maintaining the number of channels per layer. Each layer l is composed of a Conv2D operator, followed by batch-normalization and ReLU activation function.

We found that inference significantly improved by *space-carving* (SC) [44] of the domain. SC is a known method to approximate a bounding hull of a 3D object. SC helps significantly also in physics-based inverse rendering, as reported in [46]. Clouds have significant contrast relative to the dark ocean background. Hence, a moderate cloud-mask is applied per input image, followed by SC. This yields a simple bounding constraint on which voxels may potentially have water droplets. An ablation study of the SC is described in the supplementary material.

We implemented 3DeepCT using the Pytorch framework [63]. Due to the convolution architecture, inference is possible at larger or smaller H, W. The network training and testing architectures are illustrated in Fig. 2. Its components and hyper-parameters were carefully examined and tuned by ablations studies (see the supplementary material). We used an NVIDIA GeForce RTX 2080 GPU, an Adam optimizer, StepLR scheduler (step size 10, gamma of 0.9), an initial learning rate of 1e-3, and batch size $b = 32$.

4. Data

4.1. Physics-based Clouds

Recall from Sec. 2.3 the practical infeasibility of obtaining an empirical real-world database of clouds. Thus, 3DeepCT trains using rigorous physical simulations. The simulations couple three components:

[A] Initial and environmental conditions (aerosols, wind, temperature, surface fluxes, humidity field etc.) are set using either the BOMEX [73] or CASS [85] setups. They are described in the supplementary material. Each setup yields a database having a domain size and resolution, as specified Table 1. The simulated domains are divided to voxels 50 m wide and 40 m thick.

[B] A dynamical large eddy simulation (LES), which solves the coupled equations of a turbulent atmosphere. LES is the main numerical tool for generating and studying clouds at the altitudes of relevance to our work [30, 62, 81]. Here we use the System for Atmospheric Modeling (SAM) [39]. SAM is a non-hydrostatic, inelastic model.

[C] Representation of the droplet's microphysics by a spectral model (HUJI SBM [21, 38]). It explicitly evolves physical equations of the processes that affect cloud droplet growth, yielding the size distribution of droplets per voxel. This distribution is sampled into 33 bins, logarithmically spread in the range $[2\ \mu\text{m}, 3.2\ \text{mm}]$. This leads to the ground-truth optical extinction coefficient β^{true} per voxel and the phase function, through Mie theory [27].

To enable representation and network expression of smaller optical depths, we constructed a CASS-Aux dataset from the CASS dataset. There, liquid water content of each

Dataset	Horizontal	Vertical	Voxels	Train	Evaluation	Test
BOMEX	$1.6\text{km} \times 1.6\text{km}$	1.2km	$32 \times 32 \times 32$	$4,800$	$1,200$	286
CASS	$3.2\text{km} \times 3.2\text{km}$	1.2km	$64 \times 64 \times 32$	$8,726$	$2,181$	$1,000$
CASS-Aux	$3.2\text{km} \times 3.2\text{km}$	1.2km	$64 \times 64 \times 32$	$3,744$	686	$1,000$

Table 1. Datasets specifications: horizontal and vertical extent of the volumes; number of voxels of the volume; number of samples in the training, evaluation and test sets.

Model#	Geometry	Dataset	L	$H \times W$	epochs	Training Duration [hours]	C_1^{in}	C_L^{out}
1	32 satellites	BOMEX	33	32×32	1000	11	32	32
2	10 satellites	BOMEX	33	32×32	1000	8	10	32
3	10 satellites	CASS	80	64×64	1000	48	10	32
4	10 satellites	CASS, CASS-Aux	80	64×64	800	14	10	32

Table 2. 3DeepCT models specifications: geometry from the geometries described in Sec. 4.2; dataset from Table 1 the model trained on; parameters L, H, W of the network; number of training epochs; training duration; $C_1^{\text{in}}, C_L^{\text{out}}$ parameters of the network.

CASS cloud is reduced by a factor of 10.

4.2. Physics-based Rendering

Given ground-truth clouds, rendering uses a physics-based RT solver (SHDOM), which has an open-source code [48]. Rendering is set per solar angle, viewing geometry and sensor characteristics. We separately trained versions of 3DeepCT for several distinct imaging geometries:

32 Viewpoints. A northbound string-of-pearls [41] formation of 32 satellites orbit at 600km altitude. Nearest-neighbor satellites are 100km apart. They view the same field in off-nadir angles $-75.2°$, $\pm 73.5°$, $\pm 71.7°$, $\pm 69.6°$, $\pm 67.4°$, $\pm 64.8°$, $\pm 62°$, $\pm 58.8°$, $\pm 55.2°$, $\pm 51°$, $\pm 46°$, $\pm 40.6°$, $\pm 34°$, $\pm 26°$, $\pm 18°$, $\pm 9°$, and $0°$. Each carries a perspective camera. The field of view of each camera is $0.22°$, corresponding to a ground footprint of $1.6\text{km} \times 1.6\text{km}$ at nadir , at 50 m ground resolution. The sensor noise characteristics correspond to the CMV4000 sensor [14]. Solar azimuth and zenith angles are $45°$ and $30°$, respectively.

10 Viewpoints. This geometry is motivated by the CloudCT space mission [70]. This mission plans 10 nanosatellites carrying perspective cameras, which will simultaneously image clouds in a multiview geometry. This geometry is visualized in the supplementary material. The parameters are similar to those of the **32 Viewpoints** geometry. Here, however, we use off-nadir angles $-46°$, $\pm 34°$, $\pm 26°$, $\pm 18°$, $\pm 9°$ and $0°$.

In total, we experimented with several models, each using a certain cloud database (Sec. 4.1) and geometry. Each model was then trained on a distinct data and network settings. These are summarised in Table 2.

5. Simulated Inference Results

We present inference (test) results of trained models, in different geometries. We compare four approaches:

A. **3DeepCT**. It runs on an NVIDIA GeForce RTX 2080 GPU, where it takes just *milliseconds* to reconstruct each cloud. This enables us to test hundreds of cloud samples.

B. **Physics-based inverse scattering**. This explicit optimized iterated fit of a physical model to the data is the state of the art [45, 46, 47]. It uses an SHDOM code from the public domain [48]. It is initialized by a default constant value of $\boldsymbol{\beta}^{\text{initial}} = 0.01$ globally for all the cloud's space-carving mask. To run this method we use 20 cores of Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz with 72 cores. The state-of-the-art approach does not exploit a GPU, as its complex calculations require CPUs. It takes about an *hour* to reconstruct each cloud in this system. Hence, when performing cross-method comparisons, we used a *subset of seven* specific clouds from the test dataset. This subset is shown in the supplementary material.

C. **Hybrid system**. The result of A provides an initialization to subsequent use of approach B, similar to [10]. Then, iterations explicitly optimize physics-based inverse scattering [48], as in B.

D. **Quick hybrid system** This is similar to C, but using only 10 iterations shortening the overall run time.

As an example, Figs. 3 and 4 show recovery of one of the clouds in the said *subset of seven* using Model 1 from Table 2. We report numerical comparisons to the ground truth $\boldsymbol{\beta}^{\text{true}}$. To be able to compare to prior art [45, 46, 47], we use these respective criteria for relative average error and relative total mass error.

$$\epsilon = \frac{||\boldsymbol{\beta}^{\text{true}} - \hat{\boldsymbol{\beta}}||_1}{||\boldsymbol{\beta}^{\text{true}}||_1}, \quad \delta = \frac{||\boldsymbol{\beta}^{\text{true}}||_1 - ||\hat{\boldsymbol{\beta}}||_1}{||\boldsymbol{\beta}^{\text{true}}||_1}. \tag{6}$$

Fig. 5 summarizes the run-time and quality results relating to the *subset of seven* clouds using Model 1. Clearly, in these test cases, 3DeepCT has a huge advantage in terms of runtime: it is about *five orders of magnitude faster* than the state of the art, which is a physics-based explicit approach.

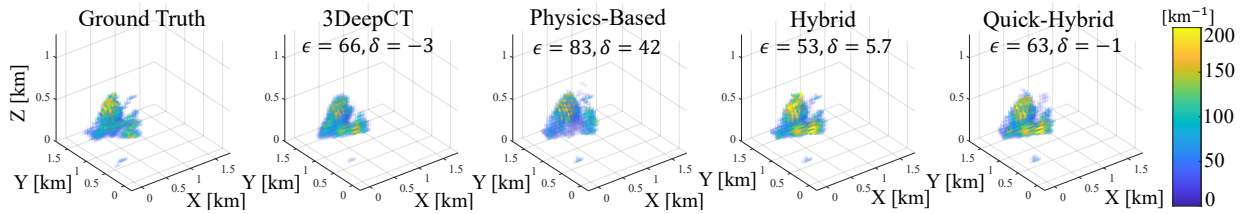


Figure 3. 3D reconstructions of cloud extinction. These recovery results correspond to an example cloud out of the *subset of seven* clouds tested. From left to right: 3D ground truth extinction of the cloud; 3D reconstructed extinction using four methods mentioned in Sec. 5: Model 1 of 3DeepCT, Physics-Based Inverse Scattering, Hybrid system, Quick Hybrid system.

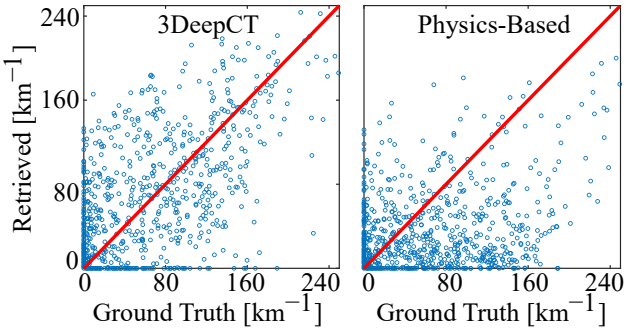


Figure 4. Scatter plots of recovery results of an example cloud out of the *subset of seven* clouds tested. The plot related to methods A and B described in Sec. 5. The reconstructed 3D extinction is $\hat{\boldsymbol{\beta}}$. The red line represents ideal reconstruction, where $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_{\text{true}}$. Voxels deep in the cloud are difficult to reconstruct, therefore we see the scatter around the red line.

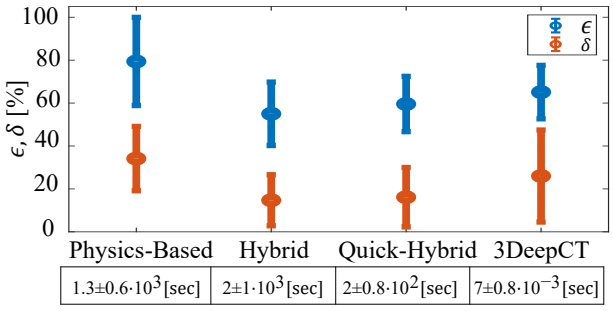


Figure 5. Numerical results of the *subset of seven* clouds, discussed in Sec. 5. Blue is ϵ and orange is δ (see Eq. 6). The circles are the mean values and the lines express $\pm$ standard deviation (std) of the results. The bottom table contains the run time (mean$\pm$std).

In terms of quality, 3DeepCT also yields, on average, better results than the state of the art.

The reason for these findings may be that physics-based inverse rendering solves a non-convex problem, while being oblivious to the nature of clouds (typical spatial structure statistics). Moreover, inverse rendering may converge to a local minimum, as suggested in [10, 25, 86]. On the other hand, 3DeepCT trains on clouds, thus implicitly learns not only RT, but also 3D cloud statistics. This may be a reason that 3DeepCT can provide a quick solution which is cloud-like and complies approximately with RT. Thus, the 3DeepCT output is valuable for initialization. In the supplementary material compares this initialization to other initialization schemes that are used in prior art.

Model#	ϵ	δ	Time [millisec]
1	82±10%	32±16%	7 ± 0.9
2	86±10%	44±16%	7 ± 0.7
3	96±18%	3±50%	18 ± 2

Table 3. Summary of 3DeepCT test results: Mean $\pm$ standard deviation (std) of Model 1 and Model 2 described in Table 2. Definitions of ϵ and δ are in Eq. (6).

The Hybrid method yields a significant improvement of quality, relative to the state of the art, for the same run-time. The Quick-Hybrid method is a compromise of run-time and quality between 3DeepCT and a Hybrid long-run.

When not comparing to slow physics-based optimization, 3DeepCT can be assessed on the full test dataset. Results of Model 1 and Model 2 are summarised in Table 3. There is a degradation in quality when the number of viewpoints decreases. However, a **10 Viewpoints** geometry is more realistic in the short term, as planned by the CloudCT space mission [70].

Model 3 trained on the CASS dataset, where cloud-fields are four times larger than in the BOMEX dataset. Two examples of results of large cloud fields are shown in Fig. 6. Contrary to scalability of CNN, we could not run the physics-based explicit optimization (state of the art) on fields larger than about a kilometer, such as these. This is because physics-based explicit optimization requires excessive resources; this problem is also reported in [52].

6. AirMSPI Real World Data

NASA's ER-2 aircraft flying at 20 km altitude imaged a cloudy ocean scene at 28N 123W, at 2PM local time on February 3, 2013. Imaging was performed by an AirMSPI pushbroom imaging instrument [17] mounted on the aircraft. We use 10 views at the 660nm channel, corresponding to the following on-track angles $38°, \pm 30°, \pm 21° \pm 11°, 0°$ off nadir. Currently there are no real simultaneous multi-

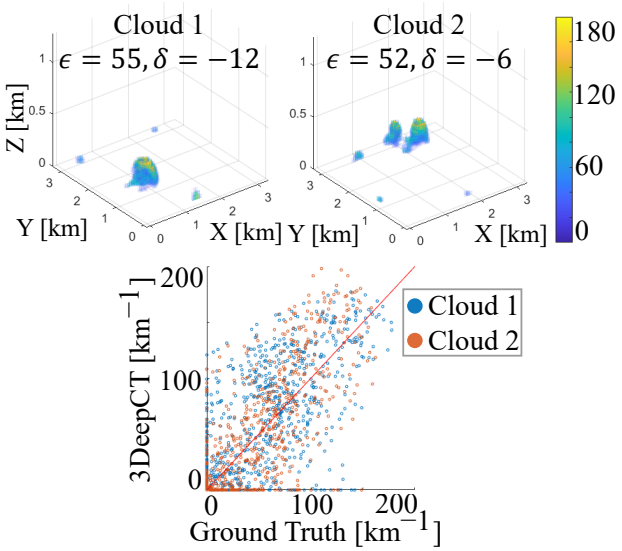


Figure 6. 3D extinction in a large cloud field, 3.2km long, using Model 3 of 3DeepCT. [Left] Recovery of one scene. [Middle] Recovery of another scene. [Right] Scatter plots of recovered values vs. ground-truth, for both scenes.

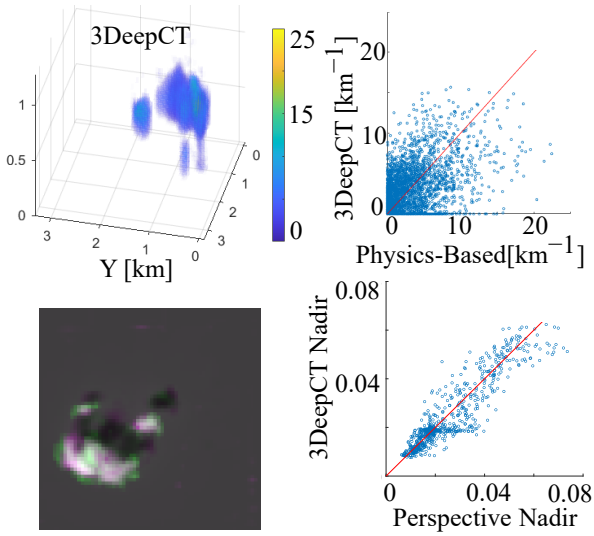


Figure 7. AirMSPI cloud recovery. First row: 3D recovery by Model 4 of 3DeepCT; scatter-plot comparison to physics-based inverse scattering. Second row: Cross validation of the nadir view. The physics-based inverse scattering left-out image is displayed in the green channel; the 3DeepCT result is displayed in magenta; scatter-plot of the cross-validation pixel values of radiance

view satellites images of clouds.[1] For consistency with the perspective of 3DeepCT we performed the following steps: (a) We took clouds appearing in AirMSPI pushbroom data. (b) Applied physics-based inverse rendering. (c) Rendered them perspectively, means the input to 3DeepCT originates from image-based rendering. Fig. 7 shows recovery using Model 4 of 3DeepCT. It also shows cross-validation: the nadir view is left-out of the NN input, yet rendered based on the estimated 3D β. This is compared to the left-out nadir view. Additional images in the supplementary material.

7. Summary

3DeepCT, a DNN for CT performs 3D reconstruction of large translucent domains, focusing on cloud fields. There is a significant advantage for this approach. In particular, it achieves improved accuracy and dramatic reduction in runtime compared to physics-based analysis. Thus, 3DeepCT should enable reconstruction of large cloud fields.

Appearance fundamentally depends on the unknown droplet size distribution (DSD), which sets the phase function, cloud albedo and extinction coefficient $\boldsymbol{\beta}$. Here we estimate $\boldsymbol{\beta}$, which is a product of the DSD. We expect that future research will generalize 3DeepCT to derive the DSD parameters, motivated by [46]. Appearance depends also on parameters known per location on Earth: solar angle (set by local time) and surface (ground/ocean) albedo, derived from prior satellites images. A NN can pre-train per Earth location and time. This enables each NN to focus on the mentioned major unknowns. Possibly, a single NN architecture may be found to be insensitive to the ground albedo and solar angle, making training more efficient. Inclusion of descriptors as in [36] can be the way.

Enriching the training using more diverse data is expected to continuously improve the system. Moreover, we believe significant benefits can stem from transfer-learning. This will allow previously-trained models to be reconfigured to new solar angles, viewpoint directions and imaging systems. Additional scientific domains can benefit from our approach. This can help solving complex tasks where physics-based datasets exist and the computational complexity of state-of-the-art reconstruction is high.

Acknowledgements: We thank Ilan Koren, Orit Altaratz and Roi Ronen for useful discussions and good advice. We thank Aviad Levis and Jesse Loveridge for the pySHDOM code and for being responsive to questions about it. We thank Johanan Erez, Ina Talmon and Daniel Yagodin for technical support. We thank the anonymous reviewers for their useful comments. Yoav Schechner is the Mark and Diane Seiden Chair in Science at the Technion. He is a Landau Fellow - supported by the Taub Foundation. His work was conducted in the Ollendorff Minerva Center. Minvera is funded through the BMBF. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (CloudCT, grant agreement No. 810370).

[1] Data in [45, 46, 47] is push-broom. Data in [1, 32] has a hemispheric field of view, and the images are taken inside the scattering domain, rather than from space or high altitude, to which 3DeepCT is trained.

References

[1] Amit Aides, Aviad Levis, Vadim Holodovsky, Yoav Y Schechner, Dietrich Althausen, and Adi Vainiger. Distributed sky imaging radiometry and tomography. In *ICCP*, pages 1–12, 2020. 2, 3, 8

[2] Derya Akkaynak, Tali Treibitz, Tom Shlesinger, Yossi Loya, Raz Tamir, and David Iluz. What is the space of attenuation coefficients in underwater computer vision? In *Proceedings of CVPR*, pages 4931–4940, 2017. 2

[3] José Baruchel, Jean-Yves Buffiere, and Eric Maire. *X-ray Tomography in Material Science*. Hermes Science Publications, Paris (France), 2000. 1

[4] Sandrine Bony, Bjorn Stevens, Dargan MW Frierson, Christian Jakob, Masa Kageyama, Robert Pincus, Theodore G Shepherd, Steven C Sherwood, A Pier Siebesma, Adam H Sobel, et al. Clouds, circulation and climate sensitivity. *Nature Geoscience*, 8(4):261–268, 2015. 2

[5] David Boublil, Michael Elad, Joseph Shtok, and Michael Zibulevsky. Spatially-adaptive reconstruction in computed tomography using neural networks. *IEEE Transactions on Medical Imaging*, 34(7):1474–1485, 2015. 1

[6] Olivier Boucher, David Randall, Paulo Artaxo, Christopher Bretherton, Gragam Feingold, Piers Forster, V-M Kerminen, Y Kondo, H Liao, U Lohmann, et al. Clouds and aerosols. climate change 2013: The physical science basis. contribution of working group I to the fifth assessment report of the intergovernmental panel on climate change. *Cambridge University Press*, pages 571–657, 2013. 2

[7] Yan-Pei Cao, Zheng-Ning Liu, Zheng-Fei Kuang, Leif Kobbelt, and Shi-Min Hu. Learning to reconstruct high-quality 3D shapes with cascaded fully convolutional networks. In *Proceedings of ECCV*, pages 616–633, 2018. 1

[8] Paulo Ceppi, Florent Brient, Mark D Zelinka, and Dennis L Hartmann. Cloud feedback mechanisms and their representation in global climate models. *Wiley Interdisciplinary Reviews: Climate Change*, 8(4):e465, 2017. 2

[9] Subrahmanyan Chandrasekhar. *Radiative Transfer*. Oxford University Press, Oxford (UK), 1950. 2

[10] Chengqian Che, Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. Towards learning-based inverse subsurface scattering. In *ICCP*, pages 1–12, 2020. 2, 4, 6, 7

[11] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of CVPR*, pages 5939–5948, 2019. 1

[12] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3D-R2N2: A unified approach for single and multi-view 3D object reconstruction. In *ECCV*, pages 628–644. Springer, 2016. 1

[13] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 424–432. Springer, 2016. 1

[14] CMV4000 sensor. https://ams.com/cmv4000. 3, 6

[15] C Cornet, L C-Labonnote, and Frédéric Szczap. Three-dimensional polarized monte carlo atmospheric radiative transfer model (3DMCPOL): 3D effects on polarized visible reflectances of a cirrus cloud. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 111(1):174–186, 2010. 3

[16] Cory Davis, Claudia Emde, and Robert Harwood. A 3-d polarized reversed monte carlo radiative transfer model for millimeter and submillimeter passive remote sensing in cloudy atmospheres. *IEEE Transactions on Geoscience and Remote Sensing*, 43(5):1096–1101, 2005. 2, 3

[17] David J. Diner, Feng Xu, Michael J. Garay, John V. Martonchik, Brian E. Rheingans, Sven Geier, Ab Davis, B. R. Hancock, V. M. Jovanovic, and M. A. Bull. The airborne multiangle spectropolarimetric imager (airmspi): a new tool for aerosol and cloud remote sensing. *Atmospheric Measurement Techniques*, 6(8):2007, 2013. 7

[18] Adrian Doicu, Dmitry Efremenko, and Thomas Trautmann. A multi-dimensional vector spherical harmonics discrete ordinate method for atmospheric radiative transfer. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 118:121–131, 2013. 3

[19] Qi Dou, Lequan Yu, Hao Chen, Yueming Jin, Xin Yang, Jing Qin, and Pheng-Ann Heng. 3D deeply supervised network for automated segmentation of volumetric medical images. *Medical Image Analysis*, 41:40–54, 2017. 1

[20] K Franklin Evans. The spherical harmonics discrete ordinate method for three-dimensional atmospheric radiative transfer. *Journal of the Atmospheric Sciences*, 55(3):429–446, 1998. 2, 3

[21] Jiwen Fan, Mikhail Ovtchinnikov, Jennifer M Comstock, Sally A McFarlane, and Alexander Khain. Ice formation in arctic mixed-phase clouds: Insights from a 3-d cloud-resolving model with size-resolved aerosol and cloud microphysics. *Journal of Geophysical Research: Atmospheres*, 114(D4), 2009. 5

[22] Linda Forster, Anthony B. Davis, David J. Diner, and Bernhard Mayer. Toward cloud tomography from space using MISR and MODIS: Locating the "Veiled Core" in opaque convective clouds. *Journal of the Atmospheric Sciences*, 78(1):155–166, 2021. 2

[23] Adam Geva, Yoav Y Schechner, Yonatan Chernyak, and Rajiv Gupta. X-ray computed tomography through scatter. In *Proceedings of ECCV*, pages 34–50, 2018. 1, 3

[24] Rohit Girdhar, David F Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. In *ECCV*, pages 484–499. Springer, 2016. 1

[25] Ioannis Gkioulekas, Anat Levin, and Todd Zickler. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *ECCV*, pages 685–701. Springer, 2016. 2, 3, 7

[26] Ioannis Gkioulekas, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin. Inverse volume rendering with material dictionaries. *TOG*, 32(6):1–13, 2013. 2, 3

[27] Roy G Grainger, Jonathan Lucas, Gareth E Thomas, and Graham BL Ewen. Calculation of Mie derivatives. *Applied Optics*, 43(28):5386–5393, 2004. 5

[28] Edward Grant, Pushmeet Kohli, and Marcel van Gerven. Deep disentangled representations for volumetric reconstruction. In *ECCV*, pages 266–279. Springer, 2016. 1

[29] Hengtao Guo, Sheng Xu, Bradford Wood, and Pingkun Yan. Sensorless freehand 3D ultrasound reconstruction via deep contextual learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 463–472. Springer, 2020. 1

[30] Thijs Heus, Harm JJ Jonker, Harry EA Van den Akker, Eric J Griffith, Michal Koutek, and Frits H Post. A statistical approach to the life cycle analysis of cumulus clouds selected in a virtual reality environment. *Journal of Geophysical Research: Atmospheres*, 114(D6), 2009. 5

[31] Roman Hochuli, Samuel Powell, Simon Arridge, and Ben Cox. Quantitative photoacoustic tomography using forward and adjoint Monte Carlo models of radiance. *Journal of Biomedical Optics*, 21(12):126004, 2016. 2

[32] Vadim Holodovsky, Yoav Y Schechner, Anat Levin, Aviad Levis, and Amit Aides. In-situ multi-view multi-scattering stochastic tomography. In *ICCP*, pages 1–12, 2016. 2, 3, 8

[33] Jehoiada Jackson, She Kun, Kwame Obour Agyekum, Ariyo Oluwasanmi, and Parinya Suwansrikham. A fast single-image dehazing algorithm based on dark channel prior and rayleigh scattering. *IEEE Access*, 8:73330–73339, 2020. 2

[34] Kamal Jnawali, Mohammad R Arbabshirani, Navalgund Rao, and Alpen A Patel. Deep 3D convolution neural network for ct brain hemorrhage classification. In *Medical Imaging: Computer-Aided Diagnosis*, volume 10575, page 105751C. International Society for Optics and Photonics, 2018. 1

[35] Adrian Johnston, Ravi Garg, Gustavo Carneiro, Ian Reid, and Anton van den Hengel. Scaling cnns for high resolution volumetric reconstruction from a single image. In *Proceedings of ICCV Workshops*, pages 939–948, 2017. 1

[36] Simon Kallweit, Thomas Müller, Brian Mcwilliams, Markus Gross, and Jan Novák. Deep scattering: Rendering atmospheric clouds with radiance-predicting neural networks. *TOG*, 36(6):1–11, 2017. 8

[37] Ryunosuke Kazahaya, Toshiya Mori, Kohei Kazahaya, and Jun-ichi Hirabayashi. Computed tomography reconstruction of so2 concentration distribution in the volcanic plume of miyakejima, japan, by airborne traverse technique using three uv spectrometers. *Geophysical Research Letters*, 35(13), 2008. 1

[38] A Khain, A Pokrovsky, M Pinsky, A Seifert, and Vaughan Phillips. Simulation of effects of atmospheric aerosols on deep turbulent convective clouds using a spectral microphysics mixed-phase cumulus cloud model. part i: Model description and possible applications. *Journal of the Atmospheric Sciences*, 61(24):2963–2982, 2004. 5

[39] Marat F Khairoutdinov and David A Randall. Cloud resolving modeling of the arm summer 1997 iop: Model formulation, results, uncertainties, and sensitivities. *Journal of the Atmospheric Sciences*, 60(4):607–625, 2003. 5

[40] Pramook Khungurn, Daniel Schroeder, Shuang Zhao, Kavita Bala, and Steve Marschner. Matching real fabrics with micro-appearance models. *TOG*, 35(1):1–1, 2015. 2

[41] Alexander Kleinschrodt, Nikolai Reed, and Klaus Schilling. A comparison of scheduling algorithms for low cost ground station networks. In *67st International Astronautical Congress. Guadalajara, Mexico*, pages 1–15, 2016. 6

[42] Alankar Kowtal, Avilash Cramer, Dufan Wu, Kai Yang, Wolfgang Krull, Ioannis Gkioulekas, and Rajiv Gupta. Signal sensing and reconstruction paradigms for a novel multi-source static computed tomography system. In *ICASSP*, pages 9274–9278, 2020. 3

[43] Louis Kratz and Ko Nishino. Factorizing scene albedo and depth from a single foggy image. In *ICCV*, pages 1701–1708. IEEE, 2009. 2

[44] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38(3):199–218, 2000. 5

[45] Aviad Levis, Yoav Y Schechner, Amit Aides, and Anthony B Davis. Airborne three-dimensional cloud tomography. In *Proceedings of ICCV*, pages 3379–3387, 2015. 2, 3, 6, 8

[46] Aviad Levis, Yoav Y Schechner, and Anthony B Davis. Multiple-scattering microphysics tomography. In *Proceedings of CVPR*, pages 6740–6749, 2017. 2, 5, 6, 8

[47] Aviad Levis, Yoav Y Schechner, Anthony B Davis, and Jesse Loveridge. Multi-view polarimetric scattering cloud tomography and retrieval of droplet size. *Remote Sensing. 12. 2831. 10.3390/rs12172831*, 2020. 2, 6, 8

[48] Aides A. Levis A. pySHDOM. https://github.com/aviadlevis/pyshdom, 2019. 6

[49] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. *TOG*, 37(6):1–11, 2018. 4

[50] Qian Liu, Hui Xu, Dexuan Sha, Tsengdar Lee, Daniel Q Duffy, Jeff Walter, and Chaowei Yang. Hyperspectral infrared sounder cloud detection using deep neural network model. *IEEE Geoscience and Remote Sensing Letters*, 2020. 2

[51] Xiaohong Liu, Yongrui Ma, Zhihao Shi, and Jun Chen. Griddehazenet: Attention-based multi-scale network for image dehazing. In *Proceedings of ICCV*, pages 7314–7323, 2019. 2

[52] Tamar Loeub, Aviad Levis, Vadim Holodovsky, and Yoav Y Schechner. Monotonicity prior for cloud tomography. In *Proceedings of ECCV*, pages 24–29, 2020. 3, 7

[53] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: learning dynamic renderable volumes from images. *TOG*, 38(4):1–14, 2019. 1, 4

[54] Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. Langevin monte carlo rendering with gradient-based adaptation. *TOG*, 2020. 3

[55] J Vanderlei Martins, Roberto Fernandez-Borda, Brent McBride, Lorraine Remer, and Henrique MJ Barbosa. The Harp Hype Ran Gular imaging polarimeter and the need for small satellite payloads with high science payoff for Earth science remote sensing. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 6304–6307, 2018. 2

[56] Bernhard Mayer. Radiative transfer in the cloudy atmosphere. In *EPJ Web of Conferences*, volume 1, pages 75–99. EDP Sciences, 2009. 3

[57] Felipe A Mejia, Ben Kurtz, Aviad Levis, Íñigo de la Parra, and Jan Kleissl. Cloud tomography applied to sky images: A virtual testbed. *Solar Energy*, 176:287–300, 2018. 1

[58] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *Proceedings of CVPR*, pages 4460–4470, 2019. 1

[59] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of ECCV*, 2020. 1

[60] Teruyuki Nakajima and Michael D. King. Determination of the optical thickness and effective particle radius of clouds from reflected solar radiation measurements. part i: Theory. *Journal of the Atmospheric Sciences*, 47(15):1878–1893, 1990. 2

[61] Teruyuki Nakajima, Michael D. King, James D. Spinhirne, and Lawrence F Radke. Determination of the optical thickness and effective particle radius of clouds from reflected solar radiation measurements. part ii: Marine stratocumulus observations. *Journal of the Atmospheric Sciences*, 48(5):728–751, 1991. 2

[62] R. A. J. Neggers, H. J. J. Jonker, and A. P. Siebesma. Size statistics of cumulus cloud populations in large-eddy simulations. *Journal of the Atmospheric Sciences*, 60(8):1060–1074, 2003. 5

[63] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NIPS*, pages 8026–8037, 2019. 5

[64] Robert Pincus and K Franklin Evans. Computational cost and accuracy in calculating three-dimensional radiative transfer: Results for new implementations of Monte Carlo and SHDOM. *Journal of the Atmospheric Sciences*, 66(10):3131–3146, 2009. 2

[65] Bingjiang Qiu, Jiapan Guo, Joep Kraeima, Haye H Glas, Ronald JH Borra, Max JH Witjes, and Peter MA van Ooijen. Automatic segmentation of the mandible from computed tomography scans for 3D virtual surgical planning using the convolutional neural network. *Physics in Medicine & Biology*, 64(17):175020, 2019. 1

[66] David Ren, Colin Ophus, Michael Chen, and Laura Waller. A multiple scattering algorithm for three dimensional phase contrast atomic electron tomography. *Ultramicroscopy*, 208:112860, 2020. 1

[67] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3D representations at high resolutions. In *Proceedings of CVPR*, pages 3577–3586, 2017. 1

[68] Roi Ronen, Yoav Y Schechner, and Eshkol Eytan. 4D cloud scattering tomography. In *Proceedings of ICCV*, 2021. 2

[69] Guy Satat, Barmak Heshmat, Dan Raviv, and Ramesh Raskar. All photons imaging through volumetric scattering. *Scientific Reports*, 6(1):1–8, 2016. 2

[70] Klaus Schilling, Yoav Y. Schechner, and Ilan Koren. CloudCT - computed tomography of clouds by a small satellite formation. In *Proceedings of the 12th IAA symposium on Small Satellites for Earth Observation*, 2019. 6, 7

[71] Mark Sheinin and Yoav Y Schechner. The next best underwater view. In *Proceedings of CVPR*, pages 3764–3773, 2016. 2

[72] Liyue Shen, Wei Zhao, and Lei Xing. Patient-specific reconstruction of volumetric computed tomography images from a single projection view via deep learning. *Nature Biomedical Engineering*, 3(11):880–888, 2019. 1

[73] A Pier Siebesma, Christopher S Bretherton, Andrew Brown, Andreas Chlond, Joan Cuxart, Peter G Duynkerke, Hongli Jiang, Marat Khairoutdinov, David Lewellen, Chin-Hoh Moeng, et al. A large eddy simulation intercomparison study of shallow cumulus convection. *Journal of the Atmospheric Sciences*, 60(10):1201–1219, 2003. 5

[74] Alok Kumar Singh, Dinesh N Naik, Giancarlo Pedrini, Mitsuo Takeda, and Wolfgang Osten. Exploiting scattering media for exploring 3D objects. *Light: Science & Applications*, 6(2):e16219–e16219, 2017. 2

[75] Dmitry Veikherman, Amit Aides, Yoav Y Schechner, and Aviad Levis. Clouds in the cloud. In *Asian Conference on Computer Vision*, pages 659–674. Springer, 2014. 4

[76] WorldView. Worldview Earthdata NASA. https://worldview.earthdata.nasa.gov/. 1

[77] Dufan Wu, Kyungsang Kim, and Quanzheng Li. Computationally efficient deep neural network for computed tomography image reconstruction. *Medical physics*, 46(11):4763–4776, 2019. 1

[78] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *NIPS*, pages 82–90, 2016. 1

[79] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. Pix2vox: Context-aware 3D reconstruction from single and multi-view images. In *Proceedings of ICCV*, pages 2690–2698, 2019. 1

[80] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. In *NIPS*, pages 492–502, 2019. 1

[81] Huiwen Xue and Graham Feingold. Large-eddy simulations of trade wind cumuli: Investigation of aerosol indirect effects. *Journal of the Atmospheric Sciences*, 63(6):1605–1622, 2006. 5

[82] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision. In *NIPS*, pages 1696–1704, 2016. 1

[83] Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. A differential theory of radiative transfer. *TOG*, 38(6):1–16, 2019. 3

[84] Jinglin Zhang, Pu Liu, Feng Zhang, and Qianqian Song. Cloudnet: Ground-based cloud classification with deep convolutional neural network. *Geophysical Research Letters*, 45(16):8665–8672, 2018. 2

[85] Yunyan Zhang, Stephen A Klein, Jiwen Fan, Arunchandra S Chandra, Pavlos Kollias, Shaocheng Xie, and Shuaiqi Tang. Large-eddy simulation of shallow cumulus over land: A composite case based on ARM long-term observations at its Southern Great Plains site. *Journal of the Atmospheric Sciences*, 74(10):3229–3251, 2017. 5

[86] Shuang Zhao, Ravi Ramamoorthi, and Kavita Bala. High-order similarity relations in radiative transfer. *TOG*, 33(4):1–12, 2014. 7

3DeepCT: Learning Volumetric Scattering Tomography of Clouds, Supplementary Material

Yael Sde-Chen, Yoav Y. Schechner, Vadim Holodovsky
Viterbi Faculty of Electrical and Computer Engineering,
Technion - Israel Institute of Technology, Haifa, Israel
yael.sde.chen@gmail.com, yoav@ee.technion.ac.il, vholod@ef.technion.ac.il

Eshkol Eytan
Department of Earth and Planetary Science,
The Weizmann Institute of Science, Rehovot, Israel
eshkol.eytan@weizmann.ac.il

Abstract

This is a supplementary document to the main manuscript. Here we provide additional numerical results.

1. Introduction

This supplementary material contains eight sections. Section 2 describes the BOMEX and CASS setups for the cloud fields simulations. Section 3 describes the noise model we use in our simulated database. Section 4 presents a comparison between 3DeepCT and other initialization schemes. Section 5 describes the **10 Viewpoints** geometry. Section 6 presents some of the ablation studies. Section 8 provides additional data and results of the *subset of seven* clouds, which were not included in the main manuscript due to space limits. Section 7 AirMSPI real data cross validation results.

2. Datasets

The cloud field was simulated using the BOMEX [5] and CASS setups [6]. The BOMEX case study is based on surface fluxes, large-scale tendencies, and profiles of wind, humidity, and temperature in trade wind cumulus cloud fields over the Atlantic ocean near Barbados.

The CASS simulation of terrestrial clouds is based on a composite case called the Continental Active Surface-Forced Shallow Cumulus (CASS) [6]. The setup was constructed from measurements of the ARM project (Atmospheric Radiation Measurements), during 1997-2009 in the Southern Great Plains of the USA. A composite of the diurnal cycle of surface fluxes, large-scale forcing, and profiles of wind, humidity, and temperature was established over all days of shallow convection with non-precipitating cumulus clouds and was used to initiate the LES model.

The simulated domain for both setups is $12.82\ \mathrm{km} \times 12.82\ \mathrm{km}$ wide, with cyclic horizontal boundary conditions. The time duration of the BOMEX simulation is 8 hours (including 2 hours of spin-up time), while CASS simulates 12 hours with varying conditions according to the diurnal cycle. From each simulation, we use a snapshot every 2 minutes, to produce the database of cloud fields. The simulation evolved in 1 second increments, each yielding a different 3D spatial field which includes dozens of clouds. The data created takes 1.2 TB of memory. It took ≈ 5 days to generate, on an Intel Xeon Gold 5115 with 256 cores.

3. Noise Model

Section 2.1 in the main manuscript explains how a forward model derives a theoretic radiance field $I(\boldsymbol{x}, \boldsymbol{\omega})$. However, real-world radiance is in the form of a random photon flux, which obeys a Poissonian distribution. The photons are converted to discrete electric charges at the sensor. Furthermore, the sensor introduces noise due to various causes, according to its specifications. Let i^{e} be the expected photo-electron count of a pixel. At darkness and infinitesimal exposure time, readout noise has standard deviation ρ_{read}, in $\mathrm{electrons}$. At temperature $\mathcal{T}$, the sensor dark current in $\mathrm{electrons/sec}$ is $D_{\mathcal{T}}$. The exposure time is Δt. The standard deviation of quantization noise in $\mathrm{electrons}$ is $\rho_{\mathrm{digit}} = g^{\mathrm{e}}/\sqrt{12}$, where g^{e} is the number of photo-electrons required to change a unit gray level [4]. Overall, in a pixel readout, in units of $\mathrm{electrons}$, the noise has variance of approximately

$$V = i^{\mathrm{e}} + D_{\mathcal{T}} \Delta t + \rho^2_{\mathrm{read}} + \rho^2_{\mathrm{digit}} \, . \tag{1}$$

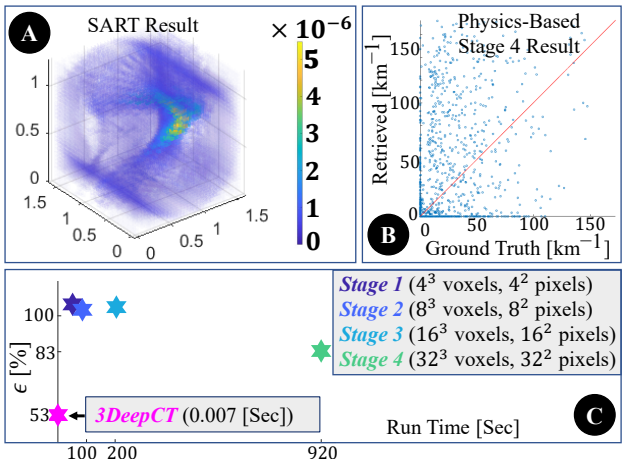


Figure 1. Recovering the cloud of Figs.3 of the main paper. (a) SART yields negligible initialization. (b) Scatter plots following coarse-to-fine analysis. (c) The error ϵ and run-time clock after each coarse-to-fine stage.

Our simulations of a perspective camera have a noise model based on the CMV4000 sensor [1]. The pixel size is $5.5 \times 5.5\text{micron}^2$, $\rho_{\text{read}} = 13$ electrons, $D_{\mathcal{T}} = 125$ electrons/sec at 25°C, full well of a pixel is 13,500 electrons. It uses 10bit quantization, thus $g^{\text{e}} = 13,500/2^{10}$. The exposure time is adjustable according to the radiance that reached the camera such that the camera sensor reaches the full well.

4. Initialization Schemes

3DeepCT provides good initialization to the physics-based inverse rendering. We compare it with initializations used in prior art. As expected, applying SART yields negligible values, 6-8 orders of magnitude lower than the true $\boldsymbol{\beta}$ (See Fig. 1a above). Initialization by SART for cloud scattering-CT is as initialization by a null cloud. Fig. 1b,c here shows progression from $\times 8$ coarser grids (in each spatial coordinate) to full resolution by a coarse-to-fine [2, 3] process. It does not seem to save much time here. It is far slower and less effective than 3DeepCT.

5. 10 Viewpoints Geometry

The **10 Viewpoints** geometry, presented in Sec. 4.2 of the main manuscript is visualized in Fig. 2 herein. Recall from the main manuscript that this geometry uses 10 satellites which orbit at altitude of 600km. Nearest-neighbor satellites are 100km apart. They view the same field in off-nadir angles -46°, $\pm 34^{\circ}$, $\pm 26^{\circ}$, $\pm 18^{\circ}$, $\pm 9^{\circ}$ and 0°. Each carries a perspective camera. The field of view of each camera is 0.22°, corresponding to a ground footprint at the nadir of 1.6km $\times$ 1.6km, at 50 m ground resolution. Solar azimuth and zenith angles are 45° and 30°, respectively.

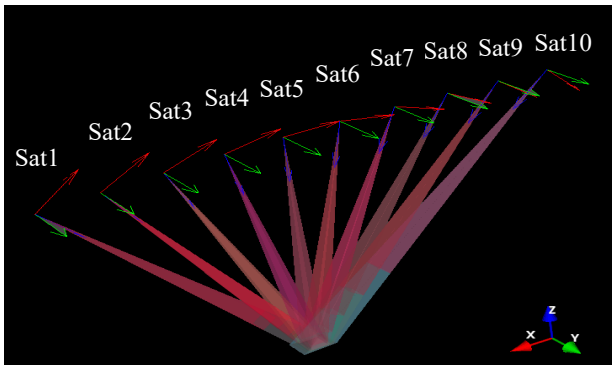


Figure 2. **10 Viewpoints** geometry; orbit at 600km altitude, nearest-neighbor satellites are 100km apart, remotely sensing clouds in the atmosphere from space.

Model	ϵ	δ	Time [millisec]
U-Net	91±12%	38±16%	7 ± 1
ResNet	94.5±12%	43±20%	5 ± 0.7

Table 1. Summary of test results: Mean $\pm$ standard deviation (std) of U-Net and ResNet models, discussed in Sec. 3.3 of the main manuscript. Equations for ϵ and δ are in Eq. (8) of the main manuscript.

6. Ablation Studies

The architecture of 3DeepCT, its components and hyperparameters were carefully examined and tuned by ablation studies. For example, Fig. 3 herein shows how the NN depth and batch normalization affect convergence, and Fig. 4 herein shows the training and evaluation loss of 3DeepCT compared to U-Net and ResNet architectures. Table 1 herein shows the summary results of these two architectures.

Space carving ablation studies were also examined. Without space-carving, we obtain $\epsilon = 85 \pm 14\%$ and $\delta = 9.5 \pm 18\%$ when using the Model 1 described in Table 2 in the main manuscript. Space-carving improved the results, as seen in Fig. 5 herein.

7. AirMSPI Cross Validation Results

Fig. 6 shows the perspective nadir view which was left out of the 3DeepCT input. 3DeepCT recovered the volumetric cloud without using the nadir view of the cloud. We rendered the recovered result in the nadir direction and perform cross validation between this rendered view and the original left-out image.

8. Simulated Inference Results

Fig. 3 in the main manuscript shows the 3D reconstructions of cloud extinction. Fig. 4 of the main manuscript shows the scatter plots of recovery results related to methods A and B, described in Sec. 5 of the main manuscript.

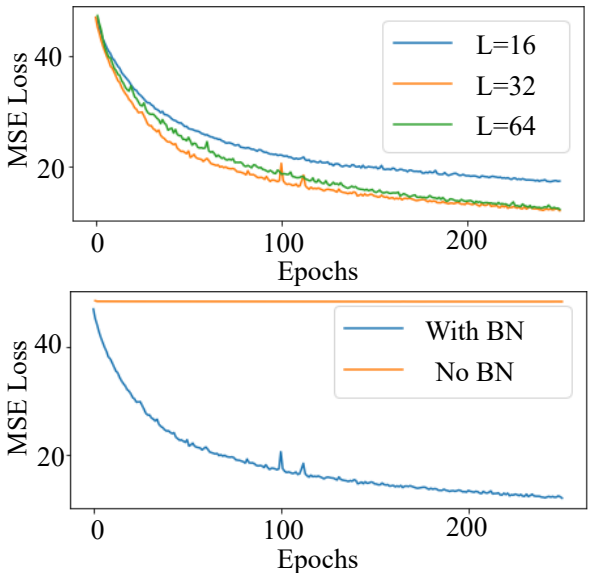


Figure 3. Hyper-parameters and ablation experiment: [First row] Tuning experiment of hyper-parameter L ; [Second row] Ablation study of the batch normalization (BN) module of the NN.

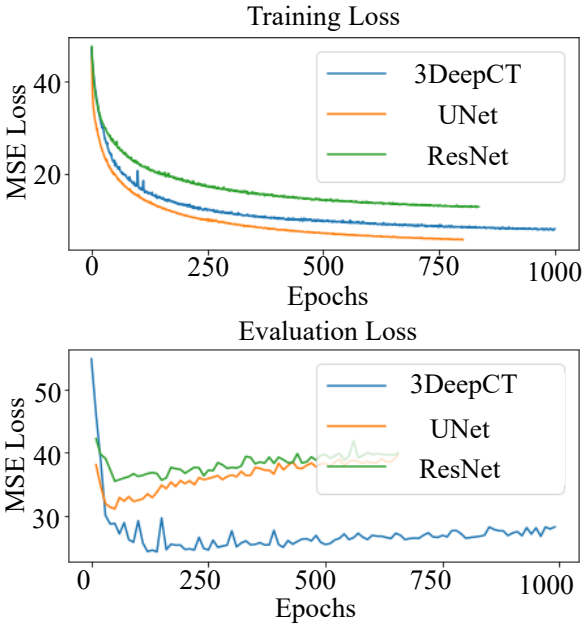


Figure 4. Training and evaluation loss of 3DeepCT, compared to those of U-Net and ResNet architectures.

Fig. 7 herein relates to the additional methods C and D. Fig. 8 herein shows the *subset of seven* clouds. In the main manuscript, inference results of our Model 1 are demonstrated on one cloud out of the *subset of seven* clouds. This cloud is shown in Fig. 8(a) herein. Three additional clouds are shown in Figures 8(b)-(d) herein. Figures 9(a)-(c) herein show their respective recovery, using the four approaches

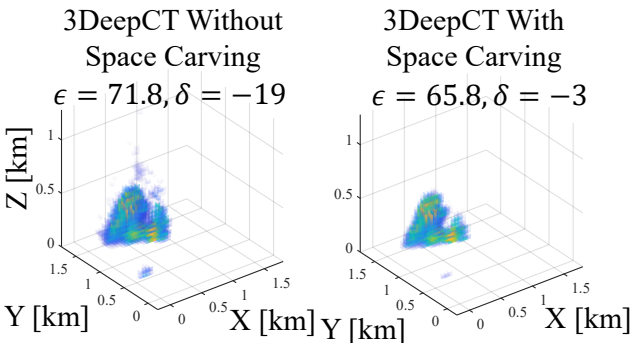


Figure 5. Recovery results: [Left] 3DeepCT raw recovery result without space-carving, [Right] 3DeepCT recovery results using the space-carving mask. The results show the improvement when using space-carving.

described in Sec. 5 of the main manuscript.

References

[1] CMV4000 sensor. https://ams.com/cmv4000. 2

[2] Ioannis Gkioulekas, Anat Levin, and Todd Zickler. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *ECCV*, pages 685–701. Springer, 2016. 2

[3] Tamar Loeub, Aviad Levis, Vadim Holodovsky, and Yoav Y Schechner. Monotonicity prior for cloud tomography. In *Proceedings of ECCV*, pages 24–29, 2020. 2

[4] Yoav Y. Schechner, Shree Nayar, and Peter N. Belhumeur. Multiplexing for optimal lighting. *PAMI*, 29:1339–1354, 2007. 1

[5] A Pier Siebesma, Christopher S Bretherton, Andrew Brown, Andreas Chlond, Joan Cuxart, Peter G Duynkerke, Hongli Jiang, Marat Khairoutdinov, David Lewellen, Chin-Hoh Moeng, et al. A large eddy simulation intercomparison study of shallow cumulus convection. *Journal of the Atmospheric Sciences*, 60(10):1201–1219, 2003. 1

[6] Yunyan Zhang, Stephen A Klein, Jiwen Fan, Arunchandra S Chandra, Pavlos Kollias, Shaocheng Xie, and Shuaiqi Tang. Large-eddy simulation of shallow cumulus over land: A composite case based on ARM long-term observations at its Southern Great Plains site. *Journal of the Atmospheric Sciences*, 74(10):3229–3251, 2017. 1

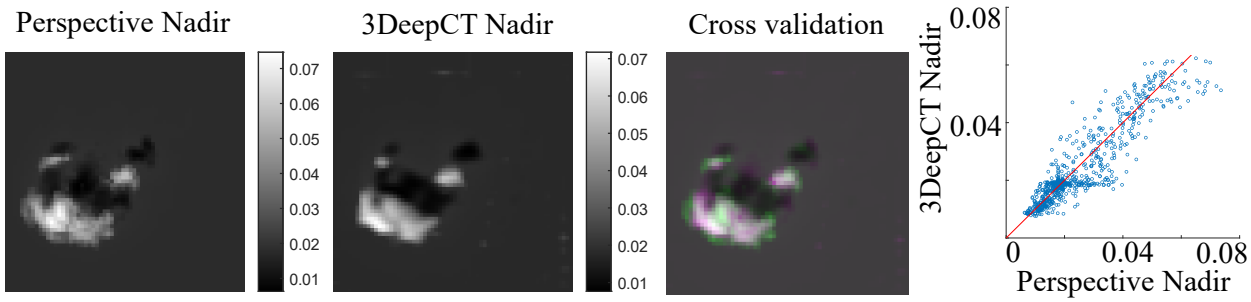


Figure 6. From left to right: Left-out perspective nadir view; Nadir view rendered from the results of 3DeepCT; Cross validation of the nadir view - the physics-based inverse scattering left-out image is displayed in the green channel; the 3DeepCT result is displayed in magenta; scatter-plot of the cross-validation pixel values of radiance.

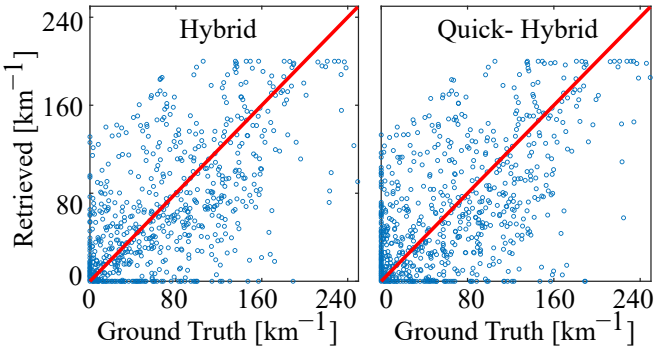


Figure 7. Recovering the cloud of Fig. 3 of the main manuscript.

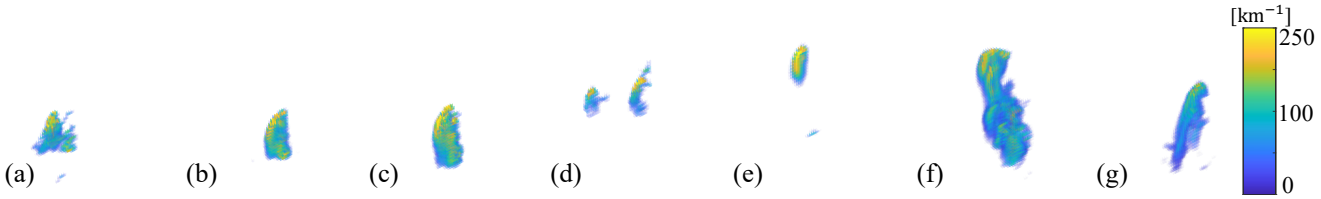
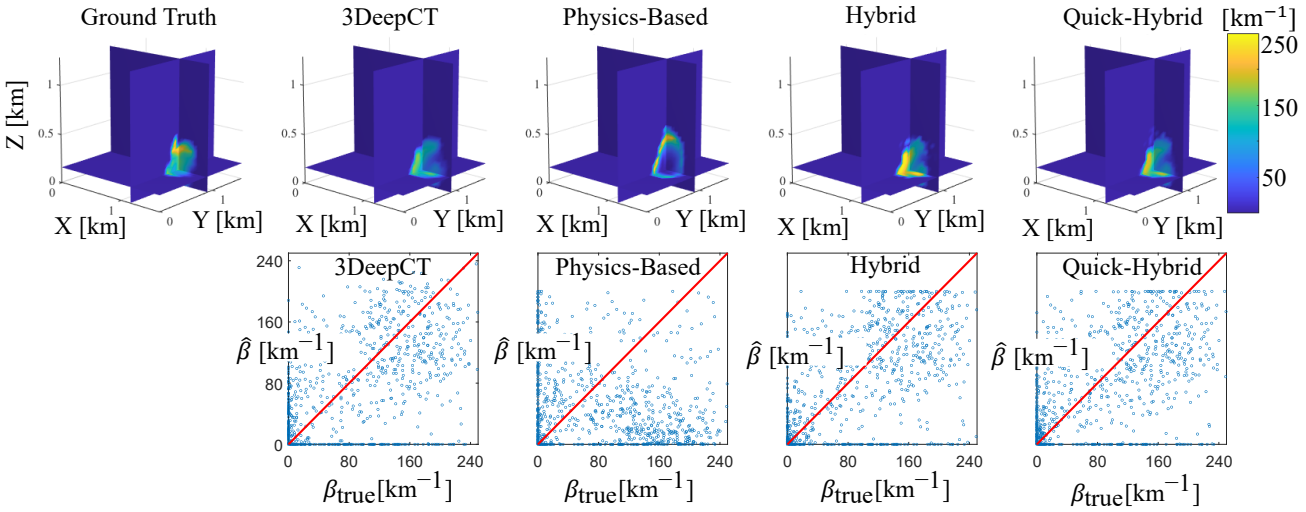
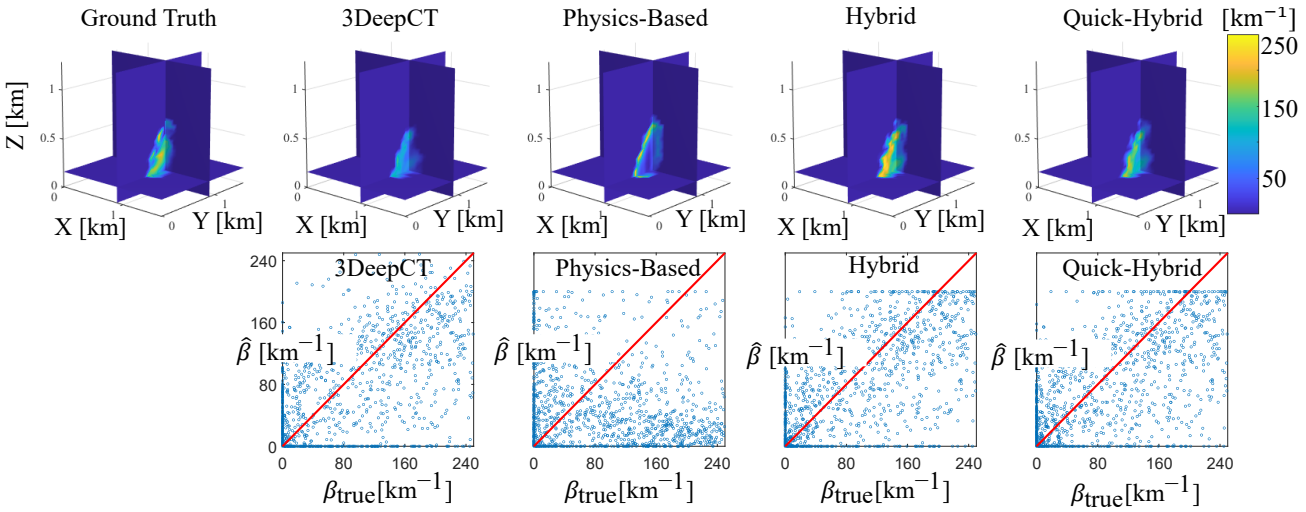


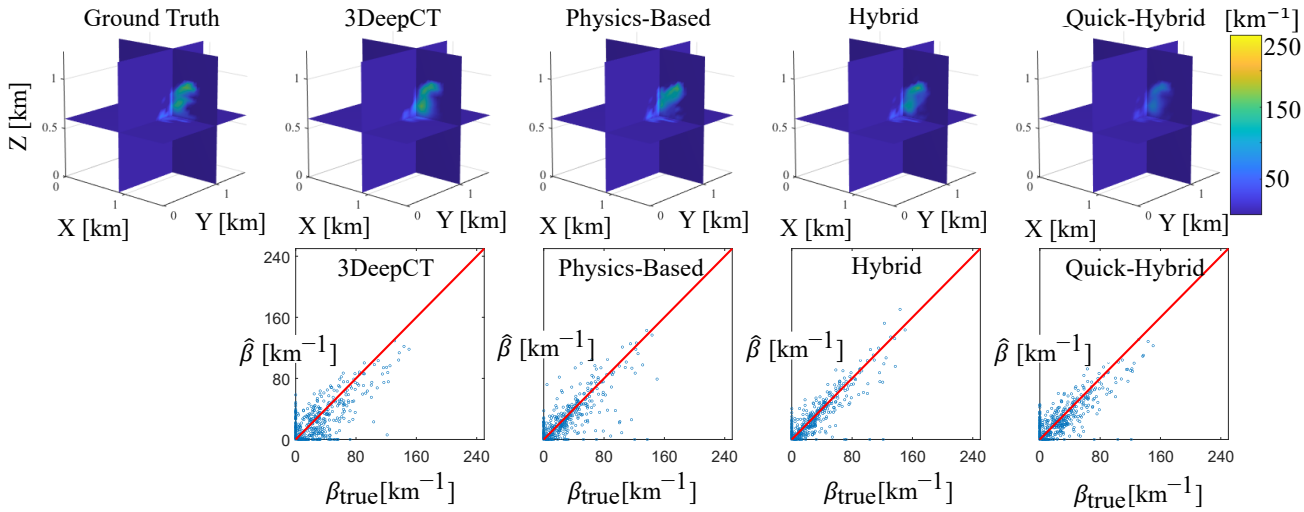
Figure 8. The *Subset of seven* clouds.



(a) 3D extinction recovery of cloud (b) from Fig. 8 herein.



(b) 3D extinction recovery of cloud (c) from Fig. 8 herein.



(c) 3D extinction recovery of cloud (d) from Fig. 8 herein.

Figure 9. [First row] From left to right: 3D ground-truth extinction of the cloud; 3D reconstructed extinction using the four methods mentioned in Sec. 5 of the main manuscript. [Second row] Scatter plots of the recovery results. The plots relate to the four methods at the top row. The reconstructed 3D extinction is $\hat{\boldsymbol{\beta}}$. The red line represents ideal reconstruction, where $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_{\text{true}}$.