Sky-NeRF: Learning 4D Cloud Topography in a Dynamic Neural Radiance Field

Theïlo Terrisse[1]*, Dawa Derksen[2], David Youssefi[2], Hugo Meric[2]

[1] CS Group, 6 rue Brindejonc des Moulinais, Toulouse, France - theilo.terrisse@cs-soprasteria.com
[2] CNES, 18 avenue Edouard Belin, Toulouse, France - (dawa.derksen, david.youssefi, hugo.meric)@cnes.fr

Keywords: Cloud Topography, Trajectory Estimation, 4D Reconstruction, Dynamic Neural Radiance Fields, Physics-Inspired Deep Learning

Abstract

We present Sky-NeRF, a novel method for cloud topography estimation based on Dynamic Neural Radiance Fields. Similar to NeRF, we propose to model the 3D structure of clouds as a radiance field, encoded in the parameters of a neural representation. Our goal is to reconstruct the 3D geometry, appearance, and motion of the cloud using a stereo-video of high-resolution top of the atmosphere radiance images. In this paper, we evaluate a novel way of modeling the dynamic behavior of clouds, with the goal of extracting added-value physical information regarding the cloud such as advection speed and direction, velocity field and cloud trajectories. We investigate how to include a simple physical prior, advection, into the learning system and evaluate its impact. Our results show that Sky-NeRF is able to provide a more complete 4D reconstruction than traditional stereo-matching-based algorithms. Moreover, thanks to a physics-based interpolation, Sky-NeRF is able to generate coherent new images from unseen viewing angles, and at any time between the observed frames.

1. Introduction

The topic of atmospheric observation is of primary importance for many applications, from weather forecasting to climate studies, cloud physics, route planning or fire and flood monitoring. In this context, the study of 3D cloud structures (cloud topography) and their evolution through time (cloud dynamics) are paramount to achieve a deeper understanding of the physical phenomena driving cloud formation and movement. A large amount of data covering a variety of cloud types and altitudes is necessary to design appropriate atmospheric fluid dynamic models (Lac et al., 2018, Strauss et al., 2019).

To this end, the Centre National d'Etudes Spatiales (CNES, French Space Agency) and the Israeli Space Agency (ISA) have designed a constellation of imaging satellites named Cluster for Cloud evolution, Climate and Lightning (C^3IEL). Their purpose is to monitor extreme weather events and understand the complex physical forces behind cloud dynamics (Dandini et al., 2022). A constellation of two Low Earth Orbit satellites is being designed to capture temporal sequences (videos) of the cloud structure, seen from multiple viewing angles (illustrated in Figure 1). A satellite segment is extremely valuable to study the differences in cloud morphology in various places on Earth. These sensors should provide data to study convective cloud development and rare weather events such as cyclones or storms.

After capturing the images, the next step of the process is to produce 3D models. This is known as 3D cloud topography (Dandini et al., 2022) and is usually based on stereo photogrammetry pipelines. Such algorithms provide relatively accurate 3D reconstructions of the cloud structure, with median absolute altitude errors below $150m$ with CARS (Youssefi et al., 2020) according to our experiments. Nonetheless, extracting the temporal behaviour of clouds based on a sequence of image pairs remains a challenging task. First, clouds are dynamic structures, due to advection and convection forces, which cause the cloud structure to change through time in a non-rigid fashion. Second, due to the flyover configuration of the sensors, only certain parts of the cloud are visible at a given time step (as visible in Figure 2). Thirdly, clouds can be semi-transparent and more appropriately described as a continuous density field rather than a point cloud, mesh, or Digital Surface Model (DSM). Finally, the variety of cloud structures and shapes makes it difficult to pre-train a large neural network on a simulated dataset without incurring months of costly fluid dynamics simulations and optical rendering.

In this context, the Neural Radiance Field (NeRF) (Mildenhall et al., 2021) appears as an interesting candidate to perform 3D cloud topography. A NeRF is a 3D neural representation trained from a set of 2D images. Each NeRF is trained on a single scene, and therefore requires no prior training on similar structures. NeRFs encode a 3D scene in a continuous fashion using two fields that represent the optical properties of the scene. First, a volumetric density field $\sigma(x, y, z) \in \mathbb{R}$, and second, an anisotropic radiance field $\boldsymbol{c}(x, y, z, \boldsymbol{d}) \in \mathbb{R}^{N_C}$ where $\boldsymbol{d}$ is the viewing angle and N_C the number of wavelengths. These fields are used to synthesize 2D images using rendering based on ray casting. NeRFs have shown remarkable success in generating photo-realistic new views from unseen angles. This so-called "implicit" representation, a continuous function parametrized by a neural network, handles reflective surfaces and fine details with a relatively low number of parameters compared to explicit representations (voxel grids, meshes, point clouds).

For 3D reconstruction based on multi-date satellite imagery, (Derksen and Izzo, 2021) proposed Shadow-NeRF. Their work handles changes in illumination that occur in real satellite images. To further handle temporal changes, (Marí et al., 2022) added a transient uncertainty predictor to model the appearance or disappearance of objects (*e.g.* vehicles). Sat-NeRF also showed how to use the Rational Polynomial Coefficient (RPC) camera model to perform ray casting for satellite sensors. However, neither of these works are directly adapted for 4D scenes with continuously moving semi-transparent objects.

NeRF variants such as Dynamic-NeRF (D-NeRF) (Pumar-

* Corresponding author

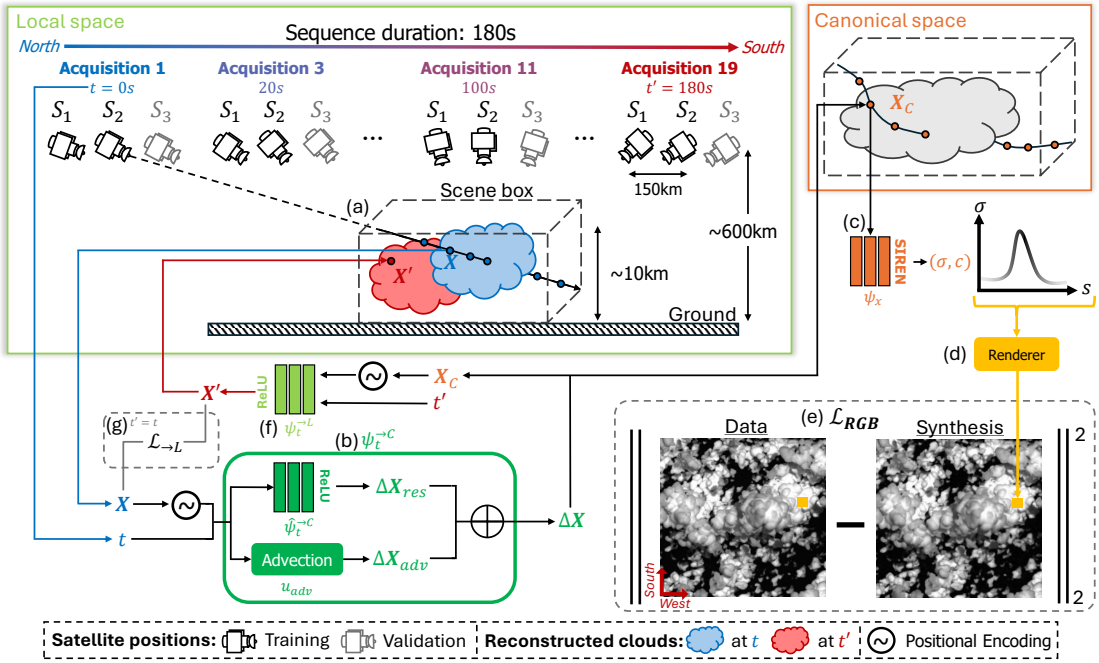

Figure 1. Geometrical configuration of the dataset and overview of Sky-NeRF. At any time t (*e.g.* $0s$), 3D samples X are collected by ray casting (a). Each sample is passed to an advection+residual decomposition $\psi_t^{\rightarrow C} = \boldsymbol{u}_{adv} + \hat{\psi}_t^{\rightarrow C}$ (b) which maps position $\boldsymbol{X}$ in “local” space at time t to position $\boldsymbol{X}_C$ in a time-independent “canonical space”, where the 3D opacity and color distributions are encoded by a network ψ_x (c). The differentiable renderer then blends the predicted samples along the ray to produce a pixel colour (d), compared with the true color through $\mathcal{L}_{RGB}$ (e). For trajectory estimation, the position X' of any point X_C of the canonical space at any time t' (*e.g.* $180s$) is estimated by $\psi_t^{\rightarrow L}$ (f) which encodes the reverse mapping of $\psi_t^{\rightarrow C}$, supervised through the consistency loss term $\mathcal{L}_{\rightarrow L}$ when $t' = t$ (g).

ola et al., 2021) have shown success in handling dynamic (3D+T) scenes. This is achieved by adding a “deformation” network before the NeRF. In a dynamic setting, a first NeRF called $\psi_x : (x, y, z, \boldsymbol{d}) \mapsto (\sigma, \boldsymbol{c})$ describes the radiance field in a canonical space, while the second network $\psi_t : (x', y', z', t) \mapsto (\Delta x, \Delta y, \Delta z)$ describes the deformation of a point at a time t into the canonical space. We focus our study on adapting the D-NeRF paradigm to cloud structures.

Our work shows that with a few adjustments, a D-NeRF architecture leads to a surprisingly accurate 4D reconstruction of cloud structures, even with no physical priors. Sky-NeRF showcases the ability to reconstruct the full 3D cloud structure throughout the entire temporal sequence. This is unlike existing works (Dandini et al., 2022) that provide a partial 3D structure, only on areas of the cloud field that are visible at a given time frame. The contributions of this article are the following:

1. We present adaptations of the D-NeRF architecture, tailored specifically for dynamic cloud scenes (Section 3.1).

2. We propose a strategy to learn trajectories by inverting the deformation field ψ_t (Section 3.2).

3. We introduce a physics-based advection module that disentangles the wind velocity from other local dynamic phenomena (Section 3.3).

4. We evaluate the quality of interpolation between the observed time steps compared to the state-of-the-art stereo photogrammetry pipeline, CARS (Youssefi et al., 2020) (Section 4).

2. Related Works

2.1 3D Reconstruction of Clouds

Cloud topography aims to recover the 3D geometry of the cloud. Tackling this problem from optical cues is an ill-posed task, and the incorporation of physical priors is not straightforward. For instance, when multispectral measurements include appropriate wavelengths, vertical temperature differences can be exploited to approximate cloud thickness using physically derived equations (Yuan and Liang, 2015), statistically estimated variables and shadow-casting constraints (Zhang et al., 2017). In the context of the C^{3}IEL mission, multi-view stereo vision has been applied to pairs of synchronous images to form point clouds, then to pairs of successive asynchronous images to match points at different times (Dandini et al., 2022).

However, tools such as S2P (de Franchis et al., 2014) and CARS (Youssefi et al., 2020) may suffer from the absence of texture. An alternative based on Deep Learning (Dhakal and Mourning, 2024) trains a neural network to predict the altitude of clouds from disparity maps computed on ground-based pairs of images simulated with Blender. Alternatively, (Lin et al., 2023) chains space carving and CNN-extracted 2D features with a 3D-CNN, accounting for dynamics by estimating a constant horizontal wind field via grid search. Neither of these methods attempt to simultaneously learn the 3D structure and deformation field. For natural datasets, supervised approaches like deepMVS (Zhang et al., 2023) and end-to-end methods based on Transformers (Leroy et al., 2024, Wang et al., 2025) have produced unprecedented reconstruction accuracy. However, these methods may suffer from the domain shift caused by the specific scene and acquisition geometries of our problem. Fine-

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-535-2026 | © Author(s) 2026. CC BY 4.0 License.

tuning these models would require a large amount of realistic simulated cloud networks with corresponding ground truths.

2.2 Neural Radiance Fields for Dynamic Scenes

Among recent novel-view synthesis methods, NeRFs stand out for their flexibility and continuous representation. Although initially developed for static, opaque, single-scatter objects, they have been adapted to semi-transparent, dynamic media such as fluids (Chu et al., 2022, Yu et al., 2023, Wang et al., 2024), making them a promising tool for cloud reconstruction.

A dynamic scene introduces a 4D problem that is even more ill-posed than static 3D reconstruction. Simply adding time as an input to a NeRF generally fails. Instead, most research imposes motion regularization, balancing temporal consistency with general priors to avoid over-constraining dynamics. Alternatively, a time-independent canonical geometry can be defined, and the scene's canonical geometry can be deformed at each time. D-NeRF (Pumarola et al., 2021) does so by compositing a time-independent model ψ_x with a time-dependent model ψ_t which maps local geometry at any time t to the canonical space. HyperNeRF (Park et al., 2021) enriches this with a higher dimensional canonical hyperspace that is sliced to yield reconstructions in local frames. A key drawback of these methods is that mappings to canonical space are not guaranteed invertible, complicating trajectory inference. Omnimotion (Wang et al., 2023) remedies this with an invertible warping network, and the authors initialize their algorithm with a pre-computed optical flow. Canonical approaches naturally enforce long-term consistency but struggle with strongly non-rigid deformations.

Of relevance for our problem are works applying NeRF to fluids such as smoke plumes (Chu et al., 2022, Yu et al., 2023, Wang et al., 2024) where physics-driven regularization is incorporated through loss terms which enforce equations involving the density of a transported passive marker and a predicted velocity field of the fluid. Using a NeRF to model the marker's mass density is justified by the Beer-Lambert law which stipulates a proportional relation between mass density and optical density. PINF (Chu et al., 2022) uses two models to predict both the marker opacity and the fluid velocity field. These are coupled not only through photometric consistency as in previous works (Li et al., 2021), but also via a loss enforcing equations of advection. Additional Navier–Stokes constraints (neglecting external forces, pressure and viscosity) are applied to the learnt fields, and vorticity priors from a pretrained model are used to avoid trivial laminar solutions. Adding complex physical constraints like the ones mentioned could be an interesting future consideration. For this work, rather than explicitly enforcing the Navier-Stokes equations, we choose to explicitly model advection in the scene, while using a deformation field to model the motion that cannot be explained simply by advection.

3. Methodology

3.1 Model Architecture and Training

As illustrated in Figure 1, our model is initially based on the two D-NeRF modules. The key difference to D-NeRF is that we also learn the advection motion, as well as the inverse function of the deformation field for trajectory estimation. Sky-NeRF is therefore composed of three modules. First, the time-dependent deformation field $\psi_t^{\rightarrow C}(x, y, z, t) = (\Delta x, \Delta y, \Delta z)$ defines the projection $\boldsymbol{X}_C = \boldsymbol{X} + \Delta\boldsymbol{X}$ of a point $\boldsymbol{X}$ from a local-time space into the time-independent canonical space. To introduce a physical prior on the regularity of the advection motion, $\psi_t^{\rightarrow C}$ is expressed as the sum of two sub-modules, namely, the advection module $\boldsymbol{u}_{adv}$ (described in Section 3.3) and the residual network $\hat{\psi}_t^{\rightarrow C}$. Second, a NeRF $\psi_x(x_C, y_C, z_C) = (\sigma, c)$ takes as input a point in the canonical space and outputs the radiance and density. We remove the dependency of the radiance network on viewing angle so that temporal variations in aspect should be explained using the deformation field. In this way, the learnt radiance c is able to explain only self-shading effects and cloud darkening, but not changes due to movement. Third, an estimate $\psi_t^{\rightarrow L}$ of the inverse of the fixed-time deformation field $\psi_t^{\rightarrow C}(\cdot, t)^{-1}(x_C, y_C, z_C)$ can project a point from canonical space back into local space in order to compute point trajectories (as explained in Section 3.2).

Another important difference is that we follow (Sitzmann et al., 2020, Derksen and Izzo, 2021) and use SIREN activation functions for the NeRF. We find that the implicit prior of SIRENs is more fitting for clouds than ReLU activations.

Following D-NeRF, our training procedure involves marching along rays extending from the camera pixel origins through the scene in the local space. This is performed using RPC camera models as suggested by (Marí et al., 2022). Each 3D point is then projected into the canonical space using the deformation field to perform volumetric rendering. The predicted radiance is compared to the pixel value (photometric consistency loss $\mathcal{L}_{RGB}$) to back-propagate errors through both ψ_x and $\psi_t^{\rightarrow C}$. For canonical-to-local mapping, $\psi_t^{\rightarrow L}$ is trained by comparing the positions of the samples collected during ray marching to positions obtained by projecting these samples back-and-forth through the canonical space. Denoting $\{\boldsymbol{X}_i, t_i\}_{i=1}^N$ the batch of samples collected by ray marching, this is done by complementing the radiometric supervision loss $\mathcal{L}_{RGB}$ with a new term:

$$\mathcal{L}_{\rightarrow L} = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{X}_i - \psi_t^{\rightarrow L}(\psi_t^{\rightarrow C}(\boldsymbol{X}_i, t_i), t_i)\|_2^2. \quad (1)$$

To encode the temporal input, the original D-NeRF architecture uses an encoding similar to the positional encoding used to encode (x, y, z) in NeRF (Mildenhall et al., 2021). In our context, we remove this temporal encoding as we find that the low-frequency bias is consistent with the regime of temporal variation observed at our acquisition frequencies.

3.2 Trajectory Estimation

Following Lagrangian conventions, a meso-particle $\boldsymbol{P}$ can be identified by its position $\boldsymbol{X}_0$ at a given instant t_0. In order to compute its trajectory $t \mapsto \boldsymbol{P}(t)$, we project $\boldsymbol{P}(t_0) = \boldsymbol{X}_0$ into canonical space to retrieve its coordinates in the time-independent space, using the deformation network: $\boldsymbol{P}_C = \psi_t^{\rightarrow C}(\boldsymbol{X}_0, t_0)$. We then use the inverse of the deformation network to extract the coordinate of the canonical point at a later time t, $\boldsymbol{P}(t) = \psi_t^{\rightarrow L}(\boldsymbol{P}_C, t)$. Put together, a trajectory operator can be defined as:

$$\mathcal{T} : (\boldsymbol{X}_0, t_0, t) \mapsto \psi_t^{\rightarrow L}(\psi_t^{\rightarrow C}(\boldsymbol{X}_0, t_0), t). \quad (2)$$

3.3 Advection Module

We evaluate the potential of an advection module to incorporate the physical notion of advection into the model. This can be in-

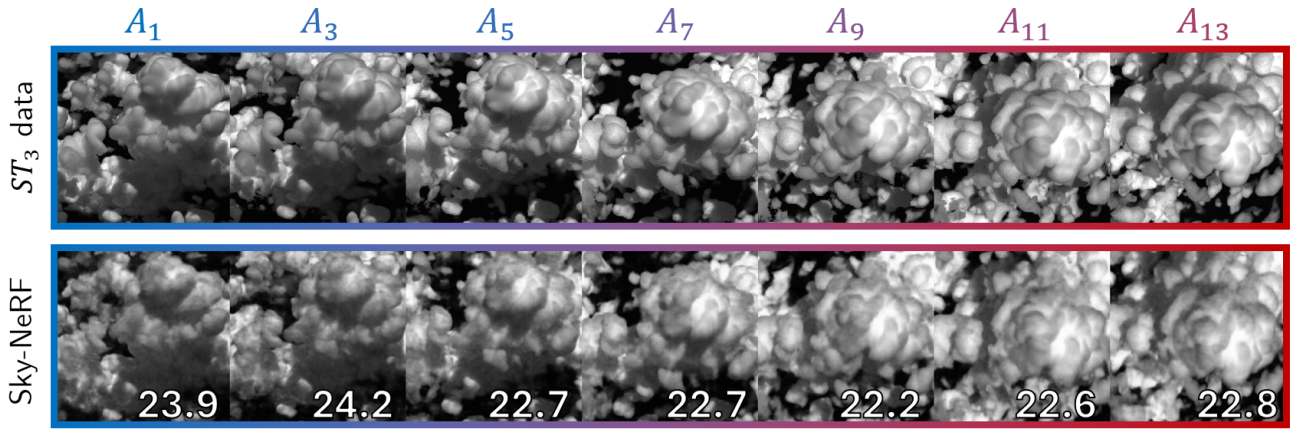

Figure 2. Unseen test images from a third satellite (crops displayed on Top) are globally well reconstructed by Sky-NeRF (Bottom). Although some blurriness remains, this results in a PSNR of 22-24 as indicated in the lower-right corner of the syntheses in the bottom row (with PSNR computed on the corresponding full images). Video results are available at `https://skynerf.github.io/` to qualify Sky-NeRF's capacity to offer an intuitive view of the cloud's motion by synthesizing images at a high frame rate, either from a fixed nadir viewpoint, or as a flyby at fixed time.

terpreted as a 3D wind field that displaces the entire cloud structure in a certain direction throughout the sequence. We suppose a static field of constant direction with a magnitude that depends only on z. As represented in Figure 1, we implement an advection function $\boldsymbol{u}_{adv}$ to enforce these hypotheses while allowing convective movements. The advection function is learnt alongside the residual deformation field $\hat{\psi}_t^{\rightarrow C}$ to disentangle the motion component related to wind from the deformation related to other “local” phenomena (turbulence, condensation, evaporation). Thus, the deformation offset of a point (x, y, z, t) in local space is modeled as:

$$\psi_t^{\rightarrow C}(x, y, z, t) = \boldsymbol{u}_{adv}(z) \times (t - T_0) + \hat{\psi}_t^{\rightarrow C}(x, y, z, t) \quad (3)$$

where T_0 is the date of the first acquisition of the sequence of images. In this way, $\hat{\psi}_t^{\rightarrow C}$ represents the residual motion (unexplained by pure advection). We define advection as $\boldsymbol{u}_{adv}(z) = \rho(z)\boldsymbol{d}_{adv}$, where $\rho(z)$, approximated by a small ReLU network, models the magnitude of the advection field as a function of altitude only, while $\boldsymbol{d}_{adv} \in \mathbb{R}^3$ is a 3D unit vector that models the direction of advection. To train the full Sky-NeRF model, we start by enabling only the advection module. After $N_{\hat{\psi}_t} = 100k$ iterations, we activate the deformation network $\hat{\psi}_t^{\rightarrow C}$ to learn the residual motion. Note that $\boldsymbol{u}_{adv}$ is never frozen during training, as we observed that $\hat{\psi}_t^{\rightarrow C}$ does not absorb the advection modeled by $\boldsymbol{u}_{adv}$ upon activation. This change to the original D-NeRF (Pumarola et al., 2021) affects how the model uses the canonical space by encouraging ψ_x to correspond to the local geometry at T_0, making the canonical space more interpretable. The advection module provides a physical estimation for the direction and magnitude of the advection (experimental results are presented in Section 4.4).

4. Results

4.1 Experimental Setup

4.1.1 Cloud dataset We perform an evaluation of Sky-NeRF on a sequence of optical images simulated to match the geometrical configuration of the C^{3}IEL mission (Dandini et al., 2022). We rely on simulated data as a validation on real-world data would require an accurate 3D model of the cloud surface at various time steps, which is not readily available.

As illustrated in Figure 1, the sequence is composed of 10 pairs of synchronous images with perfect camera models (RPC). Synthetic images are generated from the position of two cameras that follow the same flyby pass over the cloud scene with a baseline of $150km$ between them. The images are simulated at a rate of one pair every $20s$ over a time interval from $T_0 = 0s$ to $T_f = 180s$, resulting in 10 pairs of images labeled ST_jA_i, where ST_j, $j \in \{1, 2\}$ refers to the satellite and A_i, $i \in \{1, 3, \ldots, 19\}$ to the acquisition at time $T_i = T_0 + 10 \times (i - 1)s$. The geometrical configuration is such that the position of satellite 1 at A_i is the position of satellite 2 at A_{i-2}. Images are also available from a third satellite ST_3, which itself is one step ahead of satellite 2. The images from ST_3 are not used during training, but rather set aside as a test set to evaluate the capacity for novel view synthesis from a viewing angle seen at a later time (as illustrated in Figure 2).

The cloud dataset represents a deep convective cloud contained in a $15km \times 15km$ domain with a vertical extent of $13km$, divided into $300 \times 300 \times 260$ cloud cells of resolution $50m$. The images were simulated in two steps. First, the non-hydrostatic mesoscale atmospheric model Meso-NH (Lac et al., 2018) was used to compute physical variables (water content, wind speed, etc.) over the spatial domain and time span. Then, the radiative transfer model 3DMCPOL (Cornet et al., 2010), coupled with geometrical models of the satellites' orbits, attitudes and cameras, was deployed to derive realistic cloud radiance using Monte-Carlo simulations relying on the Mie theory of scattering. The images are simulated in a single wavelength close to the red band, resulting in greyscale images. Therefore, we reduce the dimension of the NeRF color output c from 3 (for RGB) to 1. The dimension of the final images is about $800px \times 800px$. More details on the dataset can be found in a previous study for the C^{3}IEL mission (Dandini et al., 2022).

The dataset also includes a set of points near the surface of the clouds at acquisitions $A_i, i \in \{1, 2, \ldots, 19\}$, which we call the “reference” 3D models PC_i^{ref}. In the dataset, a 3D reference is available at all time steps, which is used for validation of our method in terms of temporal consistency of the learnt 3D cloud structure (see experimental results in Section 4.2).

4.1.2 NeRF implementation and training Our implementation presented in Figure 1 was adapted from nerfstudio (Tancik

et al., 2023). The canonical NeRF ψ_x is parametrized as an 8-layer SIREN architecture with 256 neurons per layer and a skip connection between the 4th and 5th, followed by a 1-layer head for density and a 2-layer head (including 1 hidden layer of width 128) for radiance. As in NeRF (Mildenhall et al., 2021), we use a coarse-to-fine approach to perform hierarchical sampling, effectively duplicating network ψ_x. Models $\hat{\psi}_t^{\rightarrow C}$ and $\psi_t^{\rightarrow L}$ are 4-layer ReLU networks of width 256. To balance terms $\mathcal{L}_{RGB}$ and $\mathcal{L}_{\rightarrow L}$, we set a multiplicative factor $\lambda_{\rightarrow L} = 10$. Training was run for $400k$ iterations, using a rectified Adam optimizer with initial learning rate 6×10^{-4} exponentially decayed to 3×10^{-4}. For the ablations in Table 1, ReLU D-NeRF is trained with a learning rate fixed to 5×10^{-4}. When using the advection module $\boldsymbol{u}_{adv}$, we set $N_{\hat{\psi}_t} = 100k$ iterations. The network ρ used to model advection magnitude is a 2-layer ReLU network of width 32. The advection module is learnt using an Adam optimizer with constant learning rate 6×10^{-4}. Training currently requires 13 hours on an A100 GPU. Faster implementations were explored in early experiments. In particular, the partial implementation of NeRFPlayer (Song et al., 2023) available in nerfstudio reduced processing time to a few dozens of minutes; however, it lacked the decomposition into canonical and local geometries (or any of the advanced NeRFPlayer decompositions). Due to the resulting lack of constraint on reconstruction, performance degraded. Therefore, porting our full method to a faster implementation is left for future work.

4.1.3 CARS stereo baseline We benchmark Sky-NeRF against the stereo-photogrammetry pipeline CARS (Youssefi et al., 2020). The pipeline is run independently for each of the 10 synchronous pairs (A_iST_1, A_iST_2). To work out the disparity map, a dense correlation step uses Semi-Global Matching (Hirschmuller, 2005) with the Census metric computed over $5px \times 5px$ patches. CARS was run with 64 CPU workers and 3GB of RAM allocated to each worker. Running the full pipeline takes about 18 minutes for all pairs.

We wish to compare the performance of Sky-NeRF and CARS, including on regions that are not visible in the images, in other words, regions where the model attempts to reconstruct 3D information from the earlier and later images. Therefore, we work out an occlusion extrapolator for CARS. For a given "anchor" acquisition A_{anch}, a DSM is initialized from the CARS prediction of A_{anch}. The point cloud PC_{anch+2} predicted from a neighboring acquisition, say A_{anch+2}, can be fetched and registered onto A_{anch}. Registration is performed using Iterative Closest Point (ICP) algorithm to estimate a non-rigid deformation that approximates an advection from A_{anch+2} to A_{anch}. PC_{anch+2} is then max-rasterized to complete the DSM. The process continues iteratively to fetch increasingly distant acquisitions, where a distant prediction can be registered by chaining ICPs up to A_{anch}. The order in which acquisitions are visited is closest to A_{anch} first, then closest to A_{11} (nadir flyby for ST_2) to choose between 2 equidistant acquisitions. In the results below, we call this method "CARS (extrapolated)".

Lastly, the comparison can be further extended to temporal interpolation by deriving an advection-based interpolation for CARS that works similarly to the above extrapolator. Namely, if the target acquisition A_{target} is located between two neighboring acquisitions A_{i_1} and A_{i_2}, the advection $T_{A_{i_1} \rightarrow A_{i_2}}$ from A_{i_1} to A_{i_2} is estimated using ICP on the predictions of A_{i_1} and A_{i_2}. PC_{i_1} (either extrapolated or not) is then used as "template" and shifted to the estimated position at A_{anch} by applying half the translation $T_{A_{i_1} \rightarrow A_{i_2}}$. Again, the template is taken closest to nadir (*e.g.* $A_{i_1} = A_3$ and $A_{i_2} = A_1$ if $A_{anch} = A_2$).

4.2 3D Reconstruction

4.2.1 Point cloud comparison In addition to novel images, illustrated in Figure 2, NeRF can be used to recover depth maps from any viewpoint. For any given ray $\boldsymbol{r}$, the median of sample depths weighted by the visibility (opacity times transmittance) is used to estimate the cloud envelop. In some cases however, the ray may not traverse any cloud matter, in which case the depth will be predicted at the far point (on the scene boundary). In other cases, "ghost" matter remains in the 3D model. To prevent this from falsifying the depth maps, a threshold is applied on the accumulation $acc\,(\boldsymbol{r})$, which is the total opacity transmitted along the ray. Therefore, we mask pixels as "background" in the depth map if $acc\,(\boldsymbol{r}) < T_\alpha$. The choice of T_α is loosely related to the optical thickness beyond which a region of space is considered part of a cloud. In our experiments, we set $T_\alpha = 0.85$. A point cloud can then be predicted at any time t by estimating the depth of sampled rays within the available input images until $N_{PC} = 800\,000$ points have been generated. Following the nerfstudio implementation, the resulting point cloud is cleaned to filter out isolated points. For a fair comparison, a similar outlier filtering is applied to CARS. This filtering does not remove the larger clusters of outliers.

We observe in Figure 3 that the point cloud generated by Sky-NeRF seems more complete and appears to contain fewer outliers than the point cloud generated by CARS. Additionally, Sky-NeRF is able to recover realistic geometry not only on cloud tops, but also in regions that are hidden behind the cloud at a given time-frame, illustrating the temporal interpolation capability related to a time-continuous implicit representation.

4.2.2 DSM comparison For the quantitative evaluation, we compare Digital Surface Models (DSMs) obtained by max-rasterizing the point clouds into $300px \times 300px$ DSMs with a resolution of $50m/px$ (chosen to match the resolution of the grid used to produce the reference point clouds). Figure 5 shows the distribution of absolute error and Root Mean Square Error (RMSE) between the predicted and reference DSMs. Errors are here computed as the altitude difference between the compared DSMs. As a general observation, both Sky-NeRF and CARS yield smaller errors and missing matter ratios at dates associated with nadir views, near acquisition 11. This is because large parts of the cloud tops are occluded in off-nadir views (at earlier and later dates). In the case of Sky-NeRF, reconstructing complete geometry of the cloud top at those extreme dates is a more difficult interpolation task than when a nadir viewpoint is immediately available. Comparing the two methods overall, it appears that Sky-NeRF is able to produce DSMs with a lower RMSE but a higher Median Absolute Error (MAE) than CARS. This indicates that the Sky-NeRF prediction contains fewer large errors, but a higher number of small errors.

A direct comparison between the two error distributions is not entirely fair given that CARS only predicts valid pixels on 60-80% of the area, depending on the viewing angle, whereas Sky-NeRF predicts 90-95% of the cloud. Therefore we separately compute the error distribution of Sky-NeRF on areas that are predicted as valid by CARS (typically excluding occluded regions). This decreases both the RMSE and MAE, but Sky-NeRF still does not quite reach the accuracy of CARS in these valid regions. This is confirmed visually in Figure 4, where it appears that Sky-NeRF produces a far cleaner and more complete DSM, albeit with slightly larger errors on the areas where the cloud surface is correctly detected by CARS.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-535-2026 | © Author(s) 2026. CC BY 4.0 License.

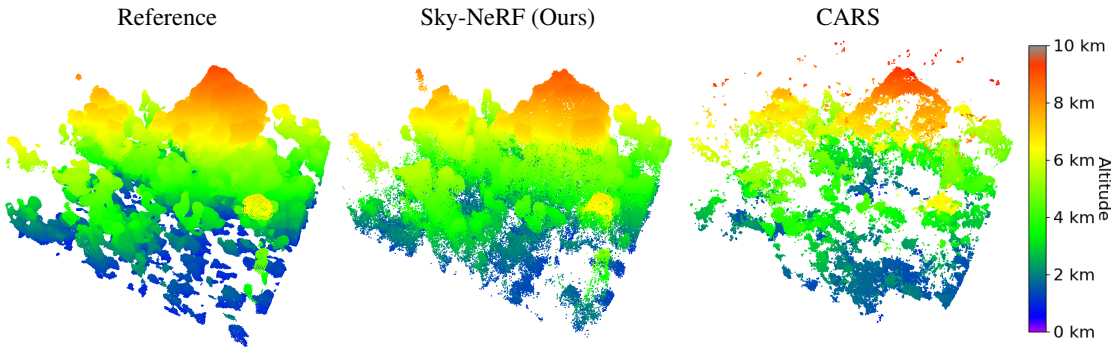

Figure 3. The point cloud at acquisition A_{11} produced by our proposed Sky-NeRF model (Middle) is both more complete and contains fewer outliers than CARS (Right), when compared to the Reference model (Left).

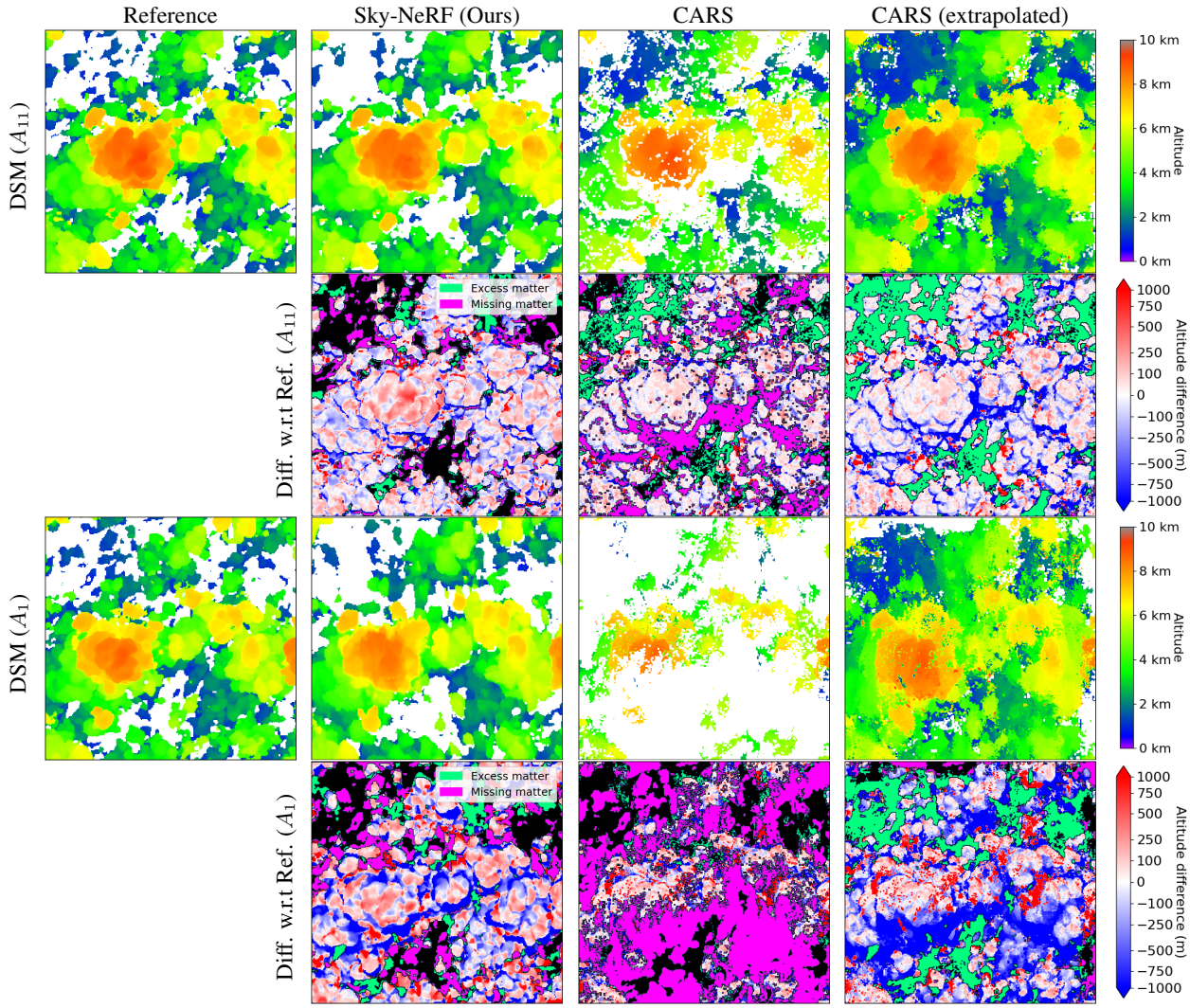

Figure 4. Comparison of the DSMs produced by Sky-NeRF, CARS, and CARS (extrapolated). The color maps of altitude difference are capped at $\pm 1000m$ for readability. We observe that Sky-NeRF paints a more complete picture of the cloud with fewer outliers. On the visible parts of the cloud, CARS remains more accurate. The two lower rows show a time step (A_1) where only one side of the cloud is visible in the images. Here, Sky-NeRF showcases the ability to perform temporal extrapolation of the cloud structure by reconstructing regions that are not directly visible.

Additionally, we show the error distribution of CARS (extrapolated) as defined in Section 4.1.3. Figure 5 shows that the naive extrapolation method is not nearly as accurate as Sky-NeRF, particularly when the satellites are off-nadir. This is confirmed qualitatively in Figure 4, where the CARS extrapolator makes more severe errors in occluded regions opposite the satellites' positions. This experiment alone is not sufficient to fully conclude that Sky-NeRF can extract the exact cloud structure and motion in unseen areas. Nonetheless, it seems that the 3D structure predicted by Sky-NeRF in hidden areas is more accurate than a “copy-pasted” version of the cloud from before or after.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-535-2026 | © Author(s) 2026. CC BY 4.0 License.

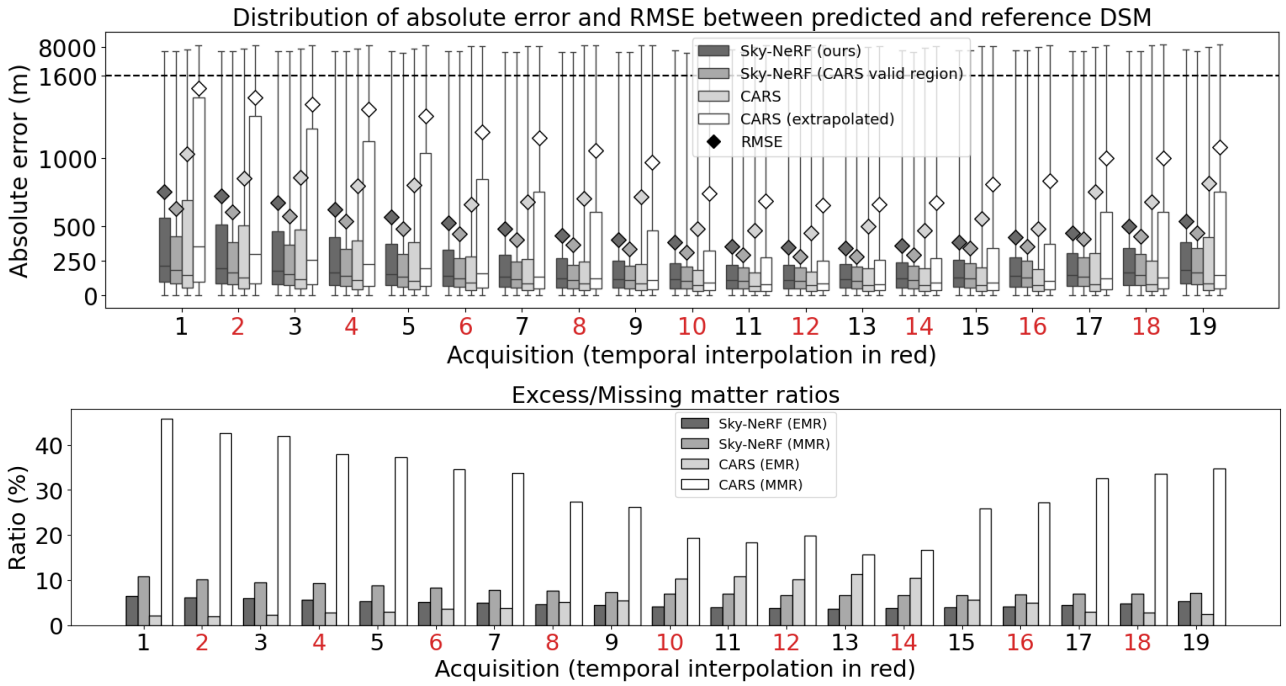

Figure 5. Evaluation of the DSMs, by date and method. Top: distribution of errors *w.r.t.* reference DSM. Boxplots indicate extrema, 1st and 3rd quartiles and median. The Y-axis is linear below $1600m$, then log-scaled. Bottom: excess and missing matter ratios, respectively measuring the portion of the predicted DSM containing a cloud altitude value when the reference DSM does not (green in Figure 4), and reversely (purple).

Looking into Figure 5 at even acquisitions reveals temporal interpolation capacities. Interestingly, both approaches yield steady performance at those dates compared to neighboring acquisition dates. This first indicates that both methods correctly estimate advection between acquisitions, as advection is the main contribution to the overall motion. Secondly, the fact that the gap between median errors of CARS and Sky-NeRF does not get narrower either implies that Sky-NeRF does not finely estimate convection, or that the variance of the error made by Sky-NeRF around the reference geometry is greater than the order of magnitude of the motion of convection.

Taken together, these observations indicate that, at their current stage of development, the two methods should be regarded as complementary. Specifically, Sky-NeRF provides intuitive and comprehensive visualizations that support a global understanding of the cloud geometry. At the same time, it constitutes an initial step toward linking information across acquisitions, thereby enabling the tracking of individual meso-particles throughout the sequence, including across occlusions. In contrast, CARS retains a modest advantage in terms of median error over visible regions of the clouds. The practical significance of the remaining performance gap for downstream applications remains to be quantified, particularly for the analysis of fine-scale turbulent motions, which require consistently accurate reconstructions over the full duration of the sequence.

4.3 Trajectory Prediction

By introducing the approximated inverse $\psi_t^{\rightarrow L}$, the trajectory of meso-particles can be directly inferred in the form of a continuous trajectory field $\mathcal{T}$. To evaluate the quality of these predicted trajectories, reference trajectories are worked out using the reference point clouds $PC_i^{ref}, i \in \{1, \dots, 19\}$. Choosing a point $P_1 \in PC_1^{ref}$, the latter can be iteratively propagated to subsequent acquisitions to give a discrete-time trajectory $(P_i)_{i\in\{1,\dots,19\}}$. The displacement from A_i to A_{i+1} is approximated by first computing a global advection motion using ICP between PC_i^{ref} and PC_{i+1}^{ref}. Then, the residual motion between PC_i^{ref} (shifted by ICP) and PC_{i+1}^{ref} is computed using the M3C2 distance (Lague et al., 2013). Given two point clouds PC_1 and PC_2, M3C2 measures the distances of points $\boldsymbol{P} \in PC_1$ to point cloud PC_2 along normals of PC_1. The reference trajectories obtained this way are precise up to the quality of the advection computed by ICP, but also to the approximation made when considering that the residual motion of particles is aligned with the cloud's normals.

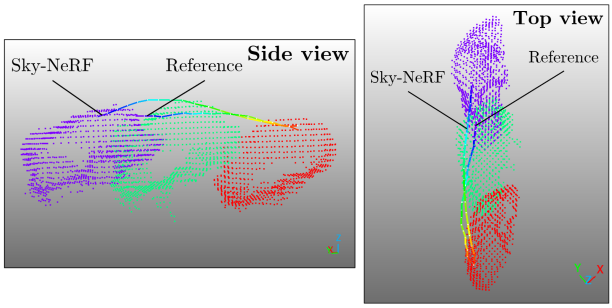

Figure 6. Example of reference trajectory and trajectory predicted by Sky-NeRF for a point initialized in PC_1^{ref}. Crops of the reference point clouds PC_1^{ref} (red), PC_{11}^{ref} (green) and PC_{19}^{ref} (purple) are shown.

Figure 6 shows a trajectory $(\mathcal{T}(\boldsymbol{P_1}, T_1, T_i))_{i\in\{1,\cdots,19\}}$ predicted by Sky-NeRF, along with the corresponding reference trajectory, for a point P_1 taken in PC_1^{ref} at A_1. The model can track the cloud's surface, but does not consistently track a single meso-particle throughout the entire sequence, resulting in a gap between the reference and predicted final positions. Therefore,

Acquisition	A_1	A_2	A_7	A_8	A_{11}	A_{12}	A_{15}	A_{16}	A_{19}
ReLU D-NeRF	17.64 / 0.46 1720	- 1515	17.12 / 0.43 950	- 1016	16.98 / 0.43 655	- 1160	18.05 / 0.45 751	- 882	18.72 / 0.50 1109
SIREN D-NeRF	22.52 / 0.69 772	- 1248	21.35 / 0.62 490	- 541	21.72 / 0.63 354	- 790	22.94 / 0.67 380	- 454	22.45 / 0.66 555
Sky-NeRF w/o u_{adv} & $\psi_t^{\rightarrow L}$	23.87 / **0.71** **750**	- **703**	22.12 / 0.64 488	- 445	**22.64 / 0.65** 367	- 360	**23.74 / 0.69** 392	- 424	**22.69 / 0.68** 555
Sky-NeRF w/o $\psi_t^{\rightarrow L}$	23.58 / 0.69 759	- 730	22.58 / 0.64 509	- 449	22.37 / 0.63 **336**	- **329**	22.87 / 0.66 **376**	- **416**	21.77 / 0.63 578
Sky-NeRF	**23.91 / 0.71** 754	- 721	**22.72 / 0.65** **484**	- **436**	22.62 / 0.64 355	- 349	23.22 / 0.67 388	- 422	22.28 / 0.65 **539**

Table 1. Impact of the Sky-NeRF adaptations on the PSNR(↗)/SSIM(↗) on test views (top of each cell) and on the RMSE(↘) of the DSM in meters (bottom). All experiments have direction encoding disabled. Moving from “SIREN D-NeRF” to “Sky-NeRF w/o u_{adv} & $\psi_t^{\rightarrow L}$” is done by removing temporal encoding. Red acquisitions indicate temporal interpolation (no input image available). Architecture adaptations (3 first rows) yield improvements in image synthesis and geometry, while u_{adv} and $\psi_t^{\rightarrow L}$ (2 last rows) produce physical outputs without significantly affecting performance.

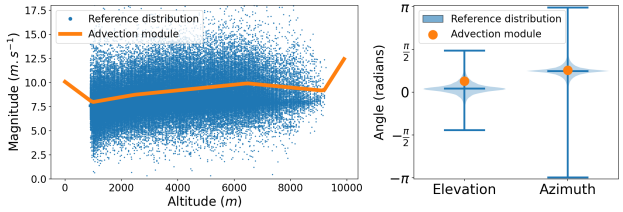

Figure 7. Evaluation of the velocity predicted by the advection module (orange), against estimated advection in the reference point clouds (blue). Left: The learnt magnitude $\rho(z)$ follows the average trend of the reference magnitude. Right: The advection azimuth is accurate, with a slight overestimation of the updraft.

Sky-NeRF is not able to precisely track single particles at fine scales, but appears to have the capacity to track global cloud cell motions.

4.4 Direct Advection Estimation

Figure 7 (Left) shows the magnitude of advection $\rho(z)$, compared to the velocities derived from the reference point clouds. Specifically, each data point is the advection velocity estimated from a point in PC_i^{ref} to PC_{i+1}^{ref}, where i can be any date in $\{1, \ldots, 18\}$. The method to obtain these estimated velocities is explained in Section 4.3. This figure illustrates that the vertical profile of the wind learnt by the advection module seems consistent with the average motion of the reference point clouds, at altitudes where clouds are present. Figure 7 (Right) shows that the elevation and azimuth of the advection vector d_{adv} (angles from the horizontal plane upward and from the North axis toward the East, respectively) are estimated correctly. The ablations in Table 1 confirm that our proposed advection module is able to learn the overall motion of the cloud accurately without negatively affecting the novel-view synthesis or 3D reconstruction performance.

By design, the dynamic radiance field does not allow for extrapolation beyond the input time series. Nonetheless, by introducing advection as a physical prior, our model can extrapolate the overall position of a cloud structure within a few minutes before or after the observed sequence. This experiment is a first step toward including more physical priors into the system and suggests that, in the future, similar approaches may enable the extraction of physical information from observed cloud data, beyond mere 3D structure.

5. Conclusion

This study is a first step toward the application of implicit neural representations for the extraction of cloud structure and motion based on sequences of satellite stereo-imagery. We propose adaptations to the Dynamic Neural Radiance Fields paradigm, tailored to the task of 4D cloud reconstruction. Our experiments showcase the ability of continuous implicit representations to model complex phenomena such as cloud structures. While perfectible, these results show that simple physical priors can be added in a NeRF paradigm to extract the advection field and decouple various sources of motion.

Direct prediction of cloud motion by a differentiable network paves the way for future work on the regularization of the motion through physics-driven consistency terms applied on the particles’ trajectories, akin to ideas explored in the literature for the reconstruction of smoke plumes (Chu et al., 2022, Yu et al., 2023, Wang et al., 2024). Doing so, we expect cloud mesoparticles to be tracked more consistently, and subsequently to improve the model geometry and long-term temporal consistency. Another notable development path consists in combining the strengths of the two methods compared in this work, as the outputs of CARS could be used as guidance for Sky-NeRF. This could be done as in SpS-NeRF (Zhang and Rupnik, 2023), by guiding Sky-NeRF’s ray termination distributions closer to CARS depth, although particular attention should be taken in handling outliers of the reconstructions produced by CARS.

Regarding scalability, the proposed method has been evaluated on images of moderate dimension but is, in principle, applicable to larger scenes. Extremely large scales may, however, incur significant memory demands during training, which could be mitigated by incorporating strategies developed in prior work adapting NeRF to Earth observation, such as Snake-NeRF (Billouard et al., 2025). In addition, Sky-NeRF would benefit from implementation in more efficient frameworks, for example by leveraging the hash encoding of Instant-NGP (Billouard et al., 2024) or by investigating the alternative paradigm of Gaussian Splatting (Aira et al., 2025).

6. Acknowledgments

We gratefully acknowledge the Laboratoire d’Optique Atmosphérique for providing access to the cloud simulation used in this study. We also sincerely thank CNES for their financial support and for granting access to high-performance computing resources at their computing center.

References

Aira, L. S., Facciolo, G., Ehret, T., 2025. Gaussian splatting for efficient satellite image photogrammetry. *Proceedings of the Computer Vision and Pattern Recognition Conference*, 5959–5969.

Billouard, C., Derksen, D., Constantin, A., Vallet, B., 2025. Tile and slide: A new framework for scaling nerf from local to global 3d earth observation. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3719–3729.

Billouard, C., Derksen, D., Sarrazin, E., Vallet, B., 2024. Satngp: Unleashing neural graphics primitives for fast relightable transient-free 3d reconstruction from satellite imagery. *IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium*, IEEE, 8749–8753.

Chu, M., Liu, L., Zheng, Q., Franz, E., Seidel, H.-P., Theobalt, C., Zayer, R., 2022. Physics Informed Neural Fields for Smoke Reconstruction with Sparse Data. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 41(4), 119:1–119:15.

Cornet, C., C-Labonnote, L., Szczap, F., 2010. 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.

Dandini, P., Cornet, C., Binet, R., Fenouil, L., Holodovsky, V., Y. Schechner, Y., Ricard, D., Rosenfeld, D., 2022. 3D cloud envelope and cloud development velocity from simulated CLOUD (C3IEL) stereo images. *Atmospheric Measurement Techniques*, 15(20), 6221–6242.

de Franchis, C., Meinhardt-Llopis, E., Michel, J., Morel, J.-M., Facciolo, G., 2014. An automatic and modular stereo pipeline for pushbroom images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3.

Derksen, D., Izzo, D., 2021. Shadow neural radiance fields for multi-view satellite photogrammetry. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1152–1161.

Dhakal, R., Mourning, C., 2024. Synthetic Cloud Height Prediction Using Stereo Matching and Deep Learning. *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, 8278–8283.

Hirschmuller, H., 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, 2, IEEE, 807–814.

Lac, C., Chaboureau, J.-P., Masson, V., Pinty, J.-P., Tulet, P., Escobar, J., Leriche, M., Barthe, C., Aouizerats, B., Augros, C. et al., 2018. Overview of the Meso-NH model version 5.4 and its applications. *Geoscientific Model Development*, 11(5), 1929–1969.

Lague, D., Brodu, N., Leroux, J., 2013. Accurate 3D comparison of complex topography with terrestrial laser scanner: Application to the Rangitikei canyon (NZ). *ISPRS journal of photogrammetry and remote sensing*, 82, 10–26. Publisher: Elsevier.

Leroy, V., Cabon, Y., Revaud, J., 2024. Grounding image matching in 3d with mast3r. *European conference on computer vision*, Springer, 71–91.

Li, Z., Niklaus, S., Snavely, N., Wang, O., 2021. Neural scene flow fields for space-time view synthesis of dynamic scenes. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6498–6508.

Lin, J., Farinha, M., Gryspeerdt, E., Clark, R., 2023. Volumetric cloud field reconstruction. *arXiv preprint arXiv:2311.17657*.

Marí, R., Facciolo, G., Ehret, T., 2022. Sat-nerf: Learning multi-view satellite photogrammetry with transient objects and shadow modeling using rpc cameras. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1311–1321.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., Ng, R., 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1), 99–106.

Park, K., Sinha, U., Hedman, P., Barron, J. T., Bouaziz, S., Goldman, D. B., Martin-Brualla, R., Seitz, S. M., 2021. HyperNeRF: A Higher-Dimensional Representation for Topologically Varying Neural Radiance Fields. *ACM Trans. Graph.*, 40(6). Publisher: ACM.

Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F., 2021. D-NeRF: Neural Radiance Fields for Dynamic Scenes. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10313–10322.

Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G., 2020. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33, 7462–7473.

Song, L., Chen, A., Li, Z., Chen, Z., Chen, L., Yuan, J., Xu, Y., Geiger, A., 2023. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics*, 29(5), 2732–2742.

Strauss, C., Ricard, D., Lac, C., Verrelle, A., 2019. Evaluation of turbulence parametrizations in convective clouds and their environment based on a large-eddy simulation. *Quarterly Journal of the Royal Meteorological Society*, 145(724), 3195–3217.

Tancik, M., Weber, E., Ng, E., Li, R., Yi, B., Kerr, J., Wang, T., Kristoffersen, A., Austin, J., Salahi, K., Ahuja, A., McAllister, D., Kanazawa, A., 2023. Nerfstudio: A modular framework for neural radiance field development. *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23.

Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D., 2025. Vggt: Visual geometry grounded transformer. *Proceedings of the Computer Vision and Pattern Recognition Conference*, 5294–5306.

Wang, Q., Chang, Y.-Y., Cai, R., Li, Z., Hariharan, B., Holynski, A., Snavely, N., 2023. Tracking everything everywhere all at once. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19795–19806.

Wang, Y., Tang, S., Chu, M., 2024. Physics-Informed Learning of Characteristic Trajectories for Smoke Reconstruction. *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH ’24.

Youssefi, D., Michel, J., Sarrazin, E., Buffe, F., Cournet, M., Delvit, J.-M., L’Helguen, C., Melet, O., Emilien, A., Bosman, J., 2020. CARS: A Photogrammetry Pipeline Using Dask Graphs to Construct a Global 3D Model. *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 453–456.

Yu, H.-X., Zheng, Y., Gao, Y., Deng, Y., Zhu, B., Wu, J., 2023. Inferring hybrid neural fluid fields from videos. *Advances in Neural Information Processing Systems*, 36, 63595–63608.

Yuan, C., Liang, X., 2015. Derivation of 3D cloud animation from geostationary satellite images. *Multimedia Tools and Applications*, 75, 8217 – 8237.

Zhang, L., Rupnik, E., 2023. SparseSat-NeRF: Dense Depth Supervised Neural Radiance Fields for Sparse Satellite Images. *ISPRS Annals*.

Zhang, Z., Liang, X., Yuan, C., Li, F. W., 2017. Modeling Cumulus Cloud Scenes from High-resolution Satellite Images. *Computer Graphics Forum*, 36(7), 229–238.

Zhang, Z., Peng, R., Hu, Y., Wang, R., 2023. Geomvsnet: Learning multi-view stereo with geometry perception. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 21508–21518.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-535-2026 | © Author(s) 2026. CC BY 4.0 License.