



RESEARCH ARTICLE

10.1029/2023MS004047

Special Collection:

Machine learning application to Earth system modeling

Key Points:

- We developed a realistic simulation of satellite observations of sea surface height and temperature
- We compare deep learning supervised and unsupervised strategies to interpolate the sea surface height
- We present a hybrid learning strategy using supervised training on our simulated data and unsupervised fine-tuning on real-world data

Correspondence to:T. Archambault,
theo.archambault@lip6.fr**Citation:**

Archambault, T., Filoche, A., Charantonis, A., Béreziat, D., & Thiria, S. (2024). Learning sea surface height interpolation from multi-variate simulated satellite observations. *Journal of Advances in Modeling Earth Systems*, 16, e2023MS004047. <https://doi.org/10.1029/2023MS004047>

Received 29 SEP 2023

Accepted 3 MAY 2024

Author Contributions:

Conceptualization: Théo Archambault, Arthur Filoche, Dominique Béreziat, Sylvie Thiria

Data curation: Théo Archambault

Formal analysis: Théo Archambault, Anastase Charantonis, Dominique Béreziat

Funding acquisition:

Anastase Charantonis, Dominique Béreziat, Sylvie Thiria

Investigation: Théo Archambault

Methodology: Théo Archambault, Arthur Filoche, Anastase Charantonis

Project administration:

Dominique Béreziat, Sylvie Thiria

© 2024 The Authors. *Journal of Advances in Modeling Earth Systems* published by Wiley Periodicals LLC on behalf of American Geophysical Union.

This is an open access article under the terms of the [Creative Commons](#)

[Attribution License](#), which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Learning Sea Surface Height Interpolation From Multi-Variate Simulated Satellite Observations

Théo Archambault^{1,2} , Arthur Filoche³, Anastase Charantonis^{2,4,5}, Dominique Béreziat¹ , and Sylvie Thiria²

¹Sorbonne Université, CNRS, LIP6, Paris, France, ²Sorbonne Université, CNRS, IRD, MNHN, LOCEAN, Paris, France,

³University of Western Australia, Oceans Institute, Perth, Australia, ⁴ENSIIE, CNRS, LaMME, Evry, France, ⁵Inria, Paris, France

Abstract Satellite-based remote sensing missions have revolutionized our understanding of the Ocean state and dynamics. Among them, space-borne altimetry provides valuable Sea Surface Height (SSH) measurements, used to estimate surface geostrophic currents. Due to the sensor technology employed, important gaps occur in SSH observations. Complete SSH maps are produced using linear Optimal Interpolations (OI) such as the widely used Data Unification and Altimeter Combination System (DUACS). On the other hand, Sea Surface Temperature (SST) products have much higher data coverage and SST is physically linked to geostrophic currents through advection. We propose a new multi-variate Observing System Simulation Experiment (OSSE) emulating 20 years of SSH and SST satellite observations. We train an Attention-Based Encoder-Decoder deep learning network (ABED) on this data, comparing two settings: one with access to ground truth during training and one without. On our OSSE, we compare ABED reconstructions when trained using either supervised or unsupervised loss functions, with or without SST information. We evaluate the SSH interpolations in terms of eddy detection. We also introduce a new way to transfer the learning from simulation to observations: supervised pre-training on our OSSE followed by unsupervised fine-tuning on satellite data. Based on real SSH observations from the Ocean Data Challenge 2021, we find that this learning strategy, combined with the use of SST, decreases the root mean squared error by 24% compared to OI.

Plain Language Summary The surface of the ocean is observed through various sensors embedded in satellites. Specifically, the height of the sea surface is a crucial variable as it can be used to estimate surface currents. It is currently measured through satellite altimeters, but the data acquisition process leaves gaps in their observations. Providing fully gridded maps of the sea surface height is thus an important interpolation problem. The widely used interpolated product has some troubles, especially when dealing with small and rapidly evolving eddies. To enhance its quality, we propose an artificial neural network, a trainable method able to estimate complete sea surface height images. The flexibility of these methods allows us to use different satellite information, such as the sea surface temperature, which has a better resolution. Usually, neural networks are trained on a data set upon which they learn the link between input and output data. However, in a realistic geoscience scenario, the output is never known for sure. We propose a data set that simulates this problem and explores methodologies to train these methods. Finally, we design a new way of learning SSH reconstruction using both simulated data and real satellite observation.

1. Introduction

1.1. Background

Since the first ocean remote sensing missions in the 1970s, satellite observation has become one of the most determining contributions to understanding ocean state and dynamics (S. Martin, 2014). Through the years, satellites have provided a huge amount of data of various physical natures with wide spatial coverage that complemented in situ data sets. Among these techniques, satellite altimetry is used to retrieve the Sea Surface Height (SSH) a determining variable of the ocean circulation. The SSH spatial gradient can be used to estimate geostrophic circulation, that is, the currents issued from the equilibrium between the Coriolis force and the pressure force in the surface layer of the Ocean. SSH (also called Absolute Dynamical Topography by the altimetry community) is currently measured by nadir-pointing altimeters, meaning that they can only take measurements vertically, along their ground tracks, by calculating the return time of a radar pulse. This leads to large gaps in the observed SSH, and providing a gap-free product (L4) is a challenging Spatio-Temporal

Resources: Théo Archambault, Dominique Béréziat

Software: Théo Archambault

Supervision: Anastase Charantonis, Dominique Béréziat, Sylvie Thiria

Validation: Théo Archambault, Dominique Béréziat

Visualization: Théo Archambault

Writing – original draft:

Théo Archambault, Arthur Filoche

Writing – review & editing:

Théo Archambault, Arthur Filoche, Anastase Charantonis, Dominique Béréziat, Sylvie Thiria

interpolation problem. One of the most widely used L4 products in oceanography applications is provided by the Data Unification and Altimeter Combination System (DUACS) (Taburet et al., 2019). It is a linear Optimal Interpolation (OI) of the nadir along-track measurements leveraging a covariance matrix tuned on 25 years of data. However, several studies show that DUACS OI misses some of the mesoscales structures and eddies (Amores et al., 2018; Stegner et al., 2021). Improving the reconstruction of gridded altimetry products remains an open challenge.

To enhance the quality of the SSH reconstruction and sea surface current estimation, using additional physical information such as the Sea Surface Temperature (SST) has been demonstrated to be beneficial (Archambault et al., 2023; Ciani et al., 2020; Fablet et al., 2023; Thiria et al., 2023; S. A. Martin et al., 2023). SST motion is linked to ocean circulation (Isen-Fontanet et al., 2006), and therefore to SSH, as currents transport heat in an advection dynamics. SST measurements obtained through passive infrared technology offer a remarkably high spatial resolution, ranging from 1.1 to 4.4 km (Emery et al., 1989), even if intermittent clouds introduce data gaps. On the other hand, microwave sensors provide lower-resolution SST data (25 km) which can be obtained through non-raining clouds. Infrared and microwave data are then combined with in situ measurements, to produce fully gridded SST maps (Chin et al., 2017; Donlon et al., 2012). Thus, a crucial challenge lies in developing efficient reconstruction methods capable of fusing data derived from different remote sensing techniques, each presenting distinct interpolation challenges. This is essential to unlock the full potential of satellite oceanography products.

1.2. SSH Interpolation With Deep Neural Networks

In the last decade, deep learning has emerged as one of the leading methods to address image inverse problems. Neural networks demonstrated remarkable flexibility in fusing observations from various sources and modalities, exhibiting their capacity to learn complex relationships given enough training samples (McCann et al., 2017; Ongie et al., 2020). Prior works proved that it is possible to use SST to enhance SSH reconstruction with a deep-learning network, whether from a downscaling perspective (Nardelli et al., 2022; Thiria et al., 2023) or an interpolation one (Fablet et al., 2023; S. A. Martin et al., 2023). However, training neural networks usually requires fully gridded ground truth, which is unavailable in a realistic geoscience scenario. To overcome this limitation, it is possible to design a twin experiment of the satellite observing system on a numerical simulation, also called an Observing System Simulation Experiment (OSSE). Neural networks can then be trained on simulated data and applied to satellite observations. The Ocean Data Challenge 2020 (CLS/MEOM, 2020) is a 1-year OSSE providing SSH simulated observations and ground truth, aiming to compare various reconstruction methods. Among them, Fablet et al. (2021) performed a supervised deep learning of the SSH interpolation and extended their study using SST showing increased performance (Fablet et al., 2023). However, if the SSH-only network was successfully applied to real data, adapting its SST-using version is still a challenging problem. Another way to overcome the lack of ground truth is to employ loss functions allowing the neural network to learn from observations alone. Archambault et al. (2023); S. A. Martin et al. (2023) trained a neural network using only SST and SSH observations showing the potential of unsupervised learning for SSH interpolation. This last option has the advantage of not suffering from the domain gap between the simulation and the real data, but we expect unsupervised interpolations to produce less accurate reconstructions.

1.3. Contributions

First, as the previously existing Ocean Data Challenge OSSE provided only 1 year of data without SST, it presents clear limitations to train neural networks. We propose a new OSSE that includes 20 years of SSH and SST data, with realistic simulated observations of these variables.

Second, we compare a fixed neural architecture trained in a supervised and unsupervised way, with or without SST. The SSH interpolation is learned by an Attention-Based Encoder-Decoder (ABED) on our OSSE. Its assessment involves evaluating errors in SSH and geostrophic currents reconstruction. Additionally, a comparison of the eddy structures is conducted, both quantitatively and visually.

Third, we propose a hybrid learning strategy consisting of supervised pre-training on our OSSE and unsupervised fine-tuning on real-world observations. Specifically, we compare the same network architecture, trained in the three following manners: supervised on our OSSE and directly applied to observations, trained directly on observations, and the proposed hybrid approach.

This paper is structured as follows. In Section 2, after giving a rationale for leveraging SST information in the interpolation method, we detail our OSSE. In Section 3 we present our architecture and the training loss functions. In Section 4 we evaluate the interpolation on our OSSE, in terms of SSH reconstruction and geostrophic circulation errors. We also perform an eddy detection to demonstrate that SST-using methods retrieve more realistic ocean structures, and we compare ourselves to existing state-of-the-art methods on the Ocean Data Challenge 2020 OSSE. Finally, we compare the learning strategies on real observations. In Section 5, we discuss the limitations and perspectives of this work.

2. Multi-Variate Data

In the following, we provide a rationale for the SSH and SST relationship, outline the reference data source we utilized (Global Ocean physics Reanalysis (CMEMS, 2020)), and detail our OSSE's SSH and SST observations. We also present the satellite observations that will be used for training and fine-tuning.

2.1. Physical Relationship Between SSH and SST

One of the most important uses of SSH data is to recover oceanic currents through geostrophic approximation. It consists of supposing a static equilibrium between the surface projection of the Coriolis force and the resultant pressure forces. Far from the Equator, where the Coriolis force projection is null, it is a good approximation of the circulation. The surface geostrophic currents can be computed from the SSH h following Equation 1

$$\mathbf{w}_{\text{geo}} = \begin{pmatrix} u_{\text{geo}} \\ v_{\text{geo}} \end{pmatrix} = \begin{pmatrix} -\frac{g}{f} \frac{\partial h}{\partial y} \\ \frac{g}{f} \frac{\partial h}{\partial x} \end{pmatrix} \quad (1)$$

where u_{geo} and v_{geo} are the Eastward and Northward geostrophic currents, x and y the Eastward and Northward coordinates and where $f = 2\Omega_r \sin(\phi)$ is the Coriolis factor, Ω_r being the Earth the rotation period, ϕ the latitude and g the gravitational acceleration.

In a first approximation, the surface temperature T can be considered as a passive tracer transported by surface currents. The evolution of a scalar in a velocity field is described by the linear advection given in Equation 2.

$$\frac{\partial T}{\partial t} + \mathbf{w} \cdot \nabla T = 0 \quad (2)$$

Combining the geostrophic and the advection Equations 1 and 2, we understand why a time series of SST observations should provide pertinent information for constraining the SSH reconstruction. Several studies pointed out the interest in using SST to reconstruct SSH as Isern-Fontanet et al. (2006) and González-Haro et al. (2020), which established spectral relations between SSH and SST in a Surface Quasi Geostrophic framework. However, the physical link between temperature and SSH is more complex, as other phenomena must be considered, such as diffusion, convection, circulation between water depths, atmosphere interactions, and viscosity. Satellite observations of temperature and sea surface height also suffer from instrumental errors and are, by nature, limited to observing the ocean surface. This is why neural network architectures, thanks to their flexibility, seem appropriate to learn the complex underlying link between the data.

2.2. Observing System Simulation Experiment

To effectively replicate the relationship between the two variables, we propose an Observing System Simulation Experiment (OSSE), meaning a twin experiment that accurately models the satellite observations of the Ocean. This approach is widely used in the geosciences community as it provides a way to test reconstruction methods and errors (Amores et al., 2018; Gaultier et al., 2016; Stegner et al., 2021). With this mindset, SSH and SST variables of a high-resolution simulation are considered as the ground truth ocean state upon which we simulate satellite measurements. The coherence of the relation between SSH and SST is ensured by the physical model, while with our OSSE we produce enough pairs of ground truth/observation to train a neural network.

In this paper, we denote \mathbf{X}^{ssh} and \mathbf{X}^{sst} the ground truth fields of the SSH and SST and \mathbf{Y}^{ssh} and \mathbf{Y}^{sst} , the simulated observations. Hereafter, we detail the reference data set of our OSSE and the observation operators of the two variables.

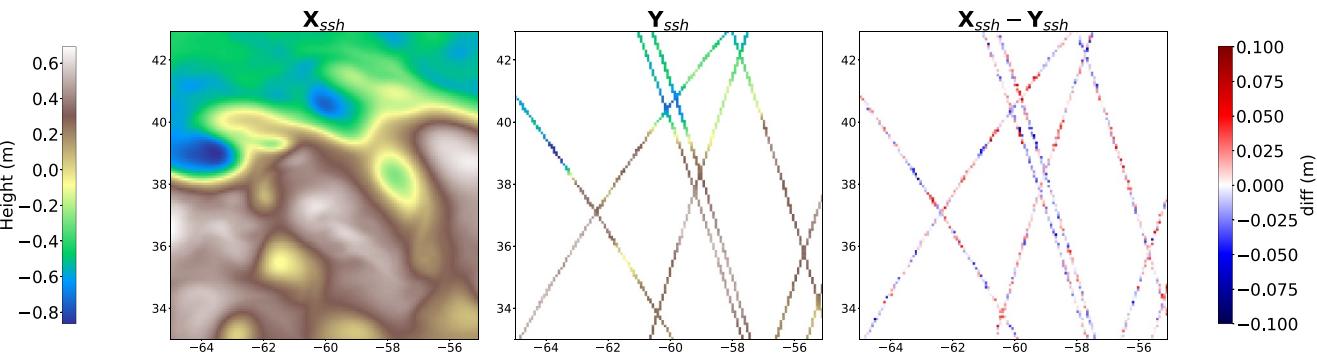


Figure 1. Images of the ground truth SSH from GLORYS12, the simulated along-track measurements, and the difference.

2.2.1. Base Simulation

We conduct our experiments on the Global Ocean Physics Reanalysis product (GLORYS12) (CMEMS, 2020). It provides various physical data such as SSH, SST, and oceanic currents with a spatial resolution of $1/12^\circ$ (around 8 km). GLORYS12 is based on the NEMO 3.6 model (Madec et al., 2017) and assimilates satellite observations (SSH along-track observations and SST full domain observations) through a reduced-order Kalman filter. It is updated annually by the Copernicus European Marine Service, making it impossible to use in near real-time applications. We select a temporal subset of this simulation from 2000/03/20/ to 2019/12/29, for a total of 7,194 days.

We select a portion of the Gulf Stream, between 33° to 43° North and -65° to -55° East. This area is known for its intense circulation, its water mass of very different temperatures, and is far enough from the equator that the geostrophic approximation can be applied. Comparing the surface circulation of the model with its geostrophic approximation, we find that an RMSE of 6.6 cm/s for u_{geo} and 6.1 cm/s for v_{geo} . Considering the high intensity and variations of the currents in the Gulf Stream (with 37.1 and 34.3 cm/s of standard deviation for u and v respectively), geostrophy seems to be an adequate estimation. Thus, we expect a significant synergy between SSH and SST which a neural network can learn. For computational reasons, we resample the data to images of size 128×128 with a bilinear interpolation, corresponding to a resolution of 0.078° by pixel (approximately 8.7 km). Doing so, the perceptive field of the network covers the entire 10° by 10° area.

2.2.2. SSH Simulated Observations

The nadir-pointing altimetry satellites take approximately a measurement per second, along their ground tracks. Their observations are a series of values with precise spatio-temporal coordinates that we aim to simulate. To do so, we retrieve the support of real-world satellite observations denoted $\Omega = \{\Omega_i = (t_i, lat_i, lon_i), i \in [0 : N]\}$ from the Copernicus sea level product (CMEMS, 2021). Using Ω and the ground truth data \mathbf{X}^{ssh} we simulate SSH observations \mathbf{Y}^{ssh} as the trilinear interpolation of the simulated field on each point of the support. We add an instrumental error $\epsilon \sim \mathcal{N}(0, \sigma)$ with $\sigma = 1.9$ cm, which is the distribution used in the Ocean data challenge 2020 (CLS/MEOM, 2020). The SSH observations \mathbf{Y}^{ssh} is defined as following:

$$\mathbf{Y}^{\text{ssh}} = \mathcal{H}^{\text{ssh}}(\mathbf{X}^{\text{ssh}}, \Omega) + \epsilon \quad (3)$$

where \mathcal{H}^{ssh} is the trilinear interpolation operator of the ground truth \mathbf{X}^{ssh} on the support Ω . An example of these simulated along-track measurements is presented on the first row of Figure 1. For the neural network input observations, we regrid these data to a daily 128×128 image. We set the pixel value with no simulated satellite observation to zero, and we average the daily measurements of SSH inside each pixel to represent the mean of the daily data from the different satellites (if any). As GLORYS12 reanalysis assimilates along-track SSH data, selecting satellite measurements at the same location as the assimilated data might introduce a bias in our observations. To overcome this issue, we desynchronize the real satellite ground tracks from the one we use to produce SSH observations by introducing a time delay (772 days) between the real L3 satellite observations and

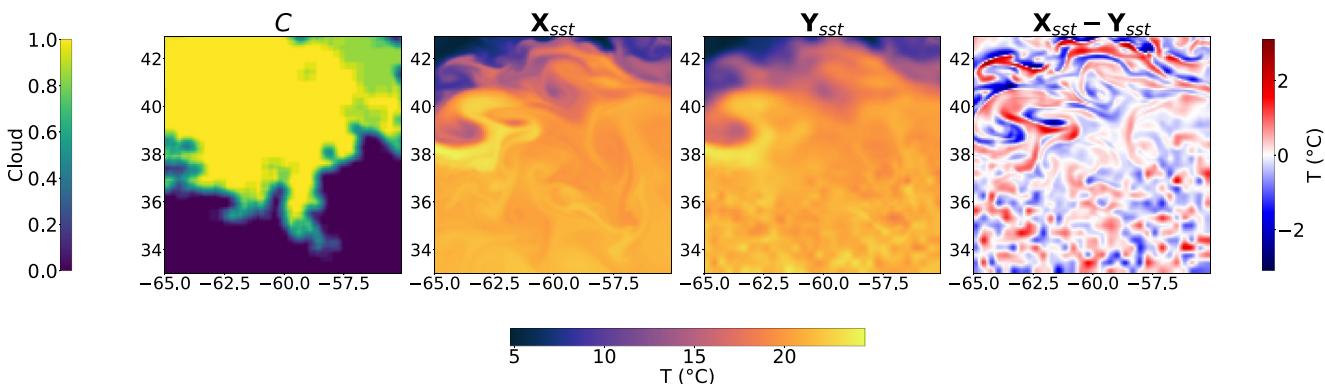


Figure 2. Images of our cloud cover, the ground truth SST from GLORYS12, the noised SST, and the difference.

the simulation. It ensures that simulated along-track data is selected randomly, rather than specifically where the model assimilated real-world observations.

2.2.3. SST Simulated Observations

SST remote sensing is based on direct infrared imaging, leading to wider measurement swaths but making the data sensitive to cloud cover. The so-called L3 satellite products have much higher data coverage, but no observation is possible when clouds are present. To fill the gaps, the L3 products from several satellites are merged and interpolated to form the fully gridded image, using complementary microwave satellite sensors (which produce lower resolution data but are less sensitive to clouds), and in situ measurements (Chin et al., 2017; Donlon et al., 2012). This results in various resolutions in the same product, where high-resolution structures are artificially smoothed when the cloud cover (C) is too thick.

We simulate the SST observation operator \mathcal{H}^{sst} as follows:

$$\mathbf{Y}^{sst} = \mathcal{H}^{sst}(\mathbf{X}^{sst}, C) = (1 - C) \odot (\mathbf{X}^{sst} + \varepsilon) + C \odot \mathcal{G}_{\sigma_t, \sigma_x} \star (\mathbf{X}^{sst} + \varepsilon) \quad (4)$$

where \odot is the element-wise product, \star the convolution product, and ε is a white Gaussian noise image of size 32×32 linearly upsampled to a 128×128 image. We also use a spatio-temporal Gaussian filter, $\mathcal{G}_{\sigma_t, \sigma_x}$ with $\sigma_t = 1.23$ days and $\sigma_x \approx 16$ (km) to simulate the smoothing of the interpolation performed by satellite products. To compute a realistic cloud cover C , we use 2 years of data from an NRT L3 product (CMEMS, 2023), which we periodically replicate to match the length of our data set. We then linearly interpolate the cloud cover to our spatial resolution, and perform an average filter with a kernel size approximately equal to 43 (km). This step is essential, as applying a binary mask results in patches at the frontiers between cloud-free and cloudy regions. Our SST observations thus present a spatially and temporally correlated noise, with different resolutions depending on cloud cover. In the end, \mathcal{H}^{sst} adds a noise with RMSE of 0.48°C where the SST standard deviation of the ground truth is 4.96°C , which we present in Figure 2. This observation operator is different from real-world degradations but produces an image with an in-equal noise resolution similar to the errors present in the L4 SST products. Also, as SST presents strong annual variations that should be removed, we deseasonalize it. For each SST image, we subtract the mean image calculated for the corresponding day across the data set. This is known to improve machine learning time-series prediction (Ahmed et al., 2010), and in our case, it produces better reconstructions as shown in Appendix A3.

2.3. Satellite Observations

To constitute a data set of real-world observations, we propose the L3 SSH product that we used to recover realistic satellite ground tracks (CMEMS, 2021). These data are the inputs used in the DUACS optimal interpolation process and are available from the years 1993–2023. For the L4 SST product, we use the Multiscale Ultrahigh Resolution (MUR) SST (NASA/JPL, 2019). MUR SST is produced through an optimal interpolation of infrared, microwave, and in situ measurements (Chin et al., 2017). Its resolution is very high (0.01°), so we linearly interpolate the data to our resolution (0.078°), and are available from 2002/05/31 to the present. We select satellite

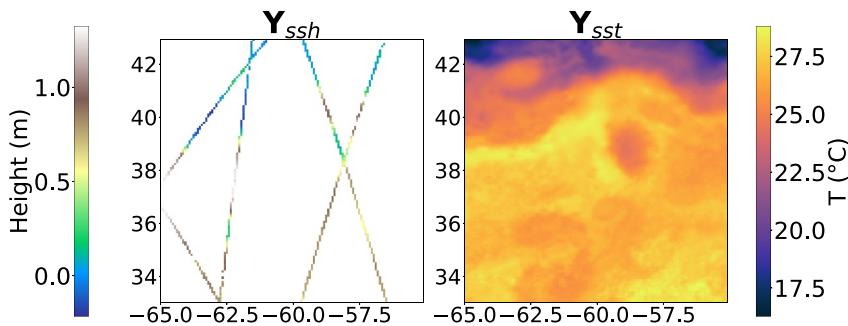


Figure 3. Images of satellite observations of the SSH and the SST, respectively.

observations from 2002/06/01 to 2022/02/09 for a total of 7,194 days which is the same number of timesteps that our OSSE. We also select the same geographical area between 33° to 43° North and -65° to -55° East. The two data are presented in Figure 3.

3. Proposed Interpolation Method

3.1. Learning the Interpolation

The observation operator \mathcal{H}^{ssh} previously described can be seen as a forward operator that we aim to inverse. In the past years, deep neural networks, especially convolutional neural networks, have proven their ability to solve ill-posed image inverse problems (McCann et al., 2017) and more specifically inpainting problems (Jam et al., 2021; Qin et al., 2021). A neural network f_{θ} is trained on a database to estimate the true state from observations $f_{\theta}(y) = \hat{x}$. Learning this inversion operator thus requires (y, x) pairs (supervised) or only y (unsupervised) (Ongie et al., 2020).

We chose to perform the interpolation on a temporal window of 21 days; the input is thus a tensor of 21 images of SSH, with or without SST images, and the output is the 21 corresponding days of SSH only. The neural network estimates the true state from observations, $\hat{\mathbf{X}}^{ssh} = f_{\theta}(\mathbf{Y})$, where $\mathbf{Y} = \mathbf{Y}^{ssh}$ for a SSH-only interpolation, and $\mathbf{Y} = (\mathbf{Y}^{ssh}, \mathbf{Y}^{sst})$ if the network uses SST. The length of the time window is discussed in Section 4.1, and training losses of the network in Section 3.3.

3.2. Architecture

Convolutional neural networks, one of the most used deep learning methods in image tasks, learn convolution operations able to identify features over space and/or time. These networks have been used for multiple tasks in geosciences, from forecasting (Che et al., 2022) to interpolation (Fablet et al., 2021; Manucharyan et al., 2020; S. A. Martin et al., 2023; Archambault et al., 2023), and from eddy detection (Moschos et al., 2020) to super-resolution (Nardelli et al., 2022; Thiria et al., 2023), to name a few. Over time, the machine learning community introduced various ways to organize these convolution operations, each one presenting distinct advantages. Residual layers learn small modifications between their input and output, making neural networks easier to train (He et al., 2016). Attention layers ponder their inputs by a factor between zero and one. This allows subsequent layers to focus on important features while neglecting irrelevant ones, which makes it well-suited to extracting information from contextual variables. It is widely used in many computer vision tasks (Guo et al., 2021) and can be transposed to geoscience applications such as (Che et al., 2022). An encoder-decoder architecture progressively compresses and decompresses the input data, identifying structures at different resolutions.

In this study, we compare different learning techniques on a fixed architecture: an attention-based encoder-decoder (ABED) presented in Figure 4. This neural network benefits from the layers described above. The overall structure of our neural network is inspired by Che et al. (2022), who introduced a residual U-Net with attention layers for rain nowcasting. We removed U-Net residual connections that were not suited for the interpolation task and changed the attention and the upsampling blocks. The encoder starts with a batch normalization and a 3D convolution (in time and space) followed by two downsampling blocks that divide spatial dimensions by 2 (see Figure 4). The decoder is composed of residual attention blocks followed by upsampling blocks.

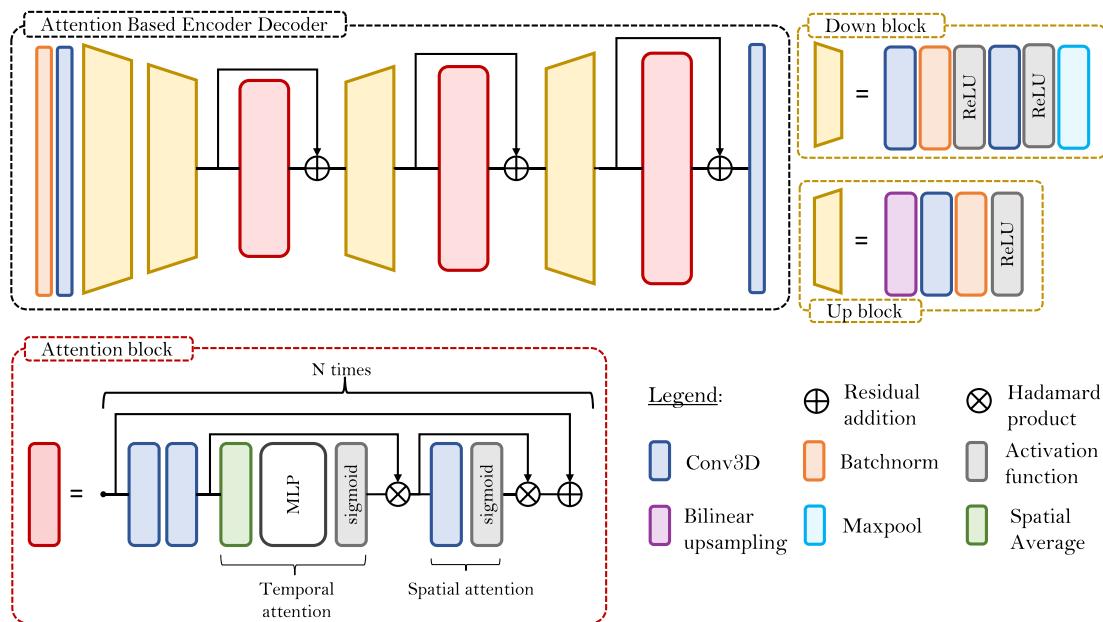


Figure 4. The architecture of the proposed Attention-Based Encoder Decoder (ABED) neural network. It is designed to take a time series of 21 images of SSH, with or without a time series of SST. The encoder divides the spatial dimensions of the images by 4 through 2 “down-block.” Then, a 3D attention layer block is used to highlight relevant information in the images, followed by a residual connection. Finally, a decoding block upsamples the images, and attention and decoding blocks are repeated to get back to the initial image size.

Hereafter, we describe our attention block, which consists of two essential steps: temporal and spatial attention modules. Our approach builds upon the Convolutional Block Attention Module (CBAM) principle introduced by Woo et al. (2018), which successively performs channel and spatial attention. We extend this idea by incorporating temporal information in the channel attention mechanism. To do so, we first compute the spatial average of each channel and instant, resulting in a tensor of size $C \times T$ where C is the channel number and T is the time series length. Subsequently, we apply two one-dimensional convolutional layers with a kernel of size 1, followed by a sigmoid activation function to estimate the attention weights. This corresponds to a 2-layer perceptron shared by every time step, which is different from the CBAM, as it includes the temporal information in the channel attention. These weights are then multiplied to each timestep of every channel, enabling the network to highlight salient features and suppress irrelevant information. After performing temporal attention, we proceed with spatial attention. This step involves utilizing a 3-dimensional convolutional operation, where the kernel size's temporal length matches the time series's length. As a result, the entire time series is aggregated into a single 2D image, which serves as the basis for deriving spatial attention. A residual skip connection is then applied, and the described block is repeated 4, 2, and 1 time for the first, second, and last block, respectively. For further details about our implementation, we provide the PyTorch implementation of our network in <https://gitlab.lip6.fr/archambault/james2024>.

3.3. Loss and Regularization

We propose to compare two main strategies to train the neural network. Thanks to the OSSE previously described, we have access to the ground truth, which we can use to learn the interpolation in a classic supervised fashion. However, it is also possible to train directly on observations by applying the \mathcal{H}^{ssh} on the generated map $\hat{\mathbf{X}}^{ssh}$ before computing the loss (see Equations 5–7). Filoche et al. (2022) performed the interpolation with SSH observations only, and, using the same principle, Archambault et al. (2023) showed that it was possible to estimate SSH images starting from SST only and constraining on SSH observations. Both these methods are fitted on one (or a small number) example and must be refitted to be applied to unseen data. Using a larger real-world satellite data set, S. A. Martin et al. (2023) trained a neural network directly from observations by constraining it on independent satellite observations that were not given in the input. However, the lack of ground truth reference makes it harder to compare the different reconstructions, especially regarding detected eddies and structures. We propose to train neural networks using the 3 following losses:

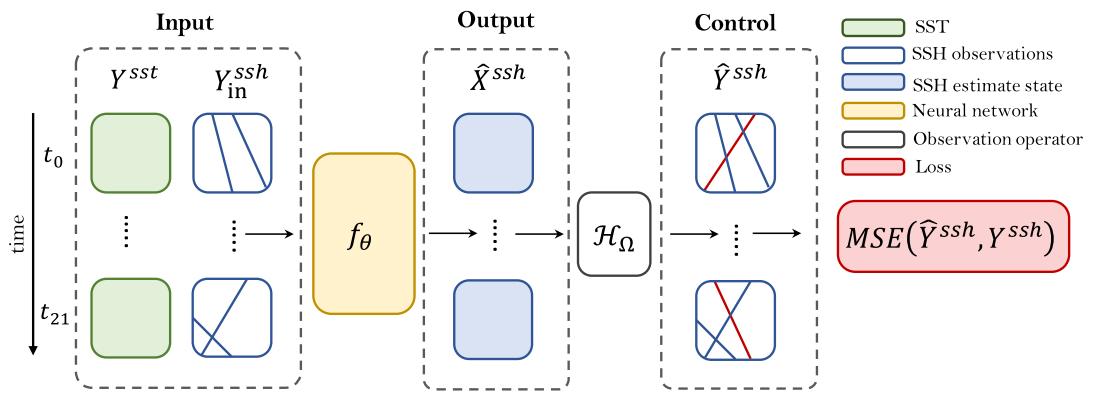


Figure 5. Computational graph of the proposed unsupervised interpolation method. The neural network input is a 21-day time series of SSH satellite observations, excluding data from a single satellite, and optionally includes SST measurements. The network estimates a time series of SSH field states, upon which the observation operator is subsequently applied in order to deduce $\hat{\mathbf{Y}}^{ssh}$. Finally, the Mean Squared Error between the $\hat{\mathbf{Y}}^{ssh}$ and \mathbf{Y}^{ssh} is used to control the network.

- The MSE using ground truth:

$$\mathcal{L}_{\text{sup}}(\mathbf{X}^{ssh}, \hat{\mathbf{X}}^{ssh}) = \frac{1}{T \times H \times W} \sum_{t,x,y} (\mathbf{X}_{t,x,y}^{ssh} - \hat{\mathbf{X}}_{t,x,y}^{ssh})^2 \quad (5)$$

- The MSE using only observations:

$$\mathcal{L}_{\text{unsup}}(\mathbf{Y}^{ssh}, \hat{\mathbf{X}}^{ssh}) = \frac{1}{N} \sum_i (\mathbf{Y}_i^{ssh} - \mathcal{H}^{ssh}(\hat{\mathbf{X}}^{ssh})_i)^2 \quad (6)$$

- The MSE using only observations and the regularization introduced by S. A. Martin et al. (2023):

$$\begin{aligned} \mathcal{L}_{\text{unsup_reg}}(\mathbf{Y}^{ssh}, \hat{\mathbf{X}}^{ssh}) &= \mathcal{L}_{\text{unsup}}(\mathbf{Y}^{ssh}, \hat{\mathbf{X}}^{ssh}) + \lambda_1 \frac{1}{N_1} \sum_i \left(\frac{\partial}{\partial s} \mathbf{Y}_i^{ssh} - \frac{\partial}{\partial s} \mathcal{H}^{ssh}(\hat{\mathbf{X}}^{ssh})_i \right)^2 \\ &\quad + \lambda_2 \frac{1}{N_2} \sum_i \left(\frac{\partial^2}{\partial s^2} \mathbf{Y}_i^{ssh} - \frac{\partial^2}{\partial s^2} \mathcal{H}^{ssh}(\hat{\mathbf{X}}^{ssh})_i \right)^2 \end{aligned} \quad (7)$$

where $\frac{\partial}{\partial s}$ is the along-track derivation of the SSH approximated by its rate of change (see Appendix A1). T is the temporal length of the time series (here 21), H and W the spatial dimensions of the images (here both equals 128), and N, N_1, N_2 , the number of satellite measurements of SSH, SSH first, and SSH second spatial derivative along satellite tracks, respectively. We take $\lambda_1 = \lambda_2 = 0.05$ the regularization coefficients, the same values used by S. A. Martin et al. (2023).

The losses $\mathcal{L}_{\text{unsup}}$ and $\mathcal{L}_{\text{unsup_reg}}$ apply the observation operator \mathcal{H}^{ssh} , before computing the MSE, which allows the training in a framework where only observations are available. Thus, from an interpolation point of view, the inversion methods that use these losses are unsupervised as they can be trained without any ground truth image. However, if we constrain the network on the same observations that were given in input, an over-fitting of along tracks will occur with no guarantee of generalization. To avoid this problem, S. A. Martin et al. (2023) constrained their network on the observations of one satellite that were withdrawn from the input. Similarly, we remove the data of one satellite from the inputs but we calculate the loss function on all satellite observations (the ones given and the ones left aside). In doing so, the network must generalize outside the along-track measurement that was given as input. In Figure 5, we call \mathbf{Y}_{in}^{ssh} the input observations and present an unsupervised inversion computational graph.

3.4. Training Details

3.4.1. Train, Validation, Test Split

We partitioned the OSSE data set into three subsets: training, validation, and test data. We used the year 2017 exclusively to test our reconstructions (every analysis conducted in the following was performed on this data). We validate our methods on three distinct time intervals: (a) from 2002/07/14 to 2003/07/28, (b) from 2008/01/05 to 2009/01/18, and (c) from 2013/06/28 to 2014/07/13. The remaining data was used for training, but leaving a 15-day period to prevent data leakage.

3.4.2. Normalization

We normalize the artificial network's input and output by subtracting the mean and dividing by the standard deviation. The normalization parameters are computed only on the neural network inputs, SST, or along-track data. Specifically, we first perform this normalization for images related to SSH along-track measurements and subsequently replace any missing values with zeros. We normalize the neural network SSH outputs with the statistics computed on the input observations (so that the method remains applicable in an unsupervised setting). When training with the regularized loss of Equation 7, we also normalize the data from the first and second SSH along-tracks derivative.

3.4.3. Training Hyperparameters

We train every method using an ADAM optimizer (Kingma & Ba, 2017) with a learning rate starting at $5 \cdot 10^{-5}$ and a decay of 0.99. We perform an early stopping with a patience of 8 epochs. For the supervised training, the stopping criteria is the RMSE of the reconstruction on the fully gridded domain on the validation data, but in the unsupervised setting, we compute this RMSE on left-aside along-track measurements. Doing so, the stopping strategy is still compliant with a situation where no ground truth is accessible.

3.4.4. Ensemble

As neural network optimization is sensitive to its weight initialization, we train 3 networks for every setting. The so-called "Ensemble" estimation is the average SSH map of the 3 networks. An ensemble estimation helps stabilize performances and enhances the reconstruction (Hinton & Dean, 2015). In the following, we call "Ensemble score" the score of the previously mentioned ensemble estimation and "Mean score" the average of the score of each network taken independently.

4. Results

In Sections 4.1 and 4.2, we compare the different training methods on our OSSE to highlight the drawbacks of unsupervised learning and the advantages of SST. In Section 4.3, we assess the similarity of our OSSE and the previously existing one, the Ocean Data Challenge 2020. In Section 4.4, we build upon the conclusions drawn in previous sections to present a transfer learning method from our OSSE to observations.

4.1. SSH Reconstruction and Quality of Derived Geostrophic Currents

We compare the fields estimated by the networks trained using the 3 losses \mathcal{L}_{sup} , $\mathcal{L}_{\text{unsup}}$ and $\mathcal{L}_{\text{unsup_reg}}$, with 3 different sets of input data: only SSH tracks, SSH and the noised SST (denoted nSST), and noise-free SST (denoted SST). The noise-free SST provides an upper-bound performance of the neural network in the case of a perfect physical link between SSH and SST. We give the RMSE of the SSH estimates fields on the test set in Table 1, and the RMSE on the velocity fields in Table 2. Systematically, the ensemble reconstruction has a lower RMSE than the mean performance, which is usual in machine learning, as individual member errors are compensated by others. Comparing the ensemble scores, we observe that the supervised loss function outperforms the unsupervised framework in every data scenario. Specifically, in the SSH + SST scenario, the supervised loss decreases the ensemble RMSE of $\mathcal{L}_{\text{unsup}}$ by 17%, and 9% without SST. Also, adding SST as an additional input to the network generally improves performance compared to using SSH alone. This improvement is observed across all three loss functions, as the error values decrease for SSH + nSST compared to SSH. For instance, the SSH-only ensemble RMSE is decreased by 31% and 20% for SST and nSST, respectively, with \mathcal{L}_{sup} . The regularization

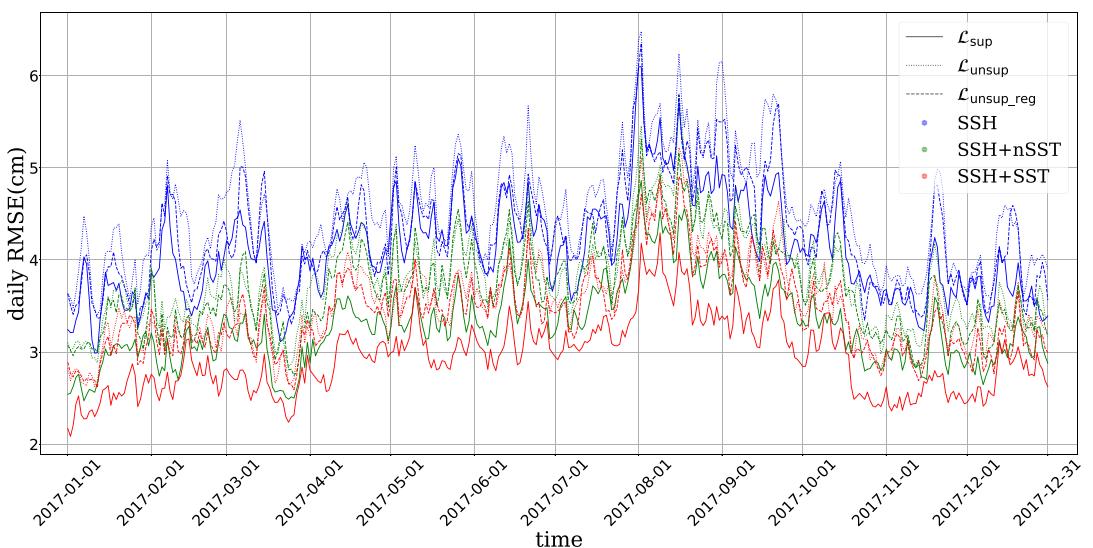


Figure 6. RMSE of the different reconstructions during the test year (2017).

where P_i is the point of the grid where we compute the LNAM, P_j is a neighbor point of the grid, $\overrightarrow{PP_j}$ is the position vector from P_i to P_j and \vec{V}_j is the velocity vector in P_j . Thus, the unnormalized angular momentum L_i is computed through a sum of cross products and is bounded by BL_i , so that if P_i is the center of an axisymmetric cyclone (resp anticyclone), $\text{LNAM}(P_i)$ will be equal to 1 (resp -1). Also, if the circulation field is hyperbolic and not an ellipsoid, S_i will reach large values, and $\text{LNAM}(P_i)$ will be close to 0. All sum is computed on a local neighborhood of P_i , which is a hyperparameter of the method (typically a square centered in P_i). In our case, we used the default parameters where the square has a length of $2\Delta x$, with Δx being the grid resolution (≈ 9 km).

AMEDA finds potential eddy centers by searching for the local extrema of the LNAM field, more precisely by taking the points P_i where $|\text{LNAM}(P_i)| > 0.7$. The characteristic contour of an eddy is then defined as the closed streamline of maximum velocity which does not include another eddy center. We perform the AMEDA algorithm on the geostrophic velocity field of our estimation and on the ground truth currents. We then look for the eddies that are both present in the ground truth and in our estimation. An eddy is said to be detected if the distance

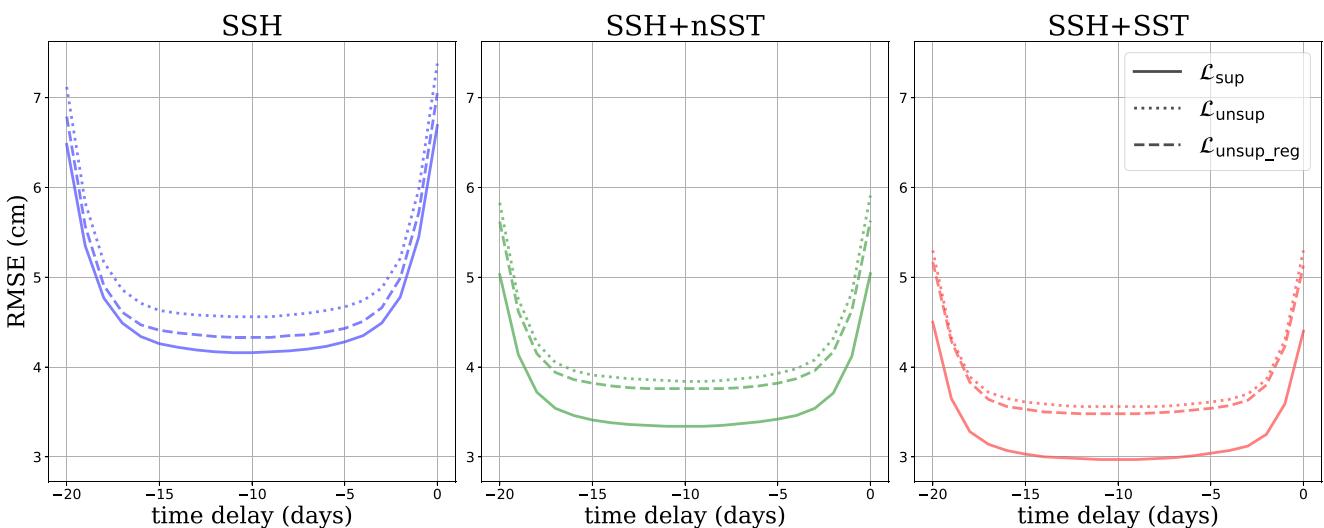


Figure 7. RMSE of the different reconstructions along the time window. The errors at a time delay of -20 correspond to an anti-causal scheme (knowing only future observations) whereas $\text{timedelay} = 0$ corresponds to a causal scheme (knowing no future observations). Knowing both past and future observations leads to the optimal reconstruction at $\text{timedelay} = -10$.

Table 3
Scores of the AMEDA Eddy Detection Performed on the Ensemble Estimation of ABED Interpolation

Loss	SSH			SSH + nSST			SSH + SST		
	F_1	Recall	Precision	F_1	Recall	Precision	F_1	Recall	Precision
\mathcal{L}_{sup}	0.699	0.607	0.825	0.735	0.657	0.833	0.762	0.705	0.83
$\mathcal{L}_{\text{unsup}}$	0.692	0.634	0.76	0.715	0.665	0.772	0.731	0.694	0.771
$\mathcal{L}_{\text{unsup_reg}}$	0.684	0.581	0.831	0.704	0.608	0.835	0.713	0.622	0.836

Note. The considered scores are the precision, the recall, and the F_1 score. The best score is in bold.

between its barycenter and the reference one is smaller than the average of the mean radius of the two characteristic contours. This definition allows “multiple” detection (i.e., colocalization with several eddies). Therefore, we exclude eddies that include more than one candidate in the ground truth. For further details about the AMEDA algorithm, we refer the reader to Vu et al. (2018).

4.2.3. Eddy Detection Performances

We present the detection scores of the different reconstruction methods, with three data scenarios and three losses. We take the ensemble SSH estimation of the neural networks and perform the AMEDA algorithm on the velocity field derived through the geostrophic approximation (see Equation 1).

In Table 3 we present the F_1 score, the recall, and the precision of the methods. The recall tells us the proportion of actual positive instances that were correctly identified by the detection (a recall of 1 means that all ground truth eddies were detected). The precision gauges our trust in the detected eddies (a precision of 1 means that all eddies in the simulation were also present in the ground truth). To aggregate recall and precision, we use the F_1 score, which is the harmonic mean of recall and precision. A value of 1 means a perfect detection: all ground truth eddies were detected, and the estimation produced no false positives.

4.2.3.1. Data Comparison

As expected, no matter which loss we consider, the noise-free temperature detection method outperforms the two other scenarios with higher F_1 scores. Even the noisy SST provides important information for eddy reconstruction, as the SSH-only method yields lower results than the two other scenarios. We also see that for each loss, the precision scores are less impacted by the input data than the recall is. This means that the SSH-only scenario does not produce a lot more false detection than the SST methods but misses much more structures.

4.2.3.2. Loss Comparison

On the other hand, the loss function used to perform the inversion substantially impacts precision and recall. The regularization of the unsupervised loss brings the detection precision to the level of the supervised method (even higher for the SSH-only and SSH + SST) but also reduces the recall of all methods compared to their unregularized version. In other words, the regularization prevents the neural network from generating false eddies and from retrieving some structures, which leads to lower F_1 scores.

4.2.3.3. Visual Comparison

We plot in Figure 8 the SSH maps and eddies detected by AMEDA, and in Figure 9 the relative vorticity ξ computed from geostrophic currents (see Equation 1) as follows:

$$\xi = \frac{\partial v_{\text{geo}}}{\partial x} - \frac{\partial u_{\text{geo}}}{\partial y} \quad (9)$$

Relative vorticity is an important quantity in the analysis of surface circulation as it highlights areas of important direction change of the stream. ξ is positive in counterclockwise spin and negative in clockwise spin. In the presented figures, we normalize relative vorticity fields by the Coriolis factor f . Figures 8 and 9 illustrate an example of the conclusions established in Table 3: the SSH-only reconstruction shows fewer

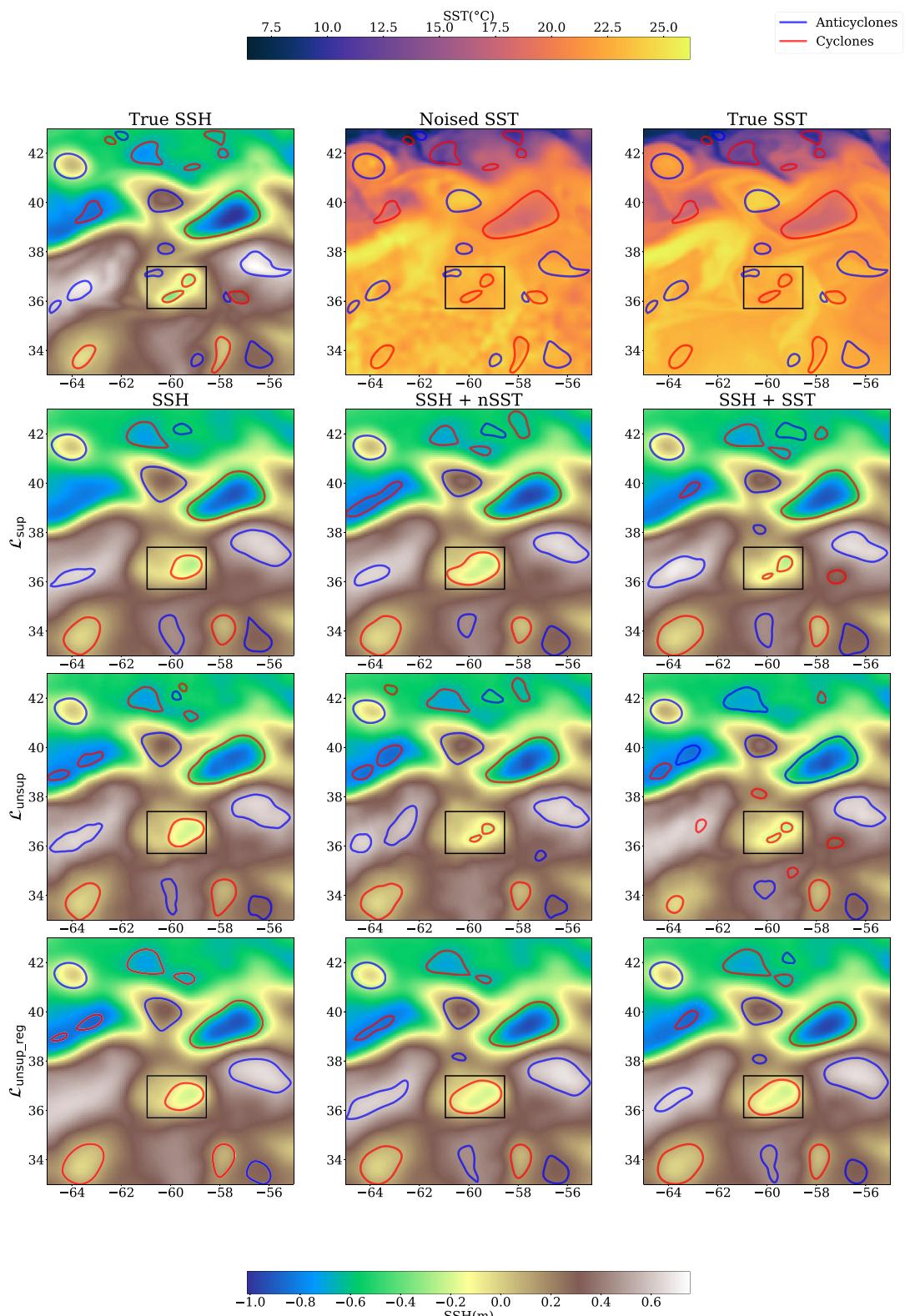


Figure 8. SSH maps and detected eddies the 1st June 2017 on our OSSE. The first line presents the True SSH, the noised SST, and the True SST, on which we plot the eddies detected on the True SSH. The second, third, and last lines present respectively the inversion using \mathcal{L}_{sup} , $\mathcal{L}_{\text{unsup}}$, and $\mathcal{L}_{\text{unsup,reg}}$. The first, second, and last columns present the maps using the SSH-only, SSH + nSST, SSH + SST data, respectively. Each SSH map is the ensemble reconstruction of 3 networks with their associated eddies.

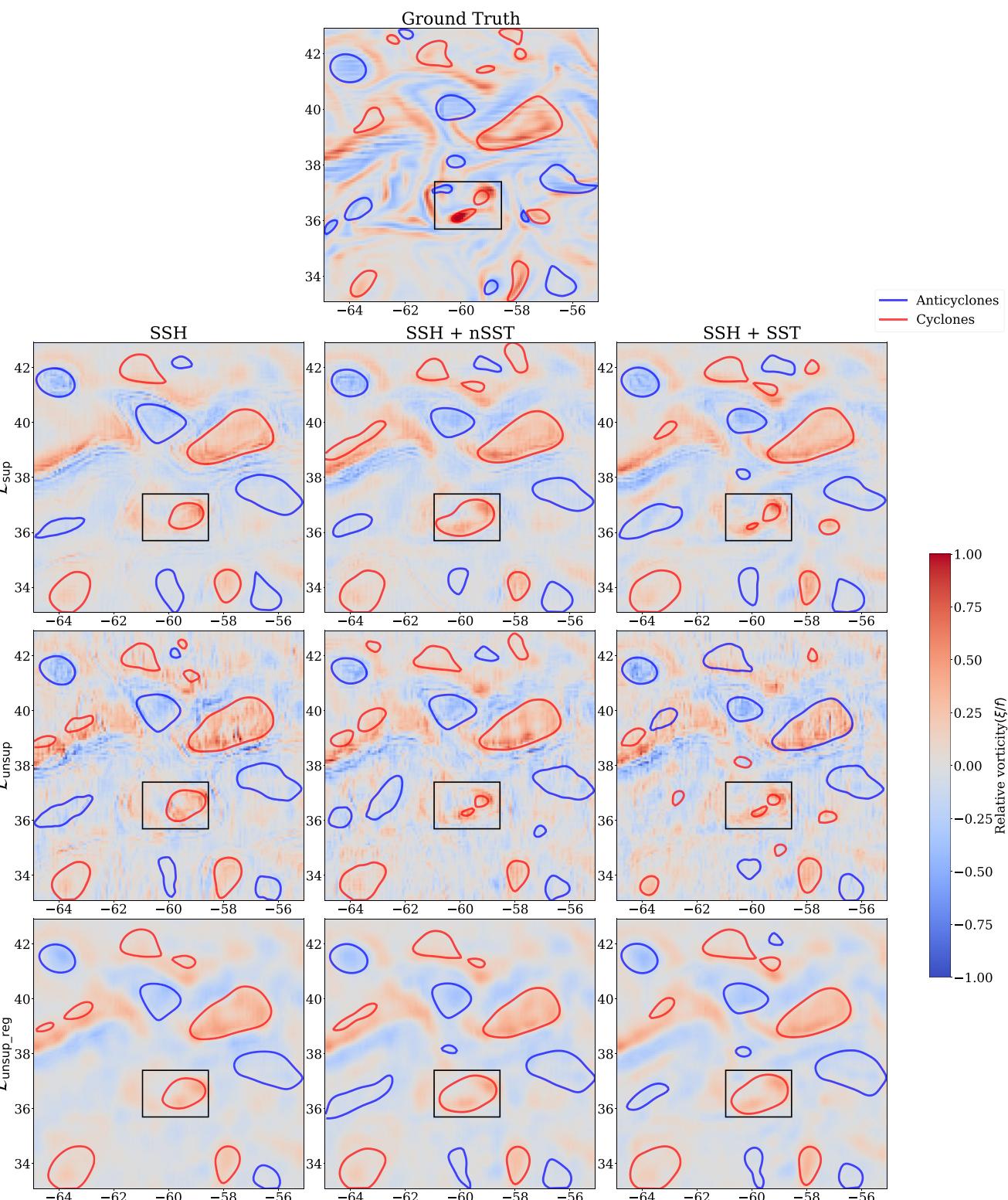


Figure 9. Relative vorticity (normalized by the Coriolis factor) and detected eddies on the 1st June 2017 in our OSSE. The first line presents the True relative vorticity. The second, third, and last lines present the neural networks trained with \mathcal{L}_{sup} , $\mathcal{L}_{\text{unsup}}$, and $\mathcal{L}_{\text{unsup_reg}}$. The first, second, and last columns present the SSH-only, SSH + nSST, and SSH + SST interpolations. Each relative vorticity map is computed from the ensemble SSH estimation of the 3 networks.

eddies than the ones using SST, and aggregates small eddies into larger ones (see highlighted eddies). We also see the effect of regularization, especially in the relative vorticity fields, which are a lot smoother than the ones in the supervised and regularized inversion. This smoothing effect results in a reduced number of detected eddies, as illustrated by the two highlighted eddies that are detected separately when SST is used without regularization.

4.2.4. Physical Properties of Detected Eddies

To further investigate the performance of the eddy detection methods, we analyze the detection outcomes based on the physical characteristics of the eddies. For instance, smaller eddies tend to have shorter lifespans, making them more challenging to detect due to their decreased likelihood of being observed by satellites. Conversely, high-speed eddies are derived from important sea surface height (SSH) variations, thus exhibiting a strong signature in the generated mapping. Figure 10 shows the detection performances as a function of some key parameters such as maximum radius, lifetime, or maximum velocity along the final closed current line.

As anticipated, using SST and nSST data contributes to the detection of eddies, as indicated by the higher F_1 scores achieved in every loss scenario. However, small and short-lived eddies are less frequently detected, resulting in lower recall scores. Specifically, only 17% of the eddies with a radius below 15 km are successfully detected in the best scenario. Nonetheless, except for the unregularized loss function, the precision scores for the detected eddies remain high, even for small and short-lived ones. This observation confirms the previously observed phenomenon where the regularization employed in the inversion process prevents the network from generating false eddy detections but also stops it from capturing a significant portion of the actual eddies. This regularization behavior is expected, as forcing a smoothness constraint on the SSH gradient field leads to denying some of the small structures.

We also want to assess the model's accuracy to estimate the eddies' physical properties. To this end, we focus on the eddies that were successfully detected by all the methods (3,534 eddies out of the 7,908 eddies in the ground truth) and compare the physical parameters of the estimated eddies to their values in the corresponding true eddy. We compute the RMSE and bias of the following parameters: maximum radius and velocity of the characteristic contour of the eddies. Once again, Tables 4 and 5 show that SST helps to estimate eddies radius, and velocity. Nonetheless, there is a bias of radius and velocity: the size of the eddy is statistically overestimated compared to its ground truth, while its speed is systematically underestimated. This is particularly true for the regularized unsupervised loss because of its smoothness constraint, with a velocity bias accountable for half of the RMSE.

4.3. Comparison With State-Of-The-Art Methods on a NATL60 OSSE

Comparing various SSH interpolation methods requires a common benchmark and metrics. The Ocean Data Challenge 2020 (CLS/MEOM, 2020) provides an OSSE similar to the one described in Section 2, the state-of-the-art estimations and metrics. The included data are the ground truth SSH, nadir-pointing observations, and a simulation of the SWOT (Surface Water and Ocean Topography) observations, a new altimetry technology (Gaultier et al., 2016). In this study, we have excluded the SWOT measurements as we do not simulate them in our OSSE and focus on nadir-pointing data. The ground truth is the NATL60 simulation (Ajayi et al., 2019) which uses the same physical model (NEMO 3.6) (Madec et al., 2017) but at finer scales than GLORYS12, and without assimilation. Given that the NATL60 model also outputs SST and ocean currents fields, we retrieved and used these variables, even though they were not included in the official depository of the challenge. The state-of-the-art frameworks presented in this challenge are the following:

- DUACS: the operational linear optimal interpolation leveraging covariance matrix tuned on 25 years of data;
- DYMOST (Ballarotta et al., 2020; Ubelmann et al., 2016) and MIOST (Ardhuin et al., 2020): two variants of the linear optimal interpolation where the Gaussian covariance model is changed for a non-linear quasi-geostrophic dynamic model (for DYMOST) or by a wavelet base (MIOST);
- bfn (Le Guillou et al., 2020): a data assimilation method that performs a back and forward nudging of a quasi-geostrophic model;
- 4Dvarnet (Fablet et al., 2021): a deep learning framework supervised on the Ocean Data Challenge 2020. In this configuration, it only takes SSH observations as input;

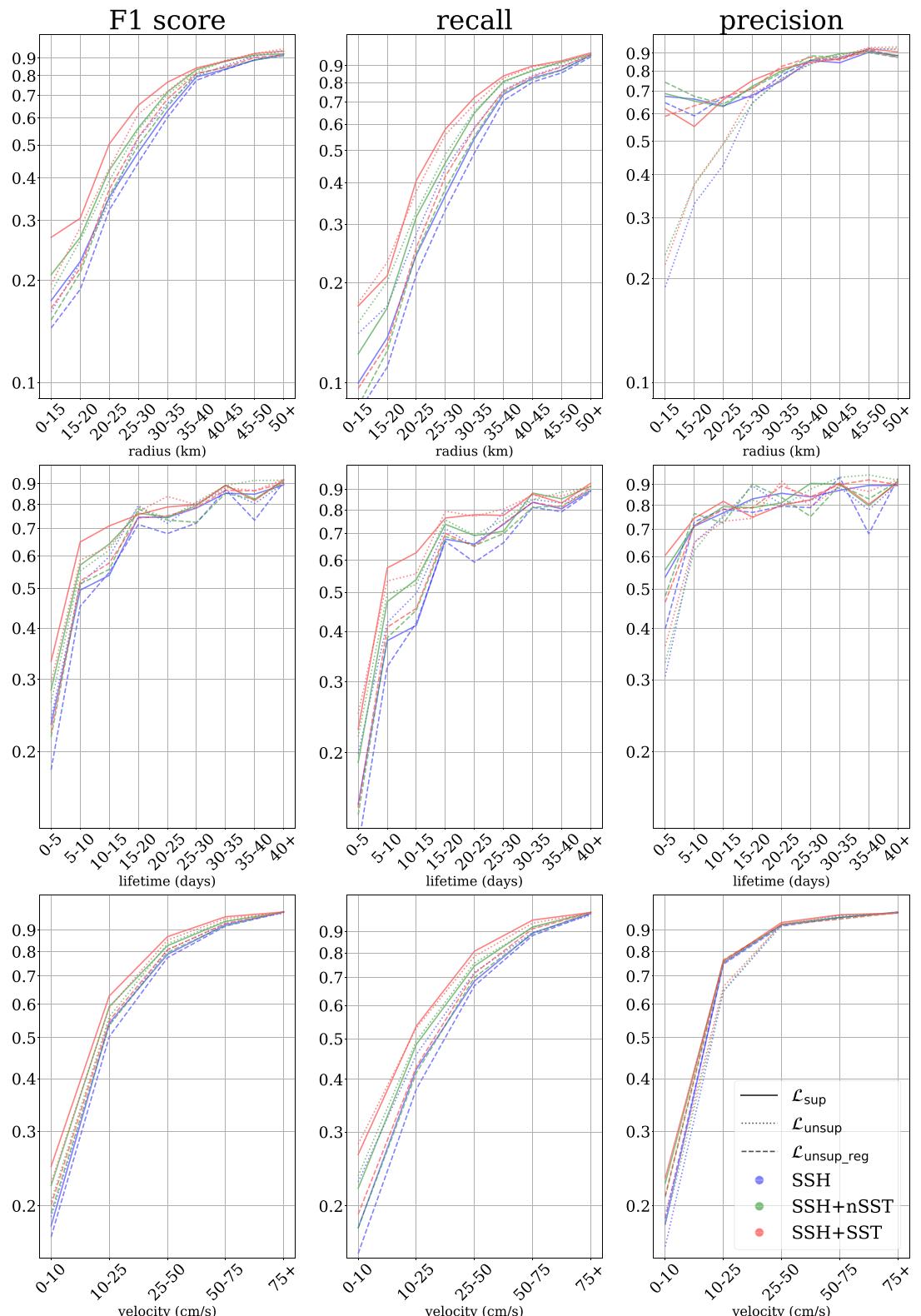


Figure 10. Detection scores of the different methods on eddies separated by radius (first row), lifetime (second row), and maximum velocity (last row). The considered scores are F_1 (first column), recall (second column), and precision (third column). The recall tells the proportion of actual positive instances that were correctly identified, the precision gauges the trust that we can put in the detected eddies, and the F_1 score aggregates these two values.

Table 4

Eddies Maximum Radius RMSE and Bias (km)

Loss	SSH		SSH + nSST		SSH + SST	
	RMSE	Bias	RMSE	Bias	RMSE	Bias
\mathcal{L}_{sup}	16.7	3.6	15.7	4.2	14.7	3.7
$\mathcal{L}_{\text{unsup}}$	16.6	0.9	16.3	1.3	15.5	1.3
$\mathcal{L}_{\text{unsup_reg}}$	16.6	3.5	16.5	4.1	15.7	4.5

Note. The eddy detection is performed on geostrophic currents of the ensemble estimation and the bias is computed from the estimated radius minus ground truth radius. The best score is in bold.

- MUSTI (Archambault et al., 2023): an unsupervised neural network fitting SSH along tracks observations starting from an SST image. The fact that this method must be fitted to new observations, limits its operational use.

This benchmark is not complete as the CONVLSTM interpolations introduced by S. A. Martin et al. (2023) were trained on real satellite observations only, and the 4DVARNET versions using SST were only computed using SSH observations from nadir pointing satellites and SWOT data (Fablet et al., 2023). Still, we are interested in evaluating the reconstructions of our networks, trained on our OSSE, on the Ocean Data Challenge 2020 to show the similarity of the two simulated observation systems. To produce our estimation, we regrid the provided data to our resolution (from 0.016° to 0.078°) using trilinear interpolation. We use the SSH simulated observations of the data challenge and the SST of the corresponding NATL60 simulation. The test period includes 42 days of simulation (between 2012/10/22 and 2012/12/02) as defined in the challenge. As such, the comparison is not fully fair since regridding and not training on the same data might bias the scores obtained. It is still a good way to evaluate the similarity of our OSSE to the Ocean Data Challenge 2020, as our approach obtains comparable performances to the state-of-the-art. Each method is then evaluated using the following metrics, and we sum up the results in Table 6:

- μ and σ_t (in cm), are respectively the RMSE of the SSH and the temporal standard deviation of this RMSE. In the data challenge, these two metrics are normalized by the root mean square of the SSH, but we prefer giving it in centimeters to be coherent with the rest of the work;
- λ_x (in degrees) and λ_t (in days) are two spectral metrics, introduced by Le Guillou et al. (2020). We compute respectively the spatial and temporal power spectrum of the error, λ_x is then the smallest spatial wavelength where the power spectrum of the error is equal to the power spectrum of the signal and λ_t its temporal equivalent. For further information, we refer the reader to (Le Guillou et al., 2020);
- μ_u and μ_v (in cm/s) are the RMSE between the NATL60 currents and the geostrophic currents of the estimation.

We see in the scores a predominance of neural network-based methods (MUSTI, 4DVARNET and ABED) as the importance of the SST in the reconstruction (MUSTI, and ABED). The ABED-SSH networks do not perform as well as 4DVARNET, but better than optimal interpolations (DUACS, DYMOST, MIOST) and BFN. This analysis further supports using SST data in deep learning-based methods for these inverse problems. We can expect around 2 cm of error reduction on the operational interpolation scheme DUACS with our best method (41% of reduction). We also significantly reduce the errors on currents compared to DUACS's, by 5.7 cm/s for u and 5.4 cm/s for v (35% and 34% error reduction).

4.4. Application to Real Satellite Observations

In this section we focus on applying the developed methods to real observations with two objectives in mind: show the utility and realism of our OSSE compared to the pre-existing one, and explore transfer learning strategies. To evaluate our method on a shared benchmark, we use the Ocean Data Challenge 2021 (CLS/MEOM, 2021), which provides one year of real SSH nadir observations and evaluation metrics. All the evaluations presented in this section are computed on the along-track data from the CryoSat-2 satellite left aside in all the benchmarked methods. The comparison is done on the entire 2017 years, which is the year that we left aside from training on our OSSE to avoid data leakage. To be coherent with the area covered by all the methods, the evaluation area is smaller than the one of the OSSE (between 34° to 42° North and -65° to -55° East). These real-world measurements present instrumental errors that produce much higher RMSE scores than the ones computed on the OSSE. Also, as we do not have access to complete SSH maps, the metrics used are μ , σ_t , and λ_x (in km this time). For methods requiring SST information, we use satellite SST from (NASA/JPL, 2019) described in Section 2.3.

Table 5

Eddies Maximum Velocity RMSE and Bias (cm/s)

Loss	SSH		SSH + nSST		SSH + SST	
	RMSE	Bias	RMSE	Bias	RMSE	Bias
\mathcal{L}_{sup}	14.3	-5.3	12.4	-3.1	11.7	-2.0
$\mathcal{L}_{\text{unsup}}$	14.4	-5.7	13.4	-3.3	12.5	-2.9
$\mathcal{L}_{\text{unsup_reg}}$	15.2	-8.3	13.9	-7.4	13.2	-6.5

Note. The best score is in bold.

Table 8

Along-Track SSH RMSE in Centimeters (Mean Score on the Left and Ensemble Score on the Right) of 3 ABED Networks, Computed on 1 Year of Data Provided by the Ocean Data Challenge 2021

Learning method	Input data		
	SSH	SSH + nSST	SSH + SST
Observation	7.07–6.75	6.63–6.27	–
Simulation	6.63–6.35	6.28–6.06	6.89–6.68
Pre-training & Fine-tuning	6.49–6.28	6.02–5.82	6.04–5.84

Note. The training strategies include observation-only training (with satellite SSH and SSH + nSST), simulation-only training (SSH, SSH + nSST, SSH + SST), and fine-tuned networks (SSH, SSH + nSST, SSH + SST). For the Fine-tuned networks, when a network is pre-trained with noise-free SST, it is still fine-tuned with noisy satellite SST. The best score is in bold.

4.4.2.3. Pre-Training on OSSE and Fine-Tuning on Satellite Data

After the supervised pre-training on OSSE data we fine-tune the neural network on satellite data for a few epochs using the unsupervised loss. The fine-tuning is done using a small learning rate of 1.10^{-5} and a decay of 0.9. We use an early stopping with a patience of 8 epochs and we save the best model on the validation set.

We present in Table 8 the RMSE on the Ocean Data Challenge 2021 of 3 ABED networks trained with the previously mentioned methodologies. One of the first conclusions we can draw from these reconstruction scores is the interest of our OSSE in the training process. The networks fitted on the simulation perform better than their equivalent trained with observations only, except for the network trained using noise-free SST. This shows that our SST noise is realistic, as introducing SST noise during pre-training is beneficial for generalization to satellite data. Second, in every data scenario, the pre-trained and fine-tuned networks perform significantly better than their version trained on observation or simulation. In particular, once fine-tuned, the networks pre-trained on nSST and on SST lead to close performance, whereas without fine-tuning, the network trained on noise-free SST produces the worst reconstruction. Given an appropriate fine-tuning strategy, the features learned on noise-free SST that do not apply to satellite data are effectively modified. From this experiment, we conclude that combining supervised training on our OSSE with unsupervised re-fitting on satellite data increases performance, especially if SST is used.

In Table 9, we compare our method to the state-of-the-art interpolation methods provided in the context of the Ocean Data Challenge 2021. The included methods are the same as in Table 6, plus the CONVLTSM-SSH and CONVLTSM-SSH-SST (S. A. Martin et al., 2023). We give ensemble scores of the three pre-trained and fine-tuned ABED networks using only SSH, or SSH and the noised SST. The enhanced scores of ABED-SSH-SST and CONVLTSM-SSH-SST compared to their SSH-only versions emphasize the improvements brought by the SST. ABED, CONVLTSM and 4DVARNET lead to better SSH gridding than optimal interpolation-based methods (DUACS, DYMOS, MIOS) both in terms of RMSE and effective spatial resolution. We also note a significant drop in RMSE score for the BFN method compared to its OSSE reconstruction, which shows that the idealized QG model is less applicable to real-world observations.

In Figure 11, we present the SSH maps on the different reconstruction methods with their associated relative vorticity (see Equation 9). The three first methods (DYMOS, DUACS, MIOS) present smooth vorticity maps as a consequence of the optimal interpolation. All the vorticity maps from neural network-based methods: 4DVARNET,

Table 9
Comparison of the State-Of-The-Art Reconstruction Methods on the Real Satellite Data of the Ocean Data Challenge 2021

Method	SST	Learning	$\mu(\text{cm})$	$\sigma_i(\text{cm})$	$\lambda_x(\text{km})$
DUACS	✗	✗	7.66	2.66	149
DYMOS	✗	✗	6.75	2.00	131
MIOS	✗	✗	6.75	2.00	139
BFN	✗	✗	7.46	2.59	119
4DVARNET	✗	Simulation	6.56	1.84	107
MUSTI	✓	Observation	6.26	1.96	114
CONVLTSM-SSH	✗	Observation	6.82	1.86	114
CONVLTSM-SSH-SST	✓	Observation	6.29	1.60	108
ABED-SSH	✗	Pre-training & Fine-tuning	6.28	1.82	118
ABED-SSH-SST	✓	Pre-training & Fine-tuning	5.82	1.61	108

Note. SST stands for whether or not the reconstruction methods are using SST. ABED-SSH and ABED-SSH-SST stands for the ensemble score of our pre-trained and fine-tuned networks. The best score is in bold.

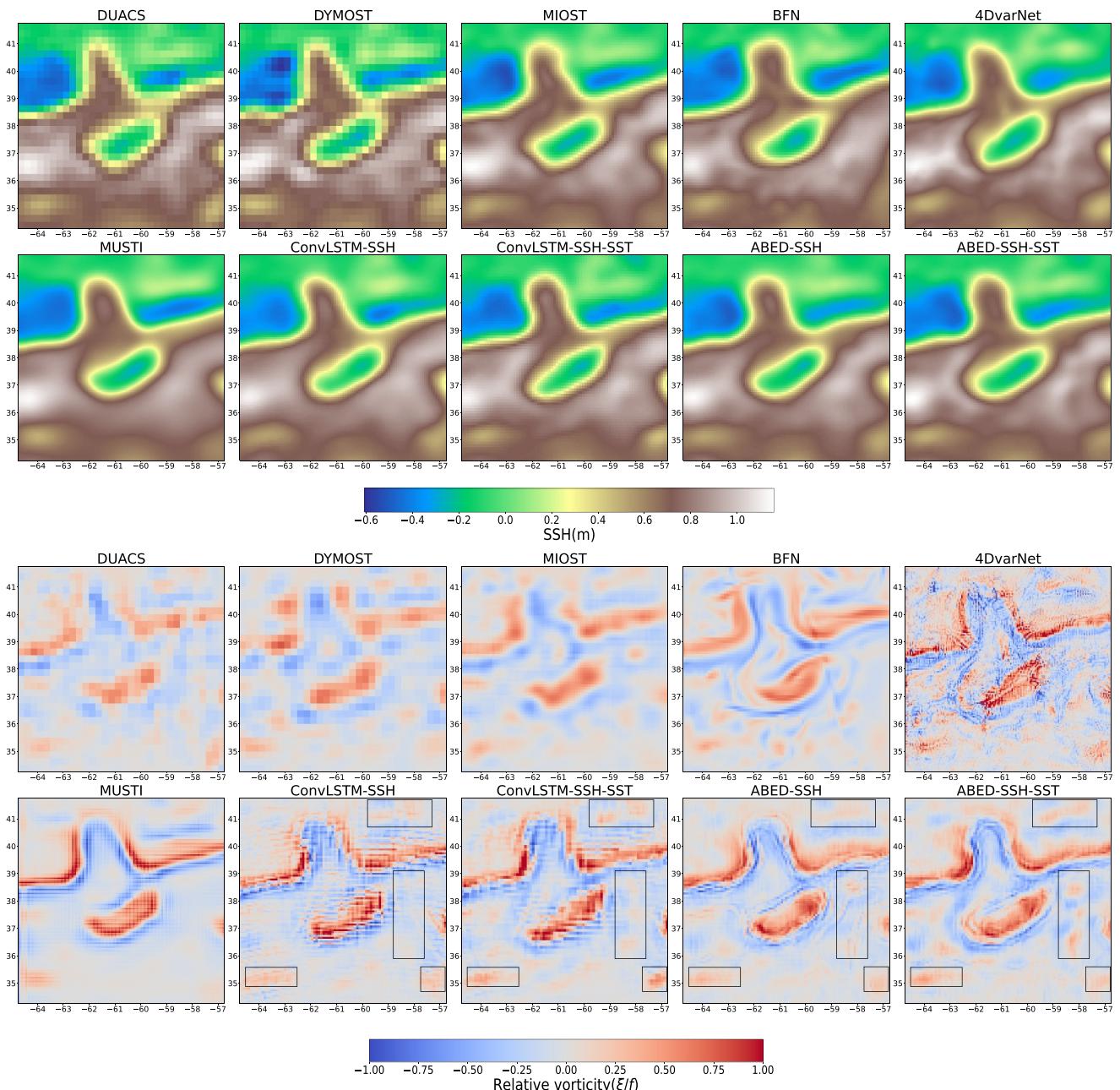


Figure 11. SSH maps and Relative Vorticity maps (normalized by the Coriolis frequency) of the methods from Table 9. The SSH maps are used to compute geostrophic currents from which we derive Relative Vorticity. Due to the different areas covered by the methods, we plot the SSH and RV on a portion of the training area; from 34.25° to 41.75° North and from -64.75° to -56.75° East. On the last four relative vorticity maps, we highlight some regions where small structures are visible in the SST-using interpolations and not visible (or less salient) in their SSH counterparts.

MUSTI, CONVLTSMs, and ABEDs have higher contrast and also some artifacts due to convolution operations. 4Dvarnet in particular, produces very high-frequency variations on which we can see the input satellite path. We suppose this is a consequence of the U-Net's skip connections whereas the other networks have Encoder Decoder architectures, less prompt to produce high-frequency noise. For the last four methods, CONVLTSM-SSH, CONVLTSM-SSH-SST, ABED-SSH, and ABED-SSH-SST, we highlight areas where small structures are visible in the vorticity maps of the SST-using methods but not in their SSH counterparts. The similar shape of the structures between CONVLTSM-SSH-SST and ABED-SSH-SST suggests that they are linked to the use of SST and not the deep learning method.

5. Conclusion and Perspectives

5.1. Summary

In this work, we designed a new OSSE emulating 20 years of satellite observations of SSH and SST while the previously existing OSSE provided only 1 year of simulated SSH observations (CLS/MEOM, 2020). We were able to train an Attention-Based Encoder Decoder using 3 different loss functions (2 of them learning the reconstruction without ground truth), on three different sets of data (SSH only, SSH and noised SST, SSH, and SST). We show a systematic interpolation improvement thanks to the use of SST. Using temperature data (noisy or not), the unsupervised inversion outperforms even the supervised SSH-only neural network (3.86 cm of RMSE for the unsupervised noisy SST against 4.18 cm for the supervised SSH-only method). This shows the importance of contextual information to constrain this inverse problem, even while learning with observation only.

Using AMEDA, an automatic eddy detection algorithm, we were able to identify cyclones and anticyclones in the ground truth and compare them with the eddies detected in the geostrophic approximation of the different mappings. This allows a deeper physical interpretation than the SSH reconstruction alone. We conclude that SST aids in capturing finer structures that might be overlooked by SSH-only methods and that SST-using methods better render the key physical properties of the detected eddies, such as size, speed, or center position. Furthermore, in unsupervised reconstruction, we show that the non-regularized and regularized inversions have close detection scores, but their errors are different. The regularized inversions exhibited lower recall scores, indicating that certain eddies were not detected due to the smoothing effect of the regularization process. However, they demonstrated higher precision scores, implying increased confidence in the successfully detected eddies.

We evaluate ABED trained using the data from our OSSE on the Ocean Data Challenge 2020 and compare it with state-of-the-art interpolation techniques. We show that the utilization of SST led to a substantial improvement of 41% in terms of RMSE for SSH compared to the widely used L4 product from DUACS. Moreover, we observed significant improvements of 34% and 35% for u and v currents, respectively. These findings present promising perspectives for advancing satellite SSH gridding through the application of deep learning methodologies and the fusion of diverse physical information.

Finally, we presented a novel training strategy, using jointly OSSE and real-world satellite observations. We proposed to perform a transfer from the OSSE to the satellite domain by pre-training the neural network on the OSSE and fine-tuning it on a real-world data set in an unsupervised way. Comparing the same network trained following three strategies: on simulation only, on observations only, or the one introduced here, we found that using together simulation and satellite data leads to better performances. Specifically, our transfer method achieves state-of-the-art performances on the Ocean Data Challenge 2021, on which we report an RMSE improvement of 24% compared to DUACS.

5.2. Perspectives

5.2.1. SSH Forecast

This study focused on a delayed time interpolation of the SSH. However, near real-time and forecast data are often useful in many operational applications, such as navigation and meteorology. In future works, we would be interested in extending the output window in the future compared to the input one. In doing so, the neural network would be trained to interpolate and forecast the SSH simultaneously. We would be interested to compare a method doing the two tasks simultaneously to a method doing it successively.

5.2.2. Global Interpolation

Furthermore, many challenges still need to be addressed to get toward a global gridded SSH product. For instance, as the geostrophic equilibrium depends on the Coriolis force surface projection and thus on the latitude considered, we may require a model to be trained on several areas with different latitudes. Also, we can wonder which strategy is more efficient between training a global model or several local models, each one specialized for a range of latitude or geographical area. Closed seas and coastal water also have very different physical interactions and might need to be reconstructed by different methods.

5.2.3. Using Different Input and Output Data

We have demonstrated the benefit of using multi-physical information, specifically SST, to enhance SSH reconstruction through the implementation of a flexible neural network framework. The integration of data from diverse physical sources exhibits promising outcomes, yet conventional model-based methods encounter challenges due to noise and observational difficulties associated with real-world data. In contrast, machine learning opens doors to augment these methods with diverse and abundant data sources. For instance, we employed noisy yet complete SST data in our investigation, but using L3 SST products is also possible. Furthermore, an intriguing prospect arises as to whether Level 4 (L4) and Level 3 (L3) SST products can be effectively combined, thereby potentially yielding even more precise and exhaustive information. Other physical measurements might improve the reconstruction, such as chlorophyll maps that track plankton advected by currents (Kahru et al., 2012).

Appendix A

A1. Along-Track Spatial Derivatives

We calculate the SSH's first and second spatial derivatives along the satellite ground track as described in Equations A1 and A2. Given \mathbf{Y}^{ssh} , the list of SSH measurements from one satellite (sorted in time), we approximate the derivative by the rate of change of the SSH:

$$\frac{\partial}{\partial s} \mathbf{Y}_i^{ssh} \approx \frac{\mathbf{Y}_{i+1}^{ssh} - \mathbf{Y}_i^{ssh}}{\Delta s_i} \quad (\text{A1})$$

$$\frac{\partial^2}{\partial s^2} \mathbf{Y}_i^{ssh} \approx \frac{\frac{\partial}{\partial s} \mathbf{Y}_{i+1}^{ssh} - \frac{\partial}{\partial s} \mathbf{Y}_i^{ssh}}{\Delta s_i} \quad (\text{A2})$$

where \mathbf{Y}_i^{ssh} is the i th measurement of SSH, Δs_i is the ground distance between the SSH measurements, and Δs_i is the ground distance between the two first derivative approximations. The lists of first and second spatial derivatives, $\frac{\partial}{\partial s} \mathbf{Y}^{ssh}$ and $\frac{\partial^2}{\partial s^2} \mathbf{Y}^{ssh}$, are re-centered on new coordinates, corresponding to the dual coordinates of the \mathbf{Y}^{ssh} and $\frac{\partial}{\partial s} \mathbf{Y}^{ssh}$, respectively. We only compute the spatial derivatives from observations coming from the same satellite and only if the measurements are taken with less than 2 seconds of delay. This way we estimate spatial derivatives only where the rate of change is a valid approximation of the derivation.

A2. Impact of the OSSE Temporal Length on Training

Our OSSE data set is composed of 7,194 days, which leads to 5,504 training days once the partition between train, validation, and test sets is made. To evaluate the interest in using more data to constrain the neural network, we train ABED network in the optimal configuration (supervised and using noise-free SST). We compare the scenario where all the samples are seen during training with those where only half, a quarter, or a single year of the data set is used. The validation and test sets remain unchanged, while the training subset is the first consecutive days from the initial training set. Table A1 presents the RMSE of the reconstructions on the test year of our OSSE. The scores of the networks trained with different data set sizes clearly show better reconstruction performance when the size increases.

Table A1

Mean RMSE Score (in cm) of 3 ABED Networks Trained on Our OSSE in a Supervised Manner Using SSH and Noise-Free SST

Size	Full	1/2	1/4	1 year
Number of training samples	5,504	2752	1,376	365
RMSE (cm)	2.97	3.97	4.77	7.76

Note. We compare the situation where the full, half, a quarter, or 1 year of the data set is used.

- Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359. <https://doi.org/10.1109/TKDE.2009.191>
- Qin, Z., Zeng, Q., Zong, Y., & Xu, F. (2021). Image inpainting based on deep learning: A review. *Displays*, 69, 102028. <https://doi.org/10.1016/J.DISPLA.2021.102028>
- Stegner, A., Le Vu, B., Dumas, F., Ghannami, M., Nicolle, A., Durand, C., & Faugere, Y. (2021). Cyclone-anticyclone asymmetry of eddy detection on gridded altimetry product in the Mediterranean Sea. *Journal of Geophysical Research: Oceans*, 126(9), e2021JC017475. <https://doi.org/10.1029/2021JC017475>
- Taburet, G., Sanchez-Roman, A., Ballarotta, M., Pujol, M.-I., Legeais, J.-F., Fournier, F., et al. (2019). DUACS DT2018: 25 years of reprocessed sea level altimetry products. *Ocean Science*, 15(5), 1207–1224. <https://doi.org/10.5194/os-15-1207-2019>
- Thiria, S., Sorror, C., Archambault, T., Charantonis, A., Béréziat, D., Mejia, C., et al. (2023). Downscaling of ocean fields by fusion of heterogeneous observations using deep learning algorithms. *Ocean Modelling*, 182, 102174. <https://doi.org/10.1016/j.ocemod.2023.102174>
- Ubelmann, C., Cornuelle, B., & Fu, L. (2016). Dynamic mapping of along-track ocean altimetry: Method and performance from observing system simulation experiments. *Journal of Atmospheric and Oceanic Technology*, 33(8), 1691–1699. <https://doi.org/10.1175/JTECH-D-15-0163.1>
- Vu, B. L., Stegner, A., & Arsouze, T. (2018). Angular momentum eddy detection and tracking algorithm (AMEDA) and its application to coastal eddy formation. *Journal of Atmospheric and Oceanic Technology*, 35(4), 739–762. <https://doi.org/10.1175/JTECH-D-17-0010.1>
- Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. Computer Vision and Pattern Recognition.
- Zhai, X., Greatbatch, R. J., & Kohlmann, J.-D. (2008). On the seasonal variability of eddy kinetic energy in the Gulf Stream region. *Geophysical Research Letters*, 35(24), L24609. <https://doi.org/10.1029/2008GL036412>