1 Reviewer 1

1.1 General Comments

> Overall, the manuscript is structured well and clearly demonstrates the application of denoising diffusion probabilistic models in the domain of satellite reconstruction, with compelling results when compared to the baseline method, denoted as DINCAE (a method which was previously applied on this task). However, before I can fully recommend the manuscript for publication, there are some shortcomings that have to be addressed. Firstly, there are some implementation details which might indicate potential errors in the algorithm's implementation. It seems that the presented equations pertaining to the diffusion model somewhat deviate from the standard definition, while the changes lack an explanation or motivation. Secondly, the results section, although demonstrating that the proposed method compares favourably to the baseline approach in terms of RMSE, variogram, and quality of reconstruction, could still benefit from some additional comparisons. Furthermore, I do not wholly agree with some of the conclusions reached by the authors with regards to the interpretation of the Talagrand diagram. I provide detailed arguments for each of the issues raised in the following Section.

We thank the reviewer for her/his careful reading of the manuscript. Essentially, we agree with the proposed changes and implement them in the updated manuscript. Unfortunately, there were some typos in the original manuscript. However, these typos only affected the presentation and not the implementation. The source code of the diffusion model has also been made available (https://github.com/gher-uliege/DINDiff.jl). As proposed by the reviewer, we extended the discussion of the results (in particular, computing the Talagrand diagram and the CRPS for the DINCAE method). The interpretation of the Talagrand diagram was also updated. More information is given in the point-by-point response below.

1.2 Specific Comments

> Equation (3): The authors state that the conditional distribution of the image x at step t given x0 in the forward pass is defined as $q(\mathbf{x}_t|\mathbf{x}_0) = N(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, \alpha_t\mathbf{I})$ This suggests that the value of the variance approaches zero as t increases, reducing the distribution to be zero mean and zero standard deviation in the limit. The value of the variance in the conditional case, if I am not mistaken, should be equal to $(1 - \bar{\alpha}_t)\mathbf{I}$, given the transformation defined by Equation (1).

The reviewer is, of course, correct. Thank you for spotting this issue, which is corrected in the revised manuscript. The equation now reads:

$$q(\mathbf{x}_t|\mathbf{x}_0) = N(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1 - \bar{\alpha}_t)\mathbf{I}) \tag{1}$$

Fortunately, Equation 3 is not used in the code implementation, only its limit (equation 4) is used, which is not affected by this error.

> Equation (9): The reverse probability $p_{\boldsymbol{\theta}}(x_T)$ should not be parameterised by θ, since the distribution is defined in Equation (5), where no such parameters are present. If these distributions indeed differ the authors should explain what properties the parameterization defines in this specific case. Additionally, a technical mistake seems to be present in the term $q(\mathbf{x}_{1:T})|\mathbf{x}_0$, which should be equal to $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ correct?

The review is certainly correct. Thank you for finding these issues. The updated equation now reads:

$$E\left[-\log(p_{\boldsymbol{\theta}}(\mathbf{x}_0))\right] \leq -E\left[\log \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right] = -E\left[\log\left(p(\mathbf{x}_T)\prod_{t=1}^{T}\frac{p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right)\right] = L_{\text{elb}} \tag{2}$$

This comment pertains to the neural network description provided in paragraphs 185 and 190, and Table 1. The definition of the neural network is given recursively, with each block l being dependent on the block $l-1$. However, given how the levels are provided $l = 1, \ldots, L$ (L being the depth level) and $C_l = [16, 32, 64, 128]$, this description might be confusing for readers unfamiliar with the architecture. For example, one can make the mistake that the number of channels on the first level C_1 is equal to 16. However, as far as I understand the provided description, the block at depth level 4 contains 16 channels while the block at depth level 1 contains 128 channels. Therefore, the initial block corresponds to $l = 4$, while the "deepest" block corresponds to $l = 1$, which seem counter-intuitive given that l denotes the depth level. Consider these two cases: if $l = 1$ and $C_1 = 16$, then the inner block of block $l = 1$ is an identity and the recursion stops immediately. However, if one assumes that $l = 4$ and $C_4 = 16$, then the inner block at $l = 3$ contains $C_3 = 64$ etc. which results in the familiar U-net architecture, where the spatial dimension is reduced with each consequent block and the number of channels increases. This later assumption is not self evident from the provided description. Therefore, I suggest that the author either flip the depth indices l, such that $l = L, ..., 1$, or that they flip the C_l values $C_l = [128, 64, 32, 16]$ while keeping the indices intact.

We are sorry for the confusion. Indeed, the recursive definition should be reversed relative to the depth index l. In particular, at line 187 (submitted version)

inner block at level $l - 1$

Should rather be:

inner block at level $l + 1$.

Similar changes have been made at the lines 191-194. The input resolution is indeed at level $l = 1$ (with 16 channels) and the deepest levels $l = 4$ (with 128 channels). We verified that this is actually the case in the code implementation used (i.e. number of channels increases as spatial dimensions decrease) .

The authors provide a short description of the DINCAE method's training setup in the Results section, in paragraphs 250 and 255. While I believe that this is beneficial to the manuscript the description somewhat breaks the flow of the Results section. Therefore, I suggest that the authors move this description to the Appendix.

We agree, and the relevant text and the hyperparameters of DINCAE is now in the appendix.

Talagrand diagram in Figure 10

The authors compute the Talagrand diagram using the ensemble as an approximate distribution function, where each ensemble memeber represents an equally probable event realization. The authors sort the ensemble members in an ascending order for each masked pixel, independently. The resulting

empirical distribution functions and their corresponding ground truth values are used to construct the diagram. The histograms for both the dev and test datasets are displayed in Figure 10 and the authors conclude that: "Figure 10 shows the Talagrand diagram computed for the development and testing dataset. It can be seen that except for the two first and two last bins (corresponding to the probabilities between 3% and 97%), the Talagrand diagram is relatively flat. This shows that the produced probabilities are reliable, except for very rare events where the produced ensemble is underdispersive. The difficulty of predicting rare events is a known issue in machine learning (e.g. Kaiser et al., 2017) and a dedicated area of research."
Here I would like to raise a minor concern regarding the use of "probabilities are reliable" in this context. The produced probabilities are marginally reliable, since each pixel is treated independently from its neighbours. However, this does not necessarily imply that the joint distribution of the ensemble is reliable, which is not decernable from the conclusion reached by the authors. For example, consider taking the same ensemble forecast produced in this work, however, with its values randomly permuted between the corresponding members for each pixel. A permuted forecast like this would exhibit the same Talagrand diagram (since the sorting on a per-pixel basis restores the initial diagram conditions), however, the forecast would not be jointly reliable as the spatial relationships would be lost. Therefore, I suggest that the authors state that this evaluation method, as is, assesses the marginal reliability only and not the joint. Again, this is not a major concern for readers familiar with the evaluation technique, however, since the spatial correlation is an important asset of this proposed reconstruction method, a clarification of this would be welcome.
However, I do not agree that the excess number of observations in the extreme ranks implies that the method perform poorly for very rare events only. The excess denotes that the distribution described by the ensemble exhibits short tails, meaning, that a disproportionate number of observations fall into those ranks. These observations can including realizations that are not rare at all and should actually be described by other ranks. Therefore, I would suggest rewording this conclusion such that it reflect the notion of the distribution tails being too short rather than an explicit comment on the reliability of rare event forecasting.

The review is certainly right that the Talagrand and other statistics only test if the probabilities are *marginally* reliable. We clarified this in the revised manuscript and changed "reliable" by "marginal reliable" at several places in this section.

We also followed the reviewer's suggestion and removed reference to "rare" events and made explicit reference to the tails of the underlying PDF. The relevant sentence now reads:

> If the ensemble is generated from the same probability distribution as the observations, the ensemble is considered reliable. However, it is important to note that the Talagrand and other statistical tests described below only allow us to assess the reliability of the marginal PDFs (probability density function) evaluated for each pixel individually and not the joint PDF accounting for spatial correlations between pixels. [...] This shows that the produced probabilities are marginally reliable, except for the tails of the marginal PDF where the produced ensemble is underdispersive.

DINCAE comparisons

The output of the DINCAE method can be interpreted as a normal distribution, where the reconstruction is its mean, and the reconstruction error its standard deviation (or variance), correct? If so, I suggest constructing the Talagrand diagram for the DINCAE method as well, which would further demonstrate the impact of the proposed method's distributional capabilities compared to the baseline. The same comment applies to the evaluation using the CRPS method, where the DINCAE approach (given that the above assumption holds) can also be included.

I also suggest that the authors include the training and inference times for the DINCAE method, as well as the number of parameters of the DINCAE method, such that the reader can better asses the relative computational complexity of this new approach compared to the baseline.

The reviewer is correct that the DINCAE method provides a mean and standard deviation for every pixel, and the marginal PDFs are treated as a Gaussian distribution.

In the revised manuscript, we added the Talagrand diagram and the CRPS statistics (and its decomposition) for the DINCAE method, as they all rely only on marginal distributions, as pointed out by the reviewer. For DINCAE, the Talagrand diagram was constructed using the Gaussian cumulative distribution function, while for the CRPS statistics, we created an ensemble with 10 000 samples following the marginal PDF.

The corresponding table and figures have been revised in the new manuscript and show that the diffusion model is more reliable (assessing the marginal PDFs) than DINCAE.

The number of parameters of the optimal DINCAE model is 3.1 millions. The training time is 12 minutes on a GeForce RTX 4090 GPU. The inference time of the test and development datasets is 2.7 seconds which is significantly faster than the diffusion model.

Diffusion model performance conditional on the number of valid input image pixels

The proposed diffusion model is dependent on the valid pixel (pixels without clouds) in the input image to construct a spatially consistent reconstruction. This approach produces realistic reconstructions with a high degree of spatial correlation as can be seen in the provided examples. This, however, prompted the following consideration: how does the performance of the reconstruction degrade in relation to the number of valid pixel available in the input image? An evaluation like this could be an interesting inclusion in the current manuscript, providing a practitioner with the knowledge on how reliable the reconstruction is given how much information is present in the original input image. The ensemble spread might already describe such notions however, it might not be marginally reliable when considering images with a high degree of missing valid data.

We agree and have added the following additional test to the manuscript.

> Among the test data, we took the images with less than 30% of cloud cover (representing 99 images here). To these relatively clear images, we applied the cloud mask (potentially flipped in the longitude or latitude direction) chosen randomly from another image in the test dataset so that the total cloud coverage for every image is within a given range of 45% to 55%. If the cloud coverage is outside this range, then another cloud mask is chosen randomly until the target range is achieved. This procedure is repeated for different ranges, up to a range of 85% to 95%

of missing data.

The trained diffusion model was applied to these images, and the RMS error relative to the withheld (and independent) data was computed and is shown in Figure 1.

As expected, the RMS error rises with an increased amount of missing data. With a large amount of missing data, the diffusion model misses the context to reconstruct the field and the model acts as an unconditional diffusion model. It can also be seen that the RMSE does not show any abrupt augmentation.

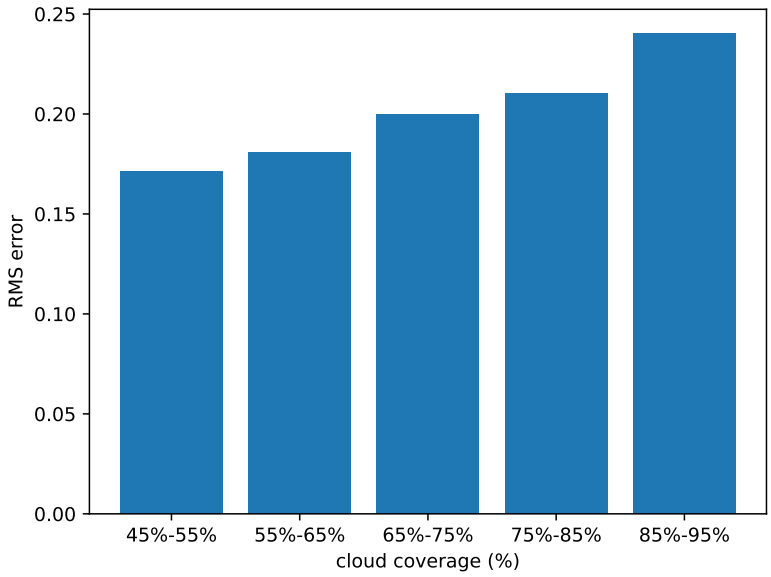

Figure 1: Impact of cloud coverage on the RMS computed relative to independent data (units $\log_{10}$ mg m^{-3})

On "For the validation and test data, we randomly took the cloud mask from other time instances to mask additional grid cells which will be used for validation. Only images with a cloud mask between 15% and 35% of the missing date were considered as an additional mask to obtain a sufficient number of "clouded" pixels without masking an image almost entirely." in paragraph 150: Does this mean that, when constructing an input image from the validation/testing datasets, a random image with 15% to 30% of missing data is selected (still from the same dataset) and its cloud mask is used to cover the current input image's pixels? If so, it seems that this approach can still result in a completely covered image if the image being masked has a coverage greater than 85%, correct? Or are only images with a coverage of less than 70% considered for the validation/test datasets? A few comments on this would be appreciated.

The reviewer is right, and the procedure, while not likely, could in theory result in a completely masked image. However, we checked that this is not the case for the generated dataset. The following has been added to the manuscript to clarify this point:

Only images with a cloud mask between 15% and 35% of the missing data were considered as an additional mask to obtain a sufficient number of "clouded" pixels and to reduce the risk that an image is masked almost entirely. We verified that, neither in the validation nor in the test dataset, were images masked entirely after applying the cloud mask.

On "During training, for each image of the training dataset, a different image is randomly selected (also from the training dataset) and its cloud mask is used to degrade clear pixels of the input image (Figure 3). The stage of degradation t of these pixels is randomly chosen between 1 and T." in paragraph 170: Can it not occur that the training image can be fully degraded after the additional cloud mask is provided (example: input image has 20% valid data and the mask has a cover of 80%)? Such training images might slow the convergence of the method as the denoising process is completely unguided. Or is this event rare in practice?

We agree that this can happen and would slow down the training process, as the reviewer points out. For the training data, we estimated this probability numerically (using 100 000 000 pairs of images chosen at random) and found that the probability is 0.00071289. It is indeed a quite rare event.

Furthermore, what is the benefit of setting the degradation value between 1 and T instead of just T ? If I understand correctly, during inference, each missing pixel is treated as being fully degraded. Is there a difference in performance compared to setting all pixels to the fully degraded value T ? Does this help in cases where the training image might be degraded to a high spatial degree (above example)? A few comments on this would be appreciated.

During inference, each missing pixel is only treated as fully degraded *initially* at step T. After removing the noise predicted by the neural network, those pixels will be at the step $T-1$. For inference, we need to apply the neural network multiple times (here $T = 800$) to reach the clear and non-degraded level ($t = 0$). So, the neural network needs to know how to handle intermediate degradation levels during inference. This information has been added to the manuscript.

On "For each image of the validation and test two datasets, the reconstruction process is repeated 64 times, leading to an ensemble of possible reconstructed fields." in paragraph 230: How did you choose the number of ensemble members (64 members) in the reconstruction? Was it determined empirically? If so, please provide an explanation.

We did not test different ensemble sizes. The number is rather guided by the typical ensemble size used in ensemble modeling in oceanography (*e.g.* Simon and Bertino, 2009; Ohishi et al., 2022) and meteorology (Buizza et al., 2008). In theory, the method should work better as the ensemble size increases towards infinity. If we had chosen too large ensemble sizes, one could have criticized the method as having only been tested in an impracticable setting.

On "In Barth et al. (2020), it has been shown that the accuracy of a reconstruction can be improved by averaging the obtained reconstruction over a certain number of epochs after the epoch 200." in paragraph 250: I do not fully understand this approach. Does this mean that, during training, intermediate models from epoch 200 onwards are saved and the mean reconstruction from each of those models is used as the final DINCAE output?

Yes, the reviewer is correct that this is essentially the approach employed here. It is similar to ensemble averaging different trained models, except that we use the same model at different epochs, which does not increase the computational costs of the training. In practice, we do not save the model weights at different epochs but apply the model to the test and development data and accumulate all the reconstructions, which are later normalized to compute the average.

1.3 Technical comments

> Figure 2, Figure 6, Figure 7, Figure 8: Consider adding lat, lon labels to the axis.

Done!

> Broken Latex mathematical symbol for $\bar{\alpha}_T$ in paragraph 180.

Fixed!

> The kernel size k_s (Table 1) does not require a subscript since it is a fixed value across levels. Consider omitting the subscript.

Ok, done.

> "As an illusion" in paragraph 215: misplaced use of the word "illusion". Consider replacing with "Illustration".

Done, sorry for the typo!

> Table 2: Typo in "desactivated". Additionally, consider explaining the meaning of the rows of the table as some are not self evident, for example "refinement step".

The typo is fixed, and the following has been added to the manuscript.

> In the case of a refinement step, the neural network is composed of two U-Nets: the first network provides an intermediate estimate of the missing data and the second U-Net uses the intermediate estimate and the original data to provide the final estimate. During training, the loss function is based on a weighted sum of the intermediate and final estimate. For inference, only the final estimate is used. The weights are considered as hyperparameters. More information is provided in Barth et al. (2022).

On "(corresponding to the probabilities between 3% and 97%)": Should this not be equal to "between 1.5% and 98.5%" since each interval has a weight of 1/65 implying, that the first rank covers realizations with a probability of occurrence between 0 and 0.015 and the last rank between 0.98 and 1? Therefore, the middle ranks exhibit a coverage between 0.015 and 0.98, correct?

Indeed, we changed the probabilities in the revised manuscript. (Originally we were considering two bins at the lower end and two bins at the high end to be affected, but we changed this assessment in the revised manuscript (in particular after adding the comparison with DINCAE)).

References

Barth, A., Alvera-Azcárate, A., Troupin, C., and Beckers, J.-M.: DINCAE 2.0: multivariate convolutional neural network with error estimates to reconstruct sea surface temperature satellite and altimetry observations, Geoscientific Model Development, https://doi.org/10.5194/gmd-2021-353, 2022.

Buizza, R., Leutbecher, M., and Isaksen, L.: Potential use of an ensemble of analyses in the ECMWF Ensemble Prediction System, Quarterly Journal of the Royal Meteorological Society, 134, 2051–2066, https://doi.org/https://doi.org/10.1002/qj.346, 2008.

Ohishi, S., Miyoshi, T., and Kachi, M.: An ensemble Kalman filter-based ocean data assimilation system improved by adaptive observation error inflation (AOEI), Geoscientific Model Development, 15, 9057–9073, https://doi.org/10.5194/gmd-15-9057-2022, 2022.

Simon, E. and Bertino, L.: Application of the Gaussian anamorphosis to assimilation in a 3-D coupled physical-ecosystem model of the North Atlantic with the EnKF: a twin experiment, Ocean Science, 5, 495–510, https://doi.org/10.5194/os-5-495-2009, 2009.