EGUsphere, author comment AC1
https://doi.org/10.5194/egusphere-2022-147-AC1, 2022

Reply on RC1

Sébastien Gardoll and Olivier Boucher

Author comment on "Classification of tropical cyclone containing images using a convolutional neural network: performance and sensitivity to the learning dataset" by Sébastien Gardoll and Olivier Boucher, EGUsphere, https://doi.org/10.5194/egusphere-2022-147-AC1, 2022

Section 1

- This study uses a very simple CNN design with two convolutional layers based on a paper from 2016. CNN research has progressed dramatically since then and more sophisticated architectures with much better performance for image classification now exist. Many of these have already found their way into atmospheric studies. While there is nothing wrong with using a simple model (and the simple model performs quite well in this case) the last 8-years of CNN research should probably be acknowledged in the background section.

We agree with the Reviewer that the field has moved since 2016. We have performed a literature review (see Table below) for TC detection and segmentation in both satellite and model data, which we have also included in the Introduction section of the revised manuscript. The Table shows that a range of architectures including ad hoc CNN, YOLO, U-Net, DeepLabv3+ have been used over the last 8 years. The bibliography of the revised manuscript has also been completed. The Reviewer is also right in saying that the simple models perform quite well, hence it is not clear that better performance can be obtained with the most complex architectures, and the simple models remain relevant for our study that focuses on cross-training of the CNN on different sets of reanalyses and the use of a wider set of performance metrics than in previous studies.

We have added the following lines in the Section 1:

"There is also a wealth of studies on the detection of TCs in satellite imagery, reanalysis and climate model outputs based on machine learning (ML) approaches. Table 1 summarizes notable studies published in the last eight years that implement neural architectures based on convolution layers. It is not surprising that this approach was favored because TCs have very distinct features which make them relatively easy to detect with convolutional neural networks. Since Liu et al. (2016), whose deep learning (DL) model only classifies pre-existent images, various subsequent studies have focused on improving the detection of all cyclones at once present in unidimensional or multidimensional meteorological images (e.g., Ebert-Uphoff and Hilburn, 2020) and climate model data (e.g., Matsuoka et al., 2018). This latter work focuses on the detection of cyclones using a CNN image classifier which operates on a sliding window of output from the Nonhydrostatic Icosahedral Atmospheric Model (NICAM) and studies the system

performance in terms of detectability. The detection can be either "coarse" by drawing rectangular envelopes around the cyclones (the studies are flagged as detection in the Purpose column of the Table) or "precise" by drawing the contours of the cyclones including their internal structure (studies flagged as segmentation). The main idea of these more recent studies is to apply new DL model architectures coming from computer vision research (e.g., U-Net, DeepLabv3, YOLOv3, Single Shot Detector, etc.) to the analysis of meteorological features such as cyclones."

Authors	**Year**	**Purpose**	**Dataset**	**Variables**	**NN Architecture**	**Performance at best**
Liu *et al.*	2016	Cyclone image classification	ERA-interim (rea) CAM5.1 (mod) NCEP-NCAR (rea) 20th Century Reanalysis	Pressure sea level, wind vectors at 10 meters and at 850 hPa, temperature at 200 and 500 hPa, total column water vapour	ad hoc CNN	Accuracy: 99 %
Racah *et al.*	2016	Cyclone segmentation	CAM5 (mod: 25 km)	The 16 channels of CAM5	ad hoc autoencoder	mAP@IoU=0.1: 52.92 %
Hong *et al.*	2017	Eye detection	COMS-1 (sat)	4 IR channels	GoogLeNet	RMSE: 0.02
Matsuoka *et al.*	2018	Detection of cyclone by sliding window and cyclone	NICAM (mod: 14 km) + NSW6 + HadISST	Outgoing longwave	ad hoc CNN	Probability of de

		image classifier		radiation		tection: 79.9–89.1 % false alarm ratio: 32.8–53.4 %
Kumler-Bonfanti *et al.*	2020	Cyclone segmentation	GFS (mod)	Total precipitable water	U-Net	Accuracy: 0.991 Dice coeff: 0.763 Tversky coeff: 0.75
Shakya *et al.*	2020	Cyclone image classification	KALPANA-I (sat) MOSDAC (sat)	IR, Vis	ad hoc CNN	Accuracy: 97 %
Shakya *et al.*	2020	Cyclone detection and path prediction	KALPANA-I (sat) MOSDAC (sat)	IR, Vis	RetinaNet and polynomial regression	RMSE: 5-15.55 %
Prabhat *et al.*	2020	Cyclone segmentation	CAM5.1 (mod: 25 km)	Integrated vapor transport, integrated	DeepLabv3+	IoU: 0.2441

				water vapor, vorticity, wind vectors at 10 meters and at 850 hPa, sea level pressure		
Pang *et al.*	2021	Cyclone detection	Satellite images from NII	Vis	DCGAN and YOLOv3	Accuracy: 97.78 % mAP@IoU=0.5: 81.39 %
Shi *et al.*	2022	Extratropical cyclone detection	ERA5 (rea)	Top net thermal radiation, mean sea level pressure, vorticity	Single Shot Detector	mAP@IoU=0.5: 79.34-86.64 %

Table. Summary of previous studies aiming at the detection or the segmentation of TC in satellite or model data. coeff: coefficient; IoU: intersection over union; IR: infrared; mAP: mean average precision; mod: model; rea: reanalysis; RMSE: root mean square error; sat: satellite; vis: visible.

Section 2

- Lack of large training data and potential difficulties applying ML models to new datasets are both concerns/points mentioned in this study. Data augmentation schemes and model regularization techniques like dropout and batch normalization are proven ways to improve the robustness of CNN models when applied to new datasets and in the face of limited training data. These have become so ubiquitous that almost all CNN-based image recognition studies now use at least one of them. I think not using or at the very least acknowledging the existence of these methods is a major omission, since they are both very common in the literature and very relevant to this specific problem.

We agree with the Reviewer that data augmentation schemes and model regularization techniques are important. They are not necessarily needed in this study because the performance of our retrievals is already very good. However they could become important in the next phase when we seek to detect TC in climate model simulations with a CNN trained on a reanalysis dataset. We have now mentioned these techniques in the

discussion Section and justify why we do not require them at this stage:

"Data augmentation (especially geometric transformations; Shorten and Khoshgoftaar, 2019) and model regularization techniques (e.g., weight-decay, batch normalization, dropout, etc.) are proven ways to improve the robustness of a CNN trained with a dataset of limited size. Our dataset contains 26,954 images, which is relatively small compared to the size of datasets encountered in many computer vision applications (for instance, Imagenet contains more than 14 million images). However using these techniques was not justified for our study because the performance of our CNN without data augmentation is already very high. Such techniques could however become very relevant in future work when we seek to detect TC in climate model simulations with a CNN trained on a reanalysis dataset. Indeed different climate models may simulate TC imperfectly and there is probably some value in offering a larger variety of TC structures to the training dataset. It is expected that the simulation of TCs increases in quality with the climate model resolution (Strachan et al., 2013) and climate models running at resolutions of 10 to 50 km are now commonplace. Likewise we would need to augment the number of images with very intense TC or TC migrating outside their usual domains because there are indications that such situations may become more frequent with global warming and we want to ensure these can be detected adequately in climate simulations."

New references added:

Shorten, C., Khoshgoftaar, T.M. A survey on image data augmentation for Deep Learning. *J Big Data*, 6, 60, 2019. doi:10.1186/s40537-019-0197-0

Strachan, J., P.L. Vidale, K. Hodges, M. Roberts, and M. Demory. Investigating global tropical cyclone activity with a hierarchy of AGCMs: The role of model resolution, *Journal of Climate*, *26*(1), 133-152, 2013. doi: 10.1175/JCLI-D-12-00012.1

- The use of the term "images" to refer to reanalysis data may be a bit confusing, particularly for those not familiar with reanalysis but perhaps familiar with CNNs which are often applied to photos. I think it is fine to use but it should be addressed early in the manuscript that "images" here does not refer to photographs, optical data, or data from any type of imager for that matter. Instead, these are chunks of regularly gridded data extracted from a numerical simulation.

We agree with the Reviewer, we have added the following text in Subsection 2.3:

"2.3 Images

In computer vision, the term image refers to a stack of matrices (also called a 3D tensor), with each matrix representing an information channel. For example, RGB images are formed of a stack of matrices of numerical values coding the red (R), green (G) and blue (B) color intensities of each pixel of a photograph. Our use of the term image is a generalization of the concept of RGB images. In the rest of our study, an image refers to a stack of gridded data extracted from a different variable of ERA5 or MERRA-2 on a given geographical area. Unlike for an RGB image, the channels cannot be combined; we thus graphically represent each channel separately."

- There is some discussion of ensuring that input data has been interpolated to the correct grid resolution for consistent inputs to the CNN, but is there any concern about the non-uniform grid spacing on a lat lon grid? Samples closer to the poles will cover a smaller area.

We believe this is not a major problem as we are concerned with cyclones in the tropical region where the deformation induced by a regular lat-lon grid is small (the area of a

surface element in spherical coordinates is proportional to cos(lat)). The Reviewer has nevertheless a very valid point for the detection of TC that migrate polewards and mid-latitude depressions as the grid spacing would matter more at mid-latitudes. This is also an interesting point because data augmentation methods may help to account for such deformation as a function of latitude and therefore could be used to increase the robustness of the CNN. We now discuss this in the Discussion section, just after the previous addition about data augmentation:

"In this study we work on images created on a regular lat-lon grid, which potentially introduces a deformation because of the cos(latitude) dependence of a displacement element along the longitude. Such a deformation is small in the tropical region and therefore is not thought to be a problem for our analysis. However it increases as a function of latitude, so it may become an important factor to consider for TC that migrate polewards or for the detection of mid-latitude depressions. Data augmentation techniques that introduce deformed images in the training datasets could help to increase the robustness of the CNN in these situations."

- Line 85: Only 5 variables are listed. You should mention that you are using winds and temps at two levels which gets you to 8 input variables.

We agree, we have modified the following sentence: "We use fields of sea level pressure, precipitable water vapor, the two components of the wind (at the surface and at 850 hPa) and the temperature at two different pressure levels (see Table 2)."

- Line 85: I think these input variables are very reasonable choices, but no motivation was given for why these specific ones were used.

Indeed, these input variables represent a very reasonable choice. First of all, we based our study on the same variables used by Liu *et al.* (2016). The choice of most of these variables is confirmed by Racah *et al.* (2017), Prabhat *et al.* (2020), as well as Kumler-Bonfanti *et al.* (2020) concerning the precipitable water and Shi *et al.* (2022) concerning the mean sea level pressure. This latter study also considers the vorticity, which is a function of the wind field (see the neuronal network architectures in the Table above). It is likely that there is redundant information in all these variables. We recognize that quantifying the relative contributions of these variables in the classification decision of the CNN would be interesting, with the aim of reducing the number of variables. It is in this perspective that we have written our potential future work section.

We also added the following lines in Section 2.2 "Meteorological reanalyses":

"We have followed Liu et al. (2016) and considered an extensive set of meteorological variables to detect TC (see Table 2). This choice is confirmed by subsequent studies (Racah *et al.*, 2017; Prabhat *et al.*, 2020; Kumler-Bonfanti *et al.*, 2020). It is likely that there is redundant information in this set of variables. An interesting follow-up work will be to investigate the relative contributions of these variables in the classification decision of the CNN, with the aim of reducing the number of variables."

Section 3

- Section 3.2.3: It seems that a great deal of effort has been put into identifying non-cyclone training samples that thoroughly cover the region where cyclones have been observed. However, this appears to exclude any negative samples from other parts of the North Atlantic basin. Including negative samples from the Eastern Atlantic or the extratropical storm track region is important if the algorithm will ultimately be applied to the entire basin to ensure there is not unexpected behavior in these areas. The background section mentioned the potential of TCs moving farther north in the Atlantic

in a warming climate.

This is a valid point but out of scope for this study. It is relatively easy to add non-TC images further north in the basin. It is more complicated to add TC images further north as there are fewer TC on the edge of the basin. This is another area where data augmentation methods could help. We now discuss this in the Discussion section (text repeated from our reply above):

"Likewise we would need to augment the number of images with very intense TC or TC migrating outside their usual domains because there are indications that such situations may become more frequent with global warming and we want to ensure these can be detected adequately in climate simulations."

- Section 3.3 and 3.4: I was left confused about exactly how the training was performed, and I think more details should be included to improve reproducibility. Here are some notes:
- Information like the loss function and optimizer used are more important to include in the main text than details about the hardware used. This information is key to reproducibility, but the hardware is not.

We agree with the Reviewer. We now present this information as a Table next in the main text. Table A3 has been moved to Section 3.3.

- Is the early stopping based on training loss (line 203) or test set loss (table A3 line 2)? Is there evidence of overfitting without early stopping?

We observe a slight overfitting with the growing number of training epochs, which is why we use the early stopping method that corresponds more or less to the elbow method. The early stopping method is based on the test set loss, which is not biased. We have added the information in Section 3.3 on model training:

"Overfitting has been noticed during the training of the model. We have observed the characteristic U-shape of underfitting followed by overfitting by plotting the value of the lost function calculated using the validation dataset against the number of epochs. In order to automatically avoid overfitting, we used two Tensorflow callbacks: early stopping and model check point. The first callback stops the training after N epochs without further improving the training metric (N is set to a value of 10). Early stopping behaves more or less like the elbow method."

- Which dataset was used for the hyperparameter optimization (HPO)?
- The exact range of parameters tested during HPO is not provided
- What kind of train/test/validation split was used for the HPO process? Was a separate test set held out during hyperparameter tuning to avoid an overfit?

First of all, we would like to clarify our intention regarding the optimization of the hyperparameters: while we follow the approach by Liu *et al.* (2016), these authors do not provide the values of the training hyperparameters such as batch size, optimizer and learning rate. Instead of fixing these values in an arbitrary way, we searched for local optimal values of these hyperparameters to maximize the performance of the CNN. However, these values depend on the image dataset being used (ERA5 32px@0.25, MERRA-2 32px@0.25, etc.) and the way it is split between the classical training/validation/test sub-datasets. The training and validation datasets are used for the optimization phase while the test dataset is used for the unbiased evaluation of the performance. We conducted four optimizations for the different image datasets but for the same split (0.7/0.15/0.15) and we obtained the same values for the optimizer and the learning rate with very similar performances. Only the batch size differs among these four

optimizations, so we decided to set a value as large as possible given the memory of the GPU cards at our disposal. Of course, these optimal values are only valid for the given split, however we think that they are close to the global optimum, because the performances vary very little according to the different values of these hyperparameters. But above all, by fixing the same hyperparameter values for all our numerical experiments, we avoid attributing the variability of the studied metrics to hyperparameter changes. The range of hyperparameters is added in a new column called search space. The ranges are based on values commonly found in the literature.

We have added these aspects in the revised manuscript:

"Our work is based on the study by Liu *et al*., but these authors did not provide the values of their training hyperparameters such as batch size, optimizer and learning rate. Instead of fixing these values in an arbitrary way, we search for local optimal values of these hyperparameters to maximize the performance of the CNN. Since training times are relatively short on our GPU cluster, we performed a grid search hyperparameter optimization to maximize the score of the training metric, using conventional hyperparameter value ranges (the number of combinations of the search space is 48). We conducted four optimizations for the different image datasets but for the same training / validation / testing split (0.70/0.15/0.15). We obtained the same values for the optimizer and the learning rate, for very close performances. Only the batch size differs, so we decided to set a value as large as possible given the memory of the GPU cards at our disposal. Of course, these optimal values are only valid for the given split, however we think that they are close to the global optimum, because the performances vary very little according to the different values of these hyperparameters."

- Section 3.4: This section would benefit from a re-write. Currently, it introduces how k-fold cross validation is typically performed only to then say you used a different procedure, then adds that you actually train 20 models with 1/10th sized test sets. Perhaps open by describing precisely what you did and follow with the motivations for some of your decisions (dealing with temporal autocorrelation and increasing the number of samples). Also, k-folds is a common ML technique. You could just cite a reference here to avoid some of the clutter of talking about typical k-folds strategies that weren't used in this paper, e.g. Bishop, C. M., 2006, "Pattern Recognition and Machine Learning", Ch. 1.

In response to the Reviewer's remark, we have rewritten the description of the evaluation of the metrics as following:

"For the evaluation and comparison of the metrics (developed in Section 4.2), we wanted to be able to calculate the expected value and the uncertainty of the metrics, without bias. To that end, we applied an iterative cross validation method which consists in repeating 20 times a cross validation method. We chose the k-fold method (Bishop, 2006), with k equal to ten, as the cross validation method. We obtained a mean of the metrics for each k-fold iteration. By applying the central limit theorem on this set of metric means, we could compute the expected value and the uncertainty of the metrics.

In order to avoid any bias, we took care to check if the central limit theorem can be applied, by testing the normality of the distribution of the metric means using the Shapiro-Wilk statistical test (brief non-mathematical presentation given in Appendix B1). Moreover, images coming from a time series of tracks from the same cyclone may be found in both the training and test datasets, which would induce some dependance between the training and test datasets due to the autocorrelation within individual cyclone tracks. In order to avoid such a bias, the k-fold split is based on sampling the years randomly and balancing the folds as much as possible. The partitioning combinations are calculated in advance in order to guarantee the uniqueness of their composition. Scale

bias is also avoided by standardizing the channels of the images online, just before training the CNN.

Finally, for the comparison of the metric means, we chose to apply the Kruskal-Wallis statistical test (brief non-mathematical presentation given in Appendix B2) for an alpha level of 1 %, because the Shapiro-Wilk test was negative for most distributions of metric values of our experiments, invalidating the use of the Student's t-test.

For the experiment of highlighting the problem with the accuracy (point developed in Section 4.1), we applied the classical hold-out method, avoiding the autocorrelation between images belonging to a same cyclone track, with the following partitioning: 70 % of the data for the training dataset and 30 % of the data for the test dataset."

- Line 241: It would be helpful to move this note about standardizing the input variables into the "data" section. Or Section 3.2 "image preparation"

We agree to mention image preparation, the standardization of the data of each channel taken separately in Section 3.2, as it is a necessity for the training of the CNN.

The following subsection has been added:

"3.2.7 Data standardization

Neural network models learn a mapping from input variables to an output variable. The input variables have nearly always different scales and large scale differences are detrimental to the learning process of neural networks. In order to ensure that each variable is equally important, whatever its range of values, input variables are rescaled to the same scale. There are several methods such as standardization (or Z-score normalization) which consists in recalculating the values of the variables so that their mean and standard deviation equal to zero and one, respectively. In our study, we have systematically standardized each channel of the images, by calculating the means and standard deviations of the channels on all the images of the training set. The validation and test image datasets are excluded from the calculation of the mean and standard deviation, to avoid that information about the validation and test datasets leak into the training phase. However the validation and test datasets are also scaled using the mean and standard deviation of the training dataset."

Moreover it is also necessary to specify the modality of this standardization in the context of the iterative k-fold method. We therefore felt that line 241 belongs better in Section 3.4 "Evaluation of metrics", just after the presentation of the iterative k-fold method.

- The authors justify use of the 2-convolution layer model by saying it will help prevent overfitting, but no analysis is done to back up this argument. Is overfitting actually a problem?

Indeed, overfitting is observed during the training of the CNN for a significant number of epochs. When plotting the value of the lost function (calculated on the validation dataset) against the number of epochs (using the software Tensorboard), we observe the characteristic U-shape of underfitting followed by overfitting. The overfitting was automatically prevented using the early stopping callback of the Keras programming library that acts as the elbow method.

Section 4

- Maybe I missed it, but I don't think the "simple models" shown in Figure 7 are described anywhere. Also, the legend calls them "metrics" is it supposed to say

"models"?

The Reviewer is quite right. The legend of Figure 7 has been fixed and we have added the following explanation about the simple models in Section 4.2.1:

"Indeed, the usefulness of a model is measured by the difference between its performance and that of models based on simple rules or a domain specific baseline. For instance, we implement the following simple models (from the software library scikit-learn): "most frequent" which always predict the most frequent class observed in the training dataset (i.e., background), "stratified" which generates randomly predictions at probabilities that respect the class distribution of the training dataset (i.e., 1/3 cyclone, 2/3 background), "uniform" which generates predictions uniformly at random background or cyclone with equal probability."

We removed the "constant" and "prior" models that behave identically to "most frequent" in our experiment.

Section 5

- I remain skeptical of the claim in the conclusions section that the lower performance of the CNN on MERRA-2 implies that MERRA has lower information content for TC identification. I suppose one can argue that ERA5 has more information content simply because it is higher resolution, but that additional information may not improve cyclone detection. Only one CNN was tested, and perhaps a larger CNN, different input fields, different training procedure, data augmentation, etc. would yield better results on the MERRA data. Are all the storms present in HURDAT2 present in MERRA? If so, one could likely design a model better suited for the MERRA dataset. I think a more reasonable claim is made at the end of Section 4.2.2: "Thus we can conclude that the ERA5 dataset is more information rich than the MERRA-2 dataset for the classification of cyclone images *using our CNN.*"

We agree with the Reviewer that our conclusion about MERRA-2 having less information content for TC image classification than ERA5 should be qualified and we adopt the rephrasing proposed by the Reviewer. The following lines have been added to the conclusion:

"Applying an ERA5-trained CNN on MERRA-2 images works better than applying a MERRA-2 trained CNN on ERA5 images, which suggests that ERA5 has a larger information content in the framework of our CNN. This is also consistent with the findings of Malakar *et al.* (2020) who analyzed the error in the location of the center, maximum winds and minimum pressure at sea level in six meteorological reanalyses including ERA5 and MERRA-2 for the evolution of 28 TCs occurring between 2006 and 2015 over the North Indian Ocean, with respect to the observations of the Indian Meteorological Department (IMD). The authors of this study show, among other things, that the ERA5 dataset captures the evolution of these TCs in a more realistic way than MERRA-2 (i.e., smaller errors in the previous variables). They also show that ERA5 and MERRA-2 can capture the intensity of the TCs from the depression stage to the very severe cyclonic storm stage but not from the extremely severe cyclonic storm stage for which the intensity of the TCs is underestimated. However, they conclude that of the six datasets, ERA5 provides the best representation of the TC structure in terms of intensity. Finally, the study published by Hodges *et al.* (2017) shows that 95 % of the Northern Hemisphere TC tracks, from the IBTrACS database that includes HURDAT2, are present in MERRA-2. Unfortunately, this study does not include ERA5. It also confirms the underestimation of cyclone intensity in MERRA-2 compared to observations."

New references added:

Hodges, K., Cobb, A., & Vidale, P. L. (2017). How well are tropical cyclones represented in reanalysis datasets?, *Journal of Climate*, *30*(14), 5243-5264.

Malakar, P., Kesarkar, A. P., Bhate, J. N., Singh, V., & Deshamukhya, A. (2020). Comparison of reanalysis data sets to comprehend the evolution of tropical cyclones over North Indian Ocean. *Earth and Space Science*, 7, e2019EA000978. doi: 10.1029/2019EA000978

Technical comments

We have implemented these technical corrections:

- Line 10: specify spatial interpolation
- Figures 4, 5, 10, and 11 are missing units. If they are dimensional you could add colorbars, if they are all standardized you could mention it in the caption.
- Line 7: "similar locations and times to the TC containing images"
- Line 8: "accuracy, but"
- Line 9: "activity, but"
- Line 19: "causes"
- Line 24: "Indeed,"
- Line 46: "output from the Nonhydrostatic"
- Line 93: why "usually"?
- Line 94: "relevant" is an odd word choice, maybe "appropriate"?
- Line 98: "are able to capture better" --> "better capture"
- Line 101: strike "So"
- Line 107: add a comma after "indeed"
- Line 148: "a third" --> "one third"
- Line 276: Figure number is missing
- Line 287: change "experienced"
- Line 287: strike "well"
- Line 241: "image" --> "images"