https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

Classification of tropical cyclone containing images using a convolutional neural network: performance and sensitivity to the learning dataset

Sébastien Gardoll[1] and Olivier Boucher[1]

[1]Institut Pierre-Simon Laplace, Sorbonne Université / CNRS, Paris, France

Correspondence: Sébastien Gardoll (sebastien.gardoll@cnrs.fr)

Abstract.

Tropical cyclones (TCs) are one of the most devastating natural disasters, which justifies monitoring and prediction on short and long timescales in the context of a changing climate. In this study, we have adapted and tested a convolutional neural network for the classification of reanalysis outputs according to the presence or absence of TCs. We use a number of meteorological variables to form TC-containing and background images from both the ERA5 and MERRA-2 reanalyses. The presence of TCs is labelled from the HURDAT2 dataset. Special attention was paid on the design of the background image set to make sure it samples similar location and time to the TC-containing image. We have assessed the performance of the CNN using accuracy but also the more objective AUC and AUPRC metrics. Many failed classifications can be explained by the meteorological context, such as a situation with cyclonic activity but not yet classified as TC by HURDAT2. We also tested the impact of interpolation and of "mix and match" the training and test image sets on the performance of the CNN. We showed that applying an ERA5-trained CNN on MERRA-2 images works better than applying a MERRA-2 trained CNN on ERA5 images.

1 Introduction

Tropical cyclones (TCs) represent a major hazard for life and property in exposed regions of the world. There are still many unanswered questions on the number, intensity, duration, trajectory and probability of landfall of tropical cyclones in a warming climate (Emanuel, 2005; Webster et al., 2005; Chan, 2006; Vecchi et al., 2019; IPCC, 2021; Wu et al., 2022). IPCC (2021) estimated that "it is likely that the global proportion of major (Category 3–5) tropical cyclone occurrence has increased over the last four decades" but "there is low confidence in long-term (multi-decadal to centennial) trends in the frequency of all-category tropical cyclones". It has also been shown that global warming cause TCs to move further north in the North Atlantic and North Pacific basins (Kossin et al., 2014; IPCC, 2021; Studholme et al., 2021), which could have dire consequences for some coastal cities.

Better modelling of TCs in climate models is a prerequisite to estimate changes in associated damages. Better projections of the changes in TCs also require an understanding of the respective roles of decadal variability and climate trends. The automatic detection of TCs in climate model outputs is central to our ability to analyze results from climate projections. Indeed

TCs can only be simulated in models with sufficient horizontal and vertical resolutions (Knutson et al., 2015; Vecchi et al., 2019; Roberts et al., 2020; Jiaxiang et al., 2020) and such models produce huge volumes of output data. Thus it is important to have the capability to analyze such datasets in an efficient manner. While climate data are first produced and then analyzed, the climate modelling community is also moving in the direction of "on-the-fly" (also called in situ) data analysis in order to reduce the volume of data to be stored and the environmental impacts of such storage.

Climate modellers have developed "physical algorithms" to detect TCs based on the translation of their physical characteristics into identification criteria (e.g., Walsh et al., 2007; Horn et al., 2014; Bosler et al., 2016; Singh et al., 2022). Such detection algorithms generally rely on the identification of a spatial feature typical of a TC at all available time steps and a temporal correlation procedure to track the time consistency of the detected features and establish a trajectory. They are usually applied in predefined regions prone to TCs though it is not unusual for a TC to move outside its natural domain, hence it is important to apply the algorithms on a larger domain. These physical algorithms require to set up a number of thresholds which may depend on the climate model being considered and its resolution. For example, in the Stride Search algorithm (Bosler et al., 2016), a TC is identified if four criteria are met: maximum vorticity above a threshold, distance between the gridpoints of maximum vorticity and minimum sea level pressure below a threshold, the presence of a maximum vertically averaged temperature larger than its environment, and distance between the gridpoints of maximum vertically averaged temperature and minimum sea level pressure below a threshold.

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 (Liu et al., 2016; Park et al., 2016; Kurth et al., 2017; Hong et al., 2017; Kim et al., 2019). This is not surprising because TCs have very distinct features which make them relatively easy to detect with convolutional neural networks (CNN). This is part of a much larger trend to use ML approaches for object detection in meteorological images (e.g., Ebert-Uphoff and Hilburn, 2020) and climate model data (e.g., Matsuoka et al., 2018). This later work focuses on the detection of cyclones using a CNN image classifier which operates on sliding window of output from Nonhydrostatic Icosahedral Atmospheric Model (NICAM) and studies system performance in terms of detectability. Most approaches for TC detection use supervised methods which require a training dataset. While such techniques are now mainstream, they are not always well documented and their description may lack sufficient details which are often key in ML. Studies evaluating the performance and sensitivity of TC detection algorithms to the input and training datasets are also relatively scarce. It should be noted that TC datasets exist for the past observed climate record (satellite data, reanalysis) but it may not be practical to generate such datasets in climate model outputs for every new simulation that is made and to which the detection algorithm is to be applied. Thus it is important to understand how a supervised method may depend on the training dataset if it is to be applied to a dataset of a slightly different nature.

In this context and for the above-mentioned reasons, we have developed in this study a detailed procedure for building training datasets and testing the performance of the TC detection algorithm to some of its parameters. In Section 1, we present the data used to generate the images to be classified. Then in Section 2, we explain the architecture of the classification model, its training, the evaluation method to assess its performances, as well as the processes for generating the images to be classified. In Section 3, we present the results of our experiments in terms of accuracy and the other evaluation metrics. We further present an investigation on misclassified images and some suggestions for future work. Finally we summarize our contribution in the last section.

2 Data

2.1 TC dataset

Several datasets of TCs exist: we can flag here ExtremeWeather (Racah et al., 2017), ClimateNet (Prabhat et al., 2020), and the International Best Track Archive for Climate Stewardship (IBTrACS, Knapp et al., 2010, and references therein). In this study we use the North Atlantic National Hurricane Centre (NHC) "best track" hurricane database (HURDAT2; available from www.nhc.noaa.gov/data/#hurdat; Landsea and Franklin, 2013) because it is known as a high quality dataset for the North Atlantic basin. Quality and quantity of the training dataset are essential for the accuracy and performance of the ML model. In particular it is important for the dataset to be comprehensive (i.e., there is no missed TC) and homogeneous (i.e., the criteria for deciding if a feature qualifies as a TC are used consistently in space and time). The HURDAT2 dataset is reputed to be comprehensive for the period after 1970 (Landsea et al., 2010). It is more difficult however to ascertain its homogeneity especially for short duration TC.

HURDAT2 contains six-hourly (0, 6,12, 18 UTC) information on the location, maximum winds, central sea level pressure, and (since 2004) size of all known tropical cyclones and subtropical cyclones. The intensity of the TC are categorized into several categories, as shown in the Table A1 in the Appendix. We consider the HU and TS categories as being TCs and the other categories (including tropical depressions) as not being TCs.

2.2 Meteorological reanalyses

We use two different reanalyses upon which we train and apply our CNN. The ECMWF Reanaysis 5$^{\text{th}}$ generation (ERA5) is the current atmospheric reanalysis from the European Centre for Medium-Range Weather Forecasts (Hersbach et al., 2020). The Modern-Era Retrospective Analysis for Research and Applications, version 2 (MERRA-2) is the current atmospheric reanalysis produced by NASA Global Modeling and Assimilation Office (Gelaro et al., 2017). These two reanalyses differ in the atmospheric models used, the range of data being assimilated, and the details of the assimilation scheme. They also differ in their spatial resolution. ERA5 is retrieved from the ECMWF archive at a resolution of $0.25° \times 0.25°$ while MERRA-2 is provided at a resolution of $0.5° \times 0.6°$. The atmospheric variables relevant to TC detection are available in both reanalyses (as proposed by Liu et al. (2016)). We use fields of sea level pressure, the two components of the wind, temperature and precipitable water vapor (see Table A2 in the Appendix).

Table 1. The layers of our CNN. The convolutional layer parameter are denoted as <filter size>−<number of filters>. The pooling layer parameters are denoted as <pooling frame>. The fully connected layer parameters are denoted as <number of neurons>. For the activation function of the neurons, "relu" stands for the rectified linear unit whereas "sigmoid" stands for the logistic sigmoid function. Output tensor shapes are also provided for each layer of the CNN for input images of size (16,16,8) and (32,32,8). The number of trainable parameters are 5,053 for images of 16×16 px and 30,653 for images of 32×32 px.

Layer type	Parameters	Activation	Output tensor shape for image of 16×16 px	Output tensor shape for image of 32×32 px
convolutional	3×3−8	relu	14, 14, 8	30, 30, 8
pooling	2×2	-	7, 7, 8	15, 15, 8
convolutional	3×3−16	relu	5, 5, 16	13, 13, 16
pooling	2×2	-	2, 2, 16	6, 6, 16
flattening	-	-	64	576
dense	50	relu	50	50
dense	1	sigmoid	1	1

3 Methods

3.1 Classification model

In this study we implemented a binary classifier of cyclone images based on the work of Liu et al. (2016), with slight modifications. Table 1 shows the architecture of our CNN which is divided into two parts: a feature extraction part and a classification part. The feature extraction part is composed of the convolution layers whose filters are responsible for the extraction of features of cyclone present in the input images of the CNN. These features are the basic elements used for the classification of the images, implemented by the dense layers, and determine if the images represent cyclones or not, usually by outputting probabilities. As noted by Liu et al. (2016), using a shallow convolutional neural network is relevant for a relatively small number of images in the training dataset because the network only has a small number of parameters to train.

Our modifications compared to the work of Liu et al., concern the size of the convolutional filters and the number of neurons in the last dense layer. The characteristics of their CNN are described in Table 2. Indeed, our convolutional filters are smaller: 3×3 instead of 5×5 for Liu et al. We thought that smaller filters are able to capture better the features of cyclones on small images, especially for the 16×16 pixels (px) images. In addition, 3×3 filters are more conventional now. Note that the number of trainable parameters are very much the same between our CNN and that of Liu et al.. Lastly, Liu et al. describe a final layer with two neurons using the logistic sigmoid activation function. So this layer outputs two probabilities: the probability that the input image represents a TC and the probability that the image represents the background, but the outputs are not correlated and the sum of the probabilities can be larger than one. In this study, we use the conventional approach of binary classification

Table 2. The layers of the CNN by Liu et al. for comparison with ours. This Table follows the same syntax as Table 1. The number of trainable parameters are 5,776 for images of 16×16 px and 24,976 for images of 32×32 px.

Layer type	Parameters	Activation	Output tensor shape	Output tensor shape
			for image of 16×16 px	for image of 32×32 px
convolutional	5×5−8	relu	12, 12, 8	28, 28, 8
pooling	2×2	-	6, 6, 8	14, 14, 8
convolutional	5×5−16	relu	2, 2, 16	10, 10, 16
pooling	2×2	-	1, 1, 16	5, 5, 16
flattening	-	-	16	400
dense	50	relu	50	50
dense	2	sigmoid	2	2

by considering one output neuron activated by a sigmoid function. So a probability value that tends to zero classifies an image as background while a value that tends to one classifies an image as cyclone.

By construction, the size and the number of channels of the input images in a CNN are fixed. Using different image sizes and/or numbers of channels would require modifying the network architecture and retraining it. Indeed the properties of the dense layers of the network depend on the image shape (i.e., the number of neurons). Thus, image classification using a CNN implies the production of training and testing datasets of a given shape, irrespectively of the atmospheric reanalyses, ERA5 or MERRA-2, being considered. In our study, the size of the images is 32×32 px or 16×16 px with the eight variables as the channels of the image (3D tensor). Of course, the channels must correspond to the same atmospheric fields in the same units across the two reanalyses and must be arranged in the same order. The next section explains how we tackled the production of a homogeneous dataset.

3.2 Image preparation

3.2.1 Principles

The training of a CNN classifier is based on the optimization of its parameters using gradient descent and backpropagation techniques. Roughly speaking, the training process presents a batch of images as an input to the CNN. The training process modifies the parameters of the CNN in order to improve the classification of the batch, according to a chosen loss function. For a binary classifier, this process implies the presentation of images containing a TC, but also images not containing a TC, called background images. We now explain the data engineering involved in selecting both TC-containing and background images using the HURDAT2 dataset of cyclone tracks and the ERA5 and MERRA-2 reanalyses.

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

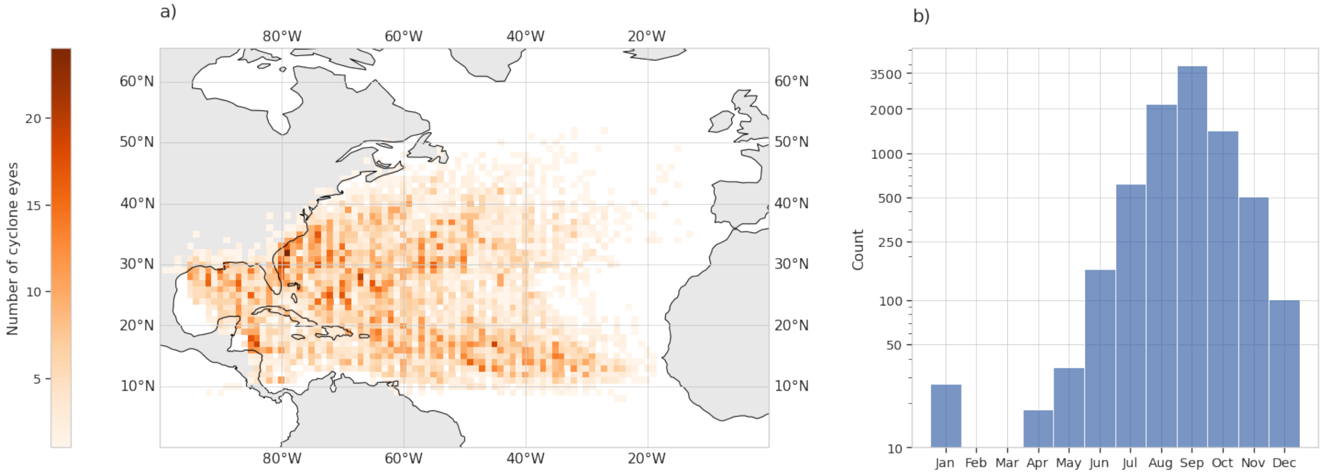

Figure 1. (a) Counts of TC-containing images per 1°×1° gridbox. (b) Histogram of the count of TC-containing images according to the month of the year. (a) and (b) are computed over the period 1980-2019.

3.2.2 TC-containing image generator

The HURDAT2 dataset provides locations and dates of TCs as part of the cyclone metadata. We create images centered on the cyclone positions in the reanalysis for the dates indicated in HURDAT2. The different channels of the images consist of the selected variables from the reanalyses as discussed above. We consider all cyclones with HU and TS status (see Table A1) that are located over the ocean, islands and coasts, over the period 1980-2019. Most TCs are found during the Atlantic hurricane season from May to December but we also consider a few events identified by HURDAT2 as TCs outside these months. Figure 1 shows the spatial and temporal distributions of the TC-containing images.

3.2.3 Background image metadata generator

Extracting background images requires some thought because the performance of the CNN depends on those and whether they sample the diversity of TC-free situations. The idea here is to reuse the HURDAT2 database so that, for each location and date with a TC, we choose two dates in the past where no TC is present. We also check that the date was not already selected as the TC-free situation for another TC-containing image, so that all background images are distinct to each other. Once the dates are selected, we can extract the corresponding images. Figure 2 shows a Unified Modeling Language version 2 (UML2) activity diagram of the background image metadata generator and specifically how we compute the two dates from each date of a TC track. The first date is computed by subtracting between 48 and 168 hours randomly (2-7 days) to the date of the TC track to generate the first date, and between 336 and 504 hours to generate the second date (2-3 weeks). Then the algorithm checks if each computed date leads to a background image that is in the immediate vicinity of any other TC track (status HU or TS as before) within a 48 hours time frame in the past or in the future, or to an already selected background image within a 12 hours time frame. If this is the case, we iterate by subtracting from the faulty date either 54 hours (48+time resolution) if

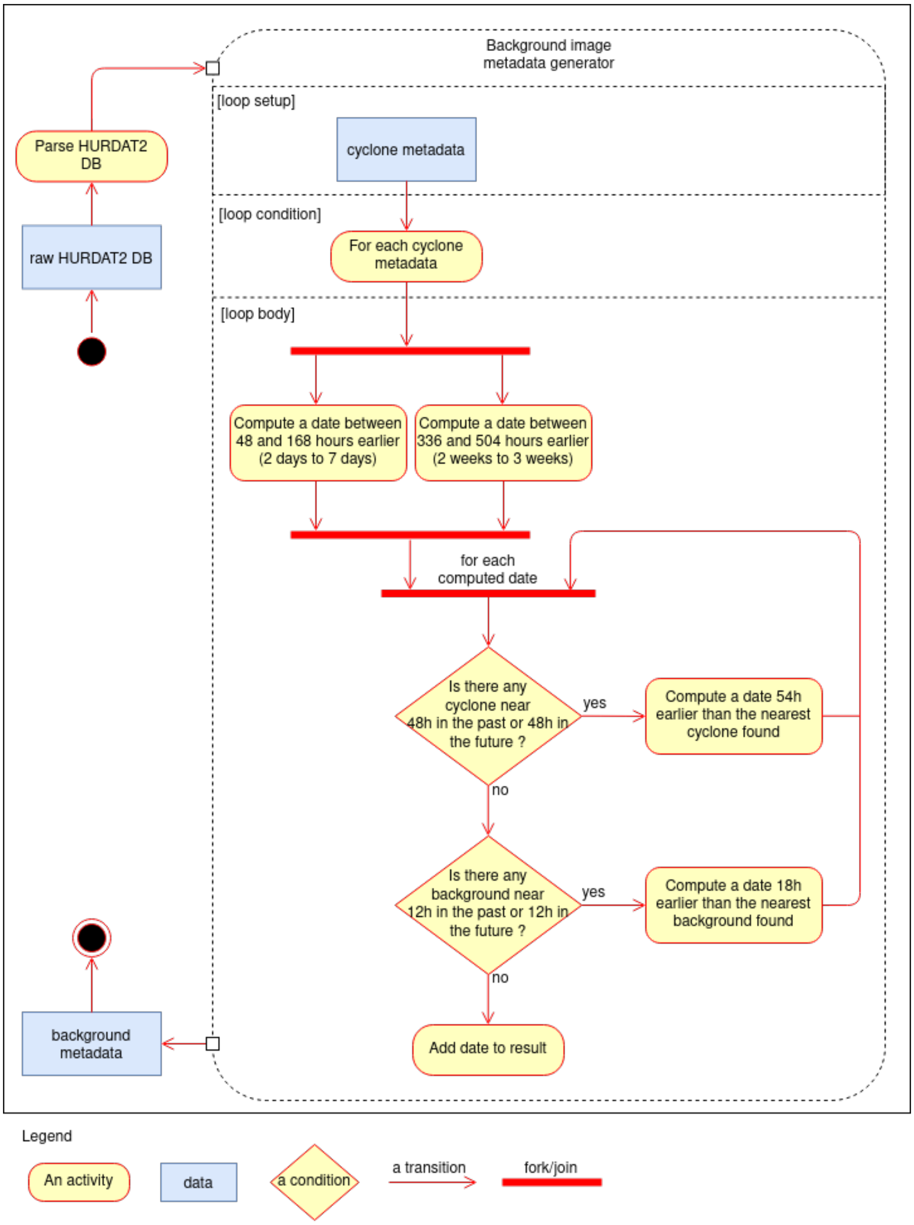
Background image
metadata generator
[loop setup]
cyclone metadata
[loop condition]
For each cyclone metadata
[loop body]
Parse HURDAT2 DB
raw HURDAT2 DB
Compute a date between 48 and 168 hours earlier (2 days to 7 days)
Compute a date between 336 and 504 hours earlier (2 weeks to 3 weeks)
for each computed date
Is there any cyclone near 48h in the past or 48h in the future ?
yes
no
Compute a date 54h earlier than the nearest cyclone found
Is there any background near 12h in the past or 12h in the future ?
yes
no
Compute a date 18h earlier than the nearest background found
Add date to result
background metadata
Legend
An activity
data
a condition
a transition
fork/join

Figure 2. UML2 activity diagram of the background image metadata generator.

the background metadata intersects a cyclone track or 18 hours (12 + time resolution) if it intersects another background image metadata.

Overall our background image metadata generator has the following advantages:

- our background images do not include a TC by construction;

- the meteorological, geographical and temporal contexts of the background images are close to those of the TC-containing images generated on the basis of the HURDAT2 data. In this way, we hope to better train the model at the classification decision boundaries;
- the ratio of background over TC-containing images is constant by construction (with a third TC-containing images and two thirds background images);
- the background images cannot be within 48 hours from a cyclone image and 12 hours from another background image, considering the geographical domain.

As a result of our image metadata generator, we obtain 9507 cyclone metadata and 19014 background metadata. The coordinates (longitude and latitude) of the cyclone and background metadata are then rounded to the respective resolutions of the ERA5 and MERRA-2 datasets, which results in two batches of metadata. Finally, we perform an additional step to check that no duplicate is created during the coordinate rounding.

3.2.4 NXTensor software library

The production of the image sets was the opportunity to create a reusable software library called NXTensor. This library is written in the Python 3.7 programming language and automates the extraction of geospatialized data, stored in NetCDF format, in a distributed and parallelized way on a computer cluster scheduled by Torque/Maui. Indeed each channel of the images is produced by a task of the cluster (multitasking) and the extractions are performed in parallel (multiprocessing). The library ensures the determinism of the data extractions and it is reusable for other experiments than ours, because the parameters of the extractions are entirely configurable through yaml files. NXTensor takes as parameters the description files of the variables (path on the disk, naming conventions of the files, etc.), notably the period covered by the NetCDF files (e.g., ERA5 files are monthly while MERRA-2 files are daily), and the image metadata (date and location).

Figure 3 illustrates the step-by-step operation of NXTensor according to the UML2 activity diagram formalism, for the production of one of the channels of all the cyclone and background images. NXTensor starts by analyzing the image metadata to group them according to the period of the variable files to ensure that the files are only read once by distributed task. This analysis produces the block metadata, i.e., the set of data extractions to be performed by period. Then NXTensor submits as many tasks to the cluster as there are channels, the determinism is ensured by sharing the same block metadata between the different distributed tasks. Within each task, the block metadata is divided into batches that are processed by a pool of workers performing the extractions of data in parallel. Each worker produces a set of blocks that are combined at the end by concatenation to form one of the channels of all the images. A special task is responsible for assembling the channels of the images in order to produce the 3D image tensor as mentioned above. For information, the elapsed time to extract a channel for 28,521 images is about six minutes when the computations are carried out on the CPU cluster of the Institut Pierre-Simon Laplace (IPSL), using eight nodes (15 Go RAM and 15 cores AMD Opteron™ 6378 at 2.4 GHz). The channel assembly task takes about one minute. The CPU time was 135 minutes.

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

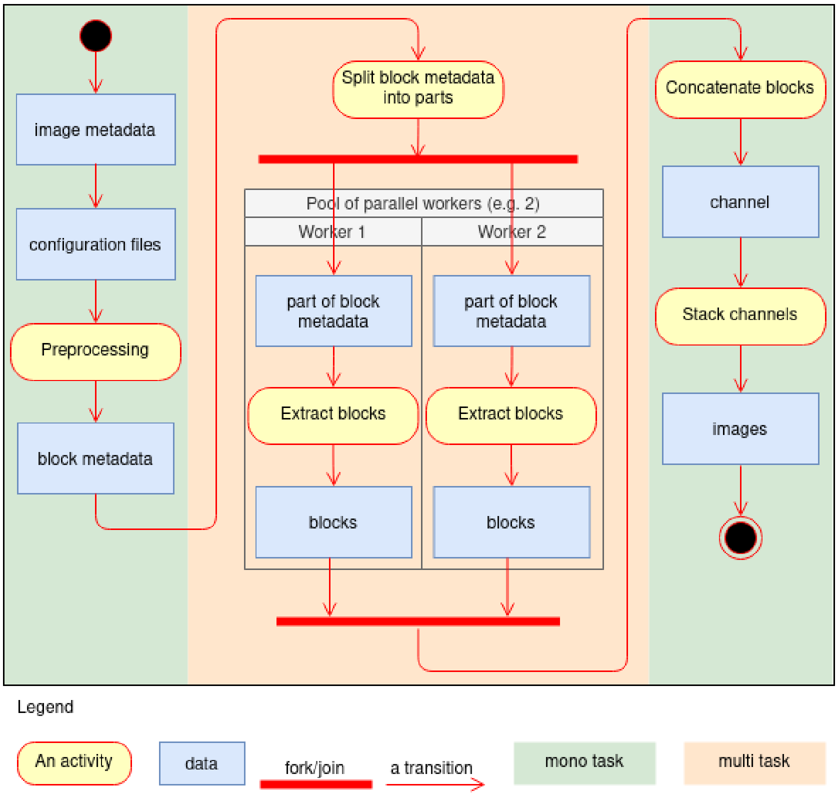

Figure 3. UML2 activity diagram of the image extractions using the NXTensor library.

3.2.5 Missing values issue

When generating images from the MERRA-2 data, we found that some of them had missing values (NaN), especially from the winds at 850 hPa. We decided to remove the metadata that resulted in incomplete images, for both the MERRA-2 and ERA5 180 batches, so that the batches of metadata are still identical. This resulted in the removal of 1,567 of them. Thus the number of cyclone metadata is 8,974 and the number of background metadata is 17,980, which gives a total of 26,954. With this final screening, we could then proceed to the extraction of the images.

3.2.6 Image interpolation

Previously we have detailed the automatic production chain of constant-shape images to satisfy the constraints of the CNN. 185 However, as mentioned above, the ERA5 and MERRA-2 reanalyses do not have the same spatial resolution (0.25° versus 0.5°). In order for the images to represent a constant domain size, and thus include cyclone of the same size as a fraction of the image domain size, we extract native images of 16×16 px for MERRA-2 and 32×32 px for ERA5 as described in Table 3. We then symmetrize the MERRA-2 native image set at a resolution of 0.5°×0.5° by linear interpolation to obtain the MERRA-2

Table 3. Properties of the image sets. MERRA-2 native is used to construct the other two MERRA-2 image sets but is not used as input to the CNN.

Image set	Size (in pixel)	Resolution (in °)
ERA5 native	32×32	0.25×0.25
ERA5 16px@0.5	16×16	0.5×0.5
MERRA-2 native	16×16	0.5×0.6
MERRA-2 16px@0.5	16×16	0.5×0.5
MERRA-2 32px@0.25	32×32	0.25×0.25

16px@0.5 image set. To resolve the difference in resolution and to study the sensitivity of the CNN to the different datasets, we further transform by linear interpolation one of the image sets to the properties of the other set (image resolution and size). Thus, we have two pairs of two image sets with similar properties: on the one hand ERA5 native and MERRA-2 32px@0.25 and on the other hand ERA5 16px@0.5 and MERRA-2 16px@0.5 (Table 3).

Figures 4 and 5 illustrate the representations of the channels of a cyclone and a background image, respectively, for the five image sets, at a same localization but for two different dates. It can be verified visually that the domain and pattern sizes of the images are independent of the choice of resolution. Finally, the input layer of the CNN is adapted dynamically to the size of the images during its instantiation, at the training phase which is described in the next section.

3.3 Model training

We performed our model training experiments on HAL, a Dell GPU cluster available at the IPSL. Each of HAL computing node is composed of two 2.6 Ghz Intel® Xeon® with four cores and two Nvidia® RTX® 2080 Ti 11 Go GPU cards but only one card was used for our training experiments. On the software side, the model is implemented in Python 3.8, using the Keras 2.3.0 library which is a layer build on top of the Tensorflow 2.2.0 library, making it simpler to use. 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 epoch without improving the training metric so the overfitting is prevented. The second callback always saves the weights of the model giving the best score of the training metric. As the number of epochs varies from one training to another (30 to 70), the training time also varies: between one and three minutes, knowing that one epoch takes less than one second of computation. Since training times are relatively short on our GPU cluster, we performed 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). The obtained values, described in the Table A3 in Appendix, are used for all experiments to avoid attributing the variability of the studied metrics to hyperparameter changes. These metrics and the methods for evaluating them are the subject of the next section.

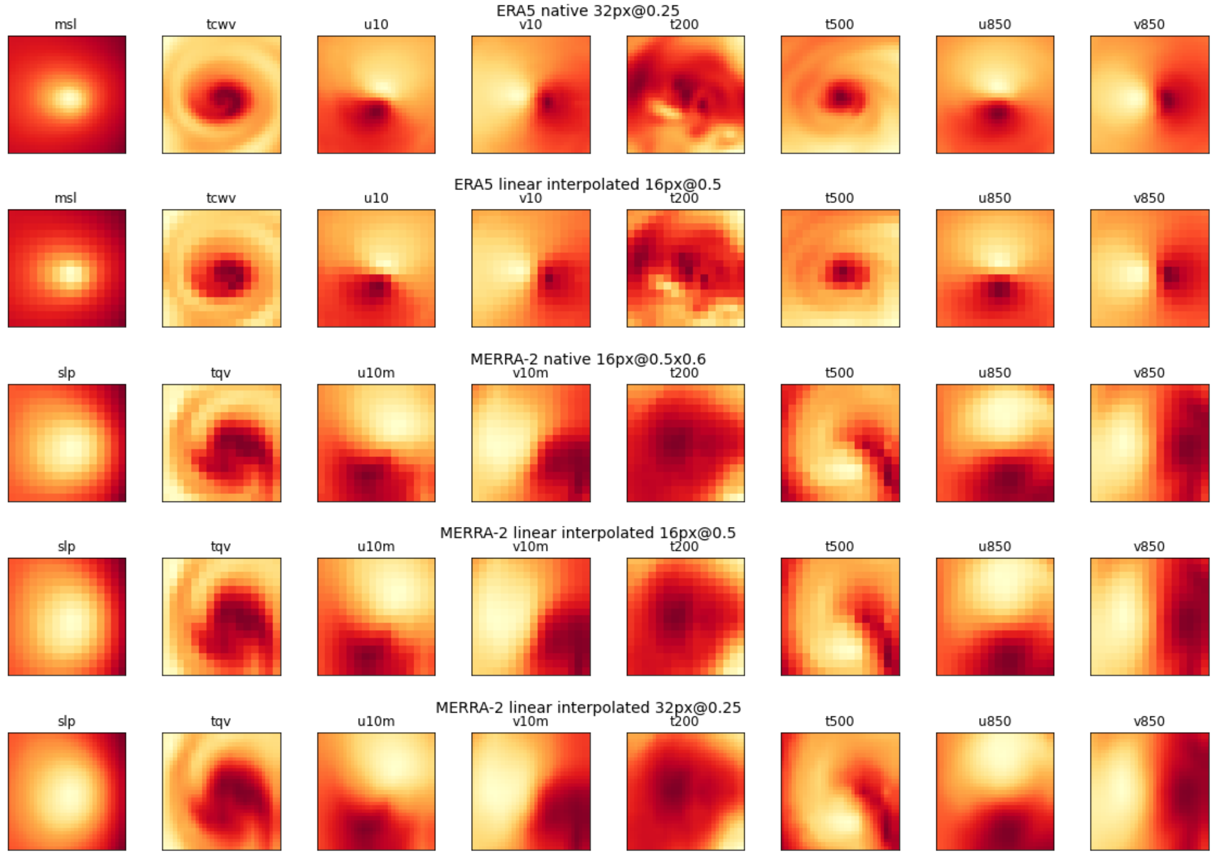

Figure 4. Channels (left to right) of the cyclone image on 22 August 1987, 00:00 UTC centered on 35.5°N, 43.125°W. The different rows show the native and interpolated images from ERA5 and MERRA-2 as per the labels.

3.4 Evaluation of metrics

In our study, we used three classical metrics to measure the performance of our binary classification model: accuracy, the Area Under the receiver operating Characteristic (AUC) and the Area Under the Precision-Recall Curve (AUPRC). The equations of the binary classification metrics are given in Appendix B1. The accuracy measures the rate of good predictions of a model. It is an easy metric to interpret, but it depends on the decision threshold for which the value of a probability is associated with one class rather than the other. It was criticized in particular by Provost et al. (1997) and Ling et al. (2003) and we discuss it further in section 4.1. The AUC measures the power of a model to discriminate the two classes for a variety of decision threshold values (recall versus false alarm ratio), while the AUPRC measures the ability of a model to identify all occurrences of a class (recall) while minimizing prediction errors (precision). AUC and AUPRC are much more interesting because they are integrated on the decision threshold values.

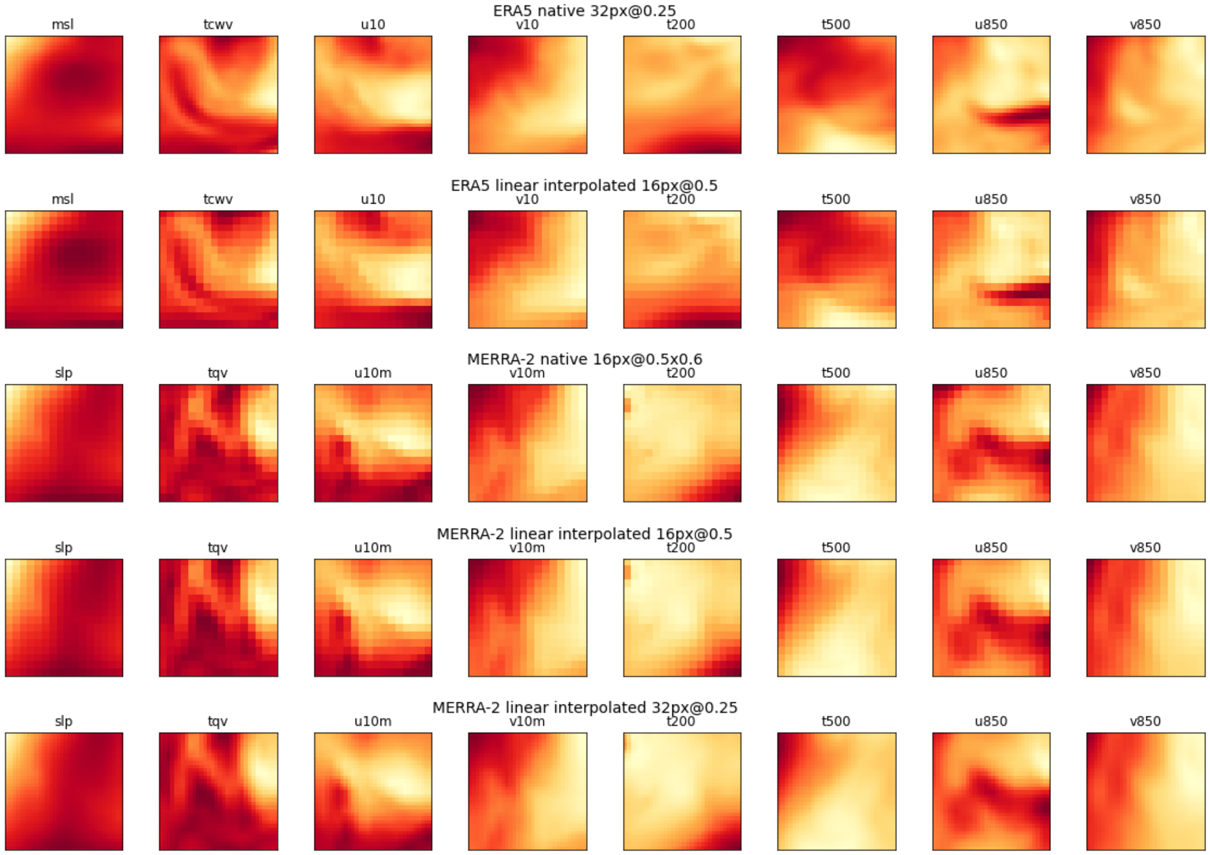

Figure 5. Same as Fig. 4 but for the background image on 6 August 1987, 18:00 UTC centered on 35.5°N, 43.125°W.

The hold-out validation is the classic method for evaluating the metrics. It consists in shuffling the dataset and randomly partitioning the dataset into two asymmetric sets (typically 80/20 or 70/30 as in our study). The model is trained on the larger of the two sets and its performance is computed on the other. This method is simple, but the performances obtained are dependent on the composition of the datasets. The more dissimilar the datasets, the more the performances will differ according to the partitioning.

Cross-validation is one way to solve this problem by evaluating the metrics on all or part of the data partitioning combinations. For our study, we have chosen the k-fold method because it is one of the non-exhaustive cross-validation methods whose computational cost remains reasonable, with k equal to ten (a common value found in the literature). It consists in randomly dividing the shuffled data into k sets called folds, each of the folds obtained is used successively to evaluate the metrics and the others to train the model. Thus the different measurements of the metrics give us access to the calculation of their uncertainty. However, ten measurements are not sufficient to obtain an accurate measurement. Iterative cross-validation is a good choice, for a reasonable computational cost, compared to exhaustive cross-validation methods (e.g., leave-p-out cross validation). It consists in running independently several times the cross-validation (twenty in our experiments). The expected value and the uncertainty of the metrics are calculated according to the central limit theorem, with the normality of the distribution of the means of the measurements computed at each iteration verified by the Shapiro-Wilk statistical test.

However the principle of random partitioning of the data may cause a bias for our application. Indeed the images coming from a time series of tracks from the same cyclone can be found in both the training and test datasets. In order to avoid this problem, we split the data by sampling the years randomly, approaching as much as possible the ratio required for the hold-out validation and balancing the folds as much as possible for the cross validation. For the iterative cross validation, the partitioning combinations are calculated in advance in order to guarantee the uniqueness of their composition.

Moreover, as the variables do not have the same scale of values, the image channels are standardized online, independently of each other, according to their mean and standard deviation calculated on the current training dataset, just before training the CNN.

Finally, for the comparison of the metric values, we chose to apply the Kruskal-Wallis statistical test 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.

4 Results and discussion

4.1 Accuracy and its threshold

Accuracy is a convenient measure, but according to Provost et al. (1997) and Ling et al. (2003), the class threshold makes it non-objective. In order to provide further evidence of this problem, we study the distribution of the classifier's predictions using the hold-out method. Rather than applying it on a single set of images, we identically partitioned the four sets of images and trained and tested the classifier for all possible combinations. Figure 6 shows plots of the distributions as log-scale histograms, colored according to the ground truth of the images. Then we computed the threshold for which the Youden's index is optimal (equation is described at Appendix B6). We seek to maximise J so as to obtain an optimal threshold for which the proportion of total misclassified results (false positives and false negatives) is minimal. By default, machine learning libraries set the threshold to 0.5; however, in our case, the optimal threshold, indicated in the title of each subplot of Fig. 6, is lower than 0.5, and for some combination of training and testing datasets, even much lower (e.g., ERA5/MERRA-2 combination in 16px@0.5). This reflects the fact that i) the image sets are not balanced and ii) the number of false negatives (orange color on the left side) is larger than the number of false positives (blue color on the right side) for this particular partitioning.

Our set of experiments shows that the choice of the threshold value depends on the partitioning, the source of the data and the relative importance given to false negatives and false positives. While accuracy is a less interesting metric than AUC and AUPRC, we decided to keep it, as a matter of information, and set its threshold to 0.5.

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

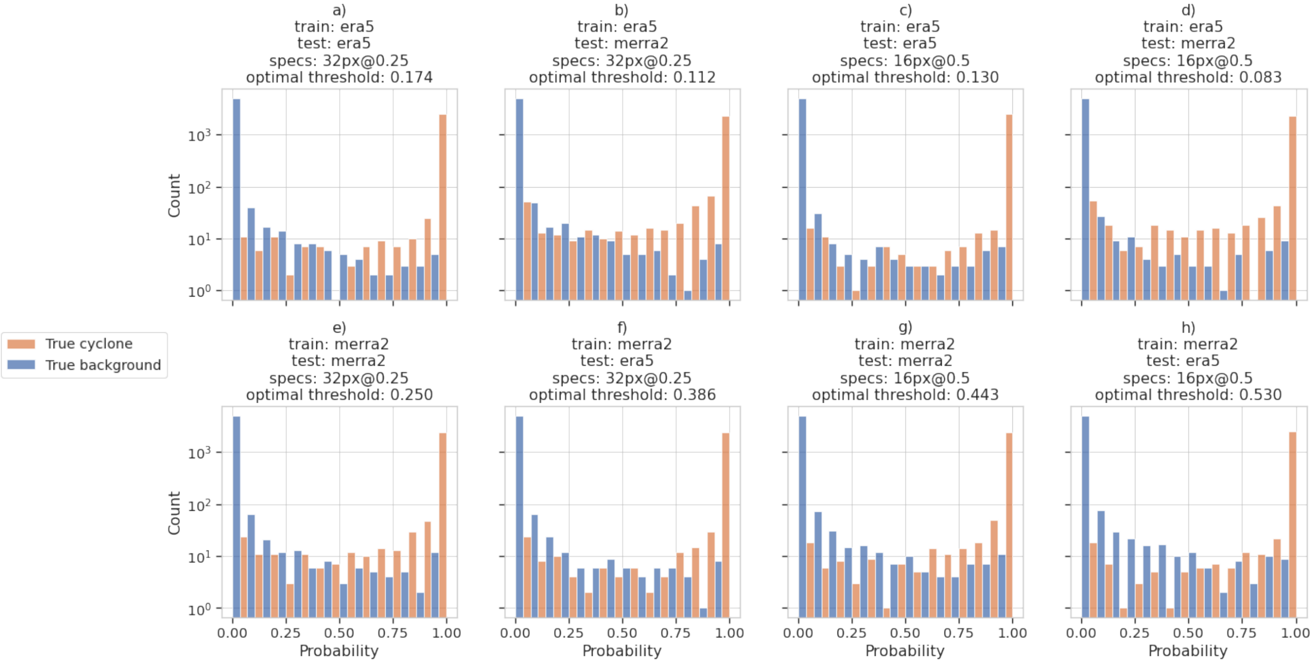

Figure 6. Histograms of the predicted probabilities for "true cyclone" images (orange bars) and background images (blue bars) for different combinations of training and test datasets and resolution. The optimal decision threshold is indicated in the title of each histogram. Note the logarithmic scale on the y-axis and that by construction there is twice as many background than cyclone images.

4.2 Metric comparisons

4.2.1 Inter-comparisons

In this section, we focus on the values of the CNN metrics obtained using the iterative cross-validation method on each of the image sets described in Table 3. Since the Shapiro-Wilk test shows that the distribution of the mean of the iteration values is normal for all metrics, under the central limit theorem, we computed the expectation and standard deviation of each of the metrics, given in Table A4. The values of the metrics are very high, over 0.9, however such values do not mean that a model is useful. Indeed the usefulness of a model is measured by the difference between its performance and that of models based on simple rules (e.g., stratified, most frequent class, prior class, uniform or constant) or a domain specific baseline. In our study, the classifier performs significantly better than simple models as shown in Fig. 7. By plotting the values of the metrics on Fig. 8, we can see that although very close, the performances of the CNN are grouped according to their original dataset (MERRA-2 and ERA5) and that the performances of these two groups seem significantly different. In order to have an objective confirmation, we chose to compare the values of the metrics using the Kruskal-Wallis test, as the distributions of the metric values are not mostly normal (see Table A4). Table A5 summarizes the pairwise comparison of the metric performances according to the image set used and confirms our interpretation of the Fig. This experiment tells us that the difference between

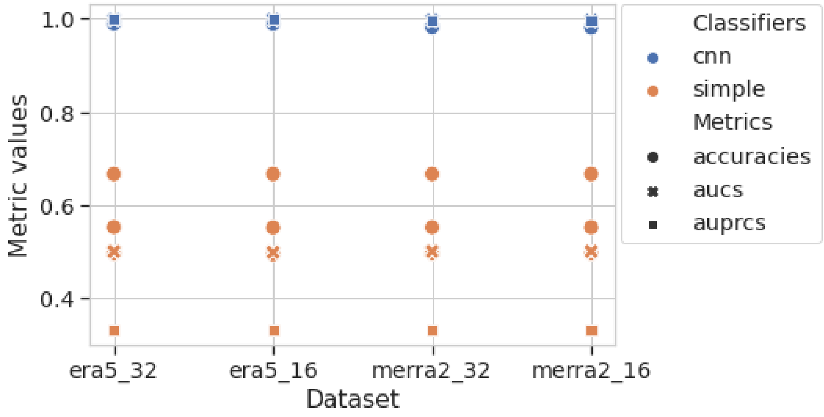

Figure 7. Metric values showing the performances of the CNN versus simple classifiers for the four datasets. The color of the symbols for the metric values computed from the simple models is orange while that of the CNN is blue. The marker shapes indicate the nature of the metric as per the legend.

the metric values computed from the same dataset, interpolated and not interpolated, can be attributed to randomness. Whereas the metrics computed from different dataset are quite distinct. So in our study, we can say that the interpolation does not impact the model performance and training with interpolated datasets has some meaning. At last, we observe that the values of the metrics from ERA5 are greater than those from MERRA-2.

4.2.2 Cross-comparisons

In this section we are interested in the values of the metrics of the CNN trained on one image set and tested on the other image set with the same properties (image resolution and size). In the same way as the previous experiment, we computed the expectation and standard deviation for each of the metrics, given in Table A6, and then compared the performance obtained previously (training and testing with the same image set) with these values (training and testing with a different image set). Figure 9 gives the graphical representation and Table A7 gives the result of the Kruskal-Wallis tests. This experiment shows us that regardless of the resolution experienced and the dataset used for model training, the metric values are statistically well distinct and the value of the metrics evaluated on the ERA5 dataset is greater than that evaluated on the MERRA2 dataset. 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.

4.3 Misclassified images

Following the comparison of the metrics, we took a closer look at the metadata of the images misclassified by the CNN. Table A8 in the Appendix summarizes the number of false alarms for each combination of training and testing datasets discussed in Section 4.1. We studied the metadata of the failed predictions that are common to all training/testing datasets so as to limit

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

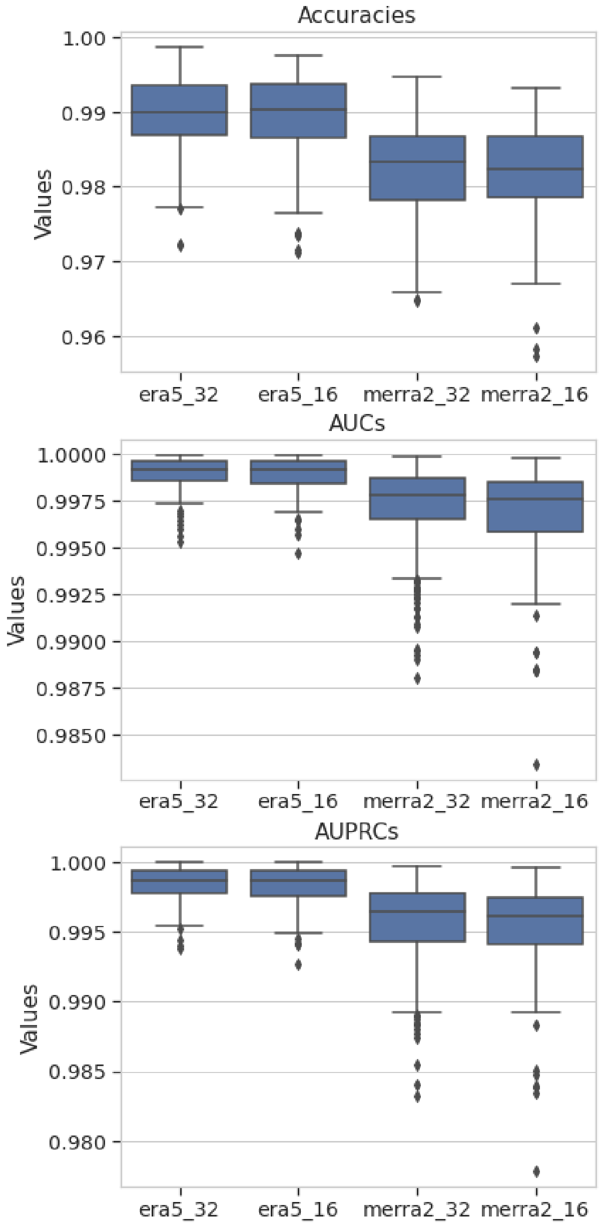

Figure 8. Box plots of the accuracy, the AUC and the AUPRC metric values for the CNN for the four image sets. The models are tested against the same image set as they are trained against (e.g., era5_32 means the CNN was trained and tested on ERA5 native).

295 the study to the most significant cases. We also contextualize the misclassified images in the HURDAT2 time series. There is a total of 15 false alarms in common, i.e. seven false positives (background images wrongly classified as cyclones) and eight false negatives (cyclone images wrongly classified as background). However, we found that the false negatives and the false

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

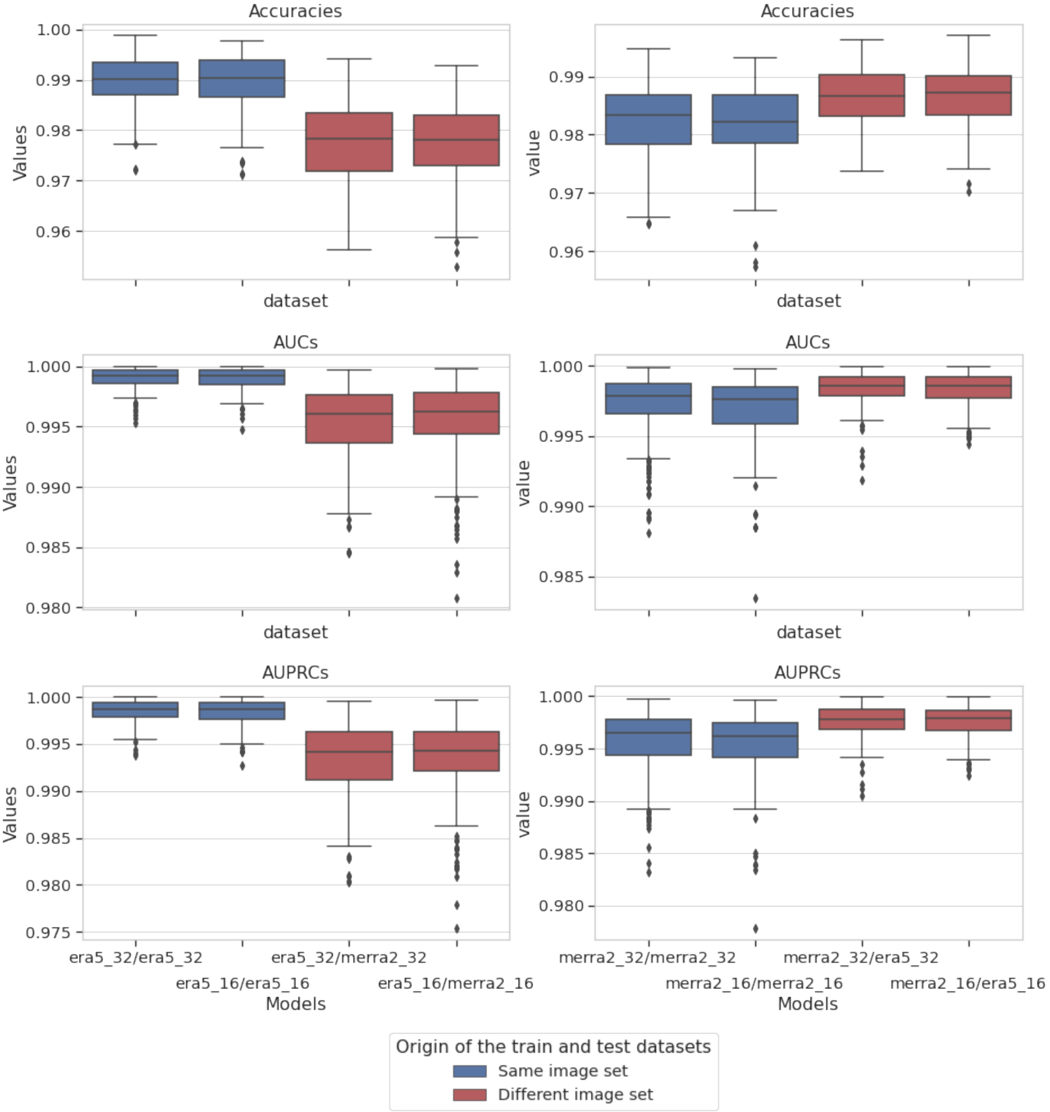

Figure 9. Box plots of the accuracy, the AUC and the AUPRC metric values for the CNN for different combinations of training and test image sets. In blue, the models are tested against the same image set that they are trained against. In purple, the models are tested against the other image set of the same resolution (e.g., era5_32/merra2_32 means the CNN was trained on ERA5 native and tested against MERRA-2 32px@0.25).

positives were generated from the tracks of the same cyclones. Thus, after removing the duplicates, there are only eight false alarms left in common, i.e. seven false positives and one false negative.

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

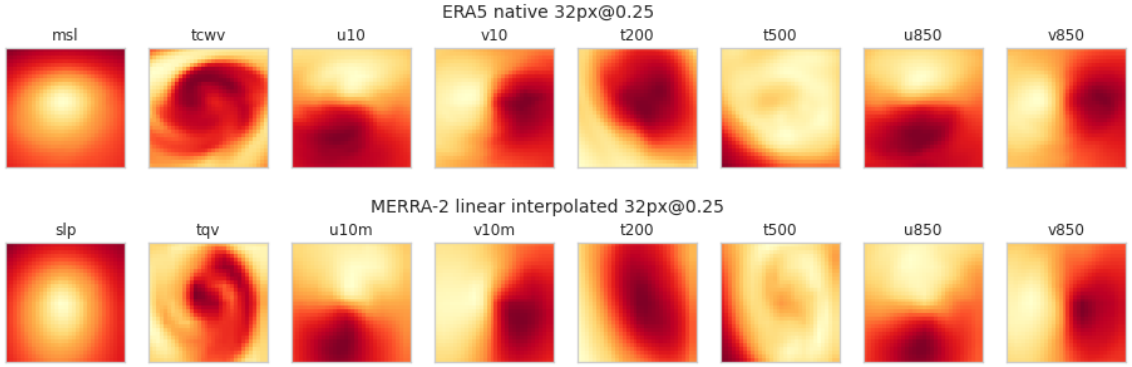

Figure 10. Channels of the image on 5 August 1990, 00:00 UTC centered on 38°N, 30.625°W, taken as an example of a wrongly classified image as a cyclone (false positive).

4.3.1 False Positives

First, we studied the false positives and have listed them in Table A9. For each image, we gave their HURDAT2 status (see Table A1) as well as the average probability given by the CNN for each dataset (mean prob column). For each of the false positives, we verified if there was a cyclone close in the past and in the future, by querying the HURDAT2 database and indicated the number of hours that separate them from a referenced cyclone (status HU or TS; respectively the past and future columns). What we can already observe is the high value of the mean probability and its low standard deviation: the CNN is wrong with high confidence for these images whatever the dataset used, which confirms the relevance of the failed predictions in common. Then, we notice that these images are temporally close to a TC by an average of 131 hours, approximately five days and a half (in the past or in the future). Thus we deduced that the false positives are essentially linked to transition states leading to a cyclone or to its dissipation. Figure 10 gives a graphical example of one of these false positives for the ERA5 and MERRA-2 image sets.

4.3.2 False negative

We have list the single example of false negative that the training/testing datasets have in common, in Table A10 and we give a graphical example in Fig. 11. The image refers to a cyclone which status is TS and we give its mean probability and standard deviation computed by the CNN for each dataset. We computed the lifetime of cyclonic activity near the geographical area of this image, as previously by querying the HURDAT2 database and indicated the number of hours that separate this image from the first track of a cyclone in the area. We observe that the probability is very low that means that the CNN is wrong with high confidence and the low standard deviation of this probability means that this false negative classification is relevant for all the combination of training/testing datasets. We also notice that this image is temporally close to a tropical depression, six hours in the future, suggesting that this false negative is essentially linked to the dissipation of a stationary cyclone.

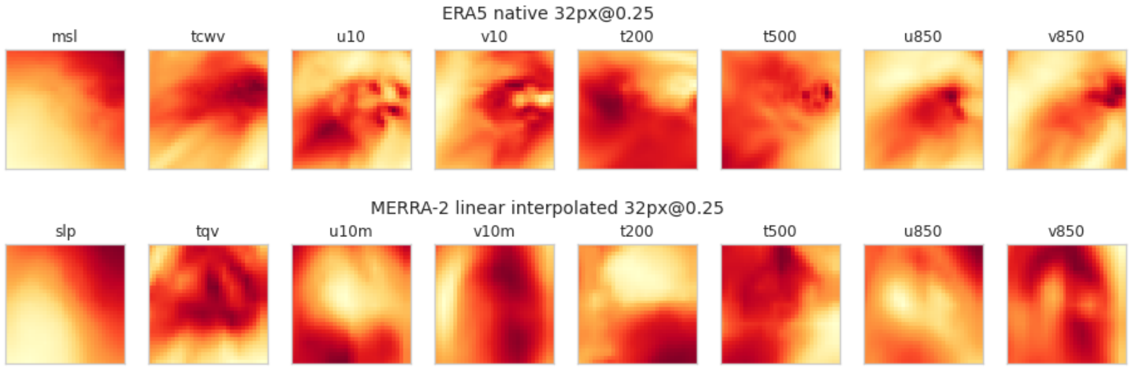

Figure 11. Channels of the image on 6 August 1990 06:00 UTC, centered on 27°N, 46.875°W, taken as an example of a wrongly classified image as background (false negative).

4.4 Potential future work

We have chosen a binary approach for the classification, but it is quite possible to design a classifier predicting the HURDAT2 status of the images presented to it (using nine neurons with the soft max activation function for the last layer of the CNN). However, training such a classifier would probably face an acute problem of image set imbalance. Indeed, four classes out of nine have a number of occurrences smaller than 400 (Fig.). To improve the situation, it would be possible to merge some classes between them (WV with DB and SD with SS) in order to mitigate the problem.

Our intercomparison experiments have shown that linear interpolation does not affect the performances of the classifier. However, there are other interpolation methods like bilinear, cubic, bicubic, nearest neighbor, etc. It would be interesting to verify if these interpolation methods have any effect on the performance of the classifier.

Some transfer learning experiments would also be interesting to conduct. For example, instead of training the CNN with randomly initialized weight values, training the CNN on one image set with weight values initialized with those of the CNN trained on the other image set with the same properties could improve the performance of the CNN.

Finally, pixel attribution experiments (saliency maps) should give us the importance of each variable, with hints on a possible reduction of their number or on the use of composite variables such as vorticity. These experiments could also give explanations on misclassified images. Occlusion - perturbation based methods like local surrogate (LIME; Ribeiro et al. 2016), Shapley values (SHAP; Lundberg and Lee 2017), and gradient based methods like Grad-CAM (Selvaraju et al., 2017) should be resourceful.

5 Conclusions

In this study, we have adapted and tested a CNN for the classification of images according to the presence or absence of tropical cyclones. The image sets for training and tests were built from the ERA5 and MERRA-2 reanalyses with labels derived from the HURDAT2 dataset. We have paid a lot of attention on the design of the background image set to make sure it samples

Table A1. HURDAT2 cyclone categories/status

Two-letter code	Storm status and Meaning
HU	Tropical cyclone of hurricane intensity (> 64 knots)
TS	Tropical cyclone of tropical storm intensity (34-63 knots)
TD	Tropical cyclone of tropical depression intensity (< 34 knots)
EX	Extratropical cyclone (of any intensity)
SD	Subtropical cyclone of subtropical depression intensity (< 34 knots)
SS	Subtropical cyclone of subtropical storm intensity (> 34 knots)
LO	A low that is neither a tropical cyclone, a subtropical cyclone, nor an extratropical cyclone (of any intensity)
WV	Tropical Wave (of any intensity)
DB	Disturbance (of any intensity)

similar location and time to the TC-containing image. We have assessed the performance of the CNN using accuracy but also the more objective AUC and AUPRC metrics. We have shown that failed classifications may be explained by the meteorological context. In particular false positives often represent a situation with cyclonic activity but not yet classified as TC by HURDAT2. It should be relatively easy to diagnose those situations if the TC are tracked in time rather than dealt with as a set of separate independent images as it is the case in this study. We have further shown that interpolation (from 0.5° to 0.25° or from 0.25° to 0.5°) does not impact the performance of the CNN. 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. This study paves the way for automatic detection of TC in climate simulations without the need to retrain the CNN for each new climate model or climate model resolution.

Code and data availability. The image sets computed from ERA5 and MERRA-2, their metadata, HURDAT2 data and the code used in this work (experiments and NXTensor) are all available at this address: https://doi.org/10.5281/zenodo.6453070 (DOI: 10.5281/zenodo.6453070). The code is open source and distributed under the CeCILL-2.1 license. More information about HURDAT2, ERA5 and MERRA-2, including how to download them, is available from https://www.nhc.noaa.gov/data/#hurdat, https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era5, https://gmao.gsfc.nasa.gov/reanalysis/MERRA-2/, respectively.

Appendix A: Tables

Appendix B: Equations

TP, TN, FP, FN stand for True Positives, True Negatives, False Positives, and False Negatives, respectively.

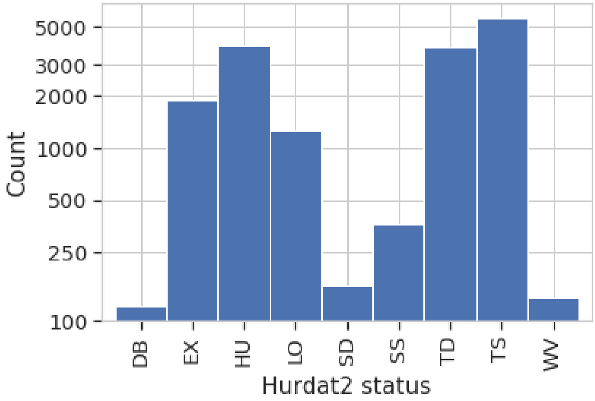

Figure A1. Distribution of the cyclone categories/status computed over the period 1980-2019

Table A2. Data set variables.

Variable	ERA5 attribute name	MERRA-2 attribute name
sea level pressure	msl	spl
precipitable water vapor	tcwv	tqv
northward wind at 10 meters	v10	v10m
northward wind at 850 hPa	v850	v850
eastward wind at 10 meters	u10	u10m
eastward wind at 850 hPa	u850	u850
temperature at 200 hPa	t200	t200
temperature at 500 hPa	t500	t500

B1 Binary classification metrics

The binary classification metrics used in this study are defined as:

360 $$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{B1}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{B2}$$

$$\text{Recall or Sensitivity} = \frac{TP}{TP+FN} \tag{B3}$$

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

Table A3. Optimal hyperparameter values.

Hyperparameter	Value
Loss function	Binary cross-entropy
Training metric	Loss computed on test set
Maximum number of epoch	100
Early stopping number of epoch	10
Batch size	256
Optimizer	Adam
Learning rate	0.0001

Table A4. The estimation of the values of the metrics based on iterative cross-validation. The train and test datasets come from the same image set (inter-comparison). The column "Shapiro p on means" refers to the p-value of the Shapiro-Wilk test computed on the mean of each iteration, whereas "Shapiro p on all" refers to the p-value computed on all the values of the metric.

Metric	Train dataset	Test dataset	Estimated mean	Estimated std	Shapiro p on means	Shapiro p on all
Accuracy	ERA5 32px@0.25	same	0.989748	0.002292	0.905230	1.363663e-04
	ERA5 16px@0.5	same	0.989547	0.002255	0.991043	9.576093e-08
	MERRA-2 32px@0.25	same	0.982276	0.002836	0.269320	2.560784e-04
	MERRA-2 16px@0.5	same	0.981858	0.002927	0.902361	3.120732e-06
AUC	ERA5 32px@0.25	same	0.998989	0.000643	0.088747	8.914557e-12
	ERA5 16px@0.5	same	0.998936	0.000638	0.506771	2.956014e-11
	MERRA-2 32px@0.25	same	0.997114	0.001140	0.017086	3.536823e-14
	MERRA-2 16px@0.5	same	0.996904	0.001107	0.239811	7.852716e-14
AUPRC	ERA5 32px@0.25	same	0.998430	0.000811	0.046393	2.159975e-09
	ERA5 16px@0.5	same	0.998374	0.000796	0.359663	4.650617e-11
	MERRA-2 32px@0.25	same	0.995573	0.001183	0.274620	3.556242e-12
	MERRA-2 16px@0.5	same	0.995337	0.001319	0.769673	3.934050e-13

$$\text{False Alarm} = \frac{FP}{FP + TN} \tag{B4}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{B5}$$

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

Table A5. Comparisons of metric values of models taken two by two (inter-comparisons). The column "Kruskal p-value" refers to the p-value of the Kruskal-Wallis test computed on all the values of the metrics. The column "Comparable" indicates whether the null hypothesis is accepted for a significance level of 1 % (α).

Metric	Train/test dataset	Train/test dataset	Kruskal p-value	Comparable
Accuracy	ERA5 32px@0.25/same	ERA5 16px@0.5/same	9.173333e-01	True
	ERA5 32px@0.25/same	MERRA-2 32px@0.25/same	8.451539e-31	False
	ERA5 32px@0.25/same	MERRA-2 16px@0.5/same	2.367641e-33	False
	ERA5 16px@0.5/same	MERRA-2 32px@0.25/same	9.721777e-29	False
	ERA5 16px@0.5/same	MERRA-2 16px@0.5/same	9.251784e-31	False
	MERRA-2 32px@0.25/same	MERRA-2 16px@0.5/same	5.506329e-01	True
AUC	ERA5 32px@0.25/same	ERA5 16px@0.5/same	5.243858e-01	True
	ERA5 32px@0.25/same	MERRA-2 32px@0.25/same	5.901992e-26	False
	ERA5 32px@0.25/same	MERRA-2 16px@0.5/same	9.964538e-32	False
	ERA5 16px@0.5/same	MERRA-2 32px@0.25/same	5.982768e-24	False
	ERA5 16px@0.5/same	MERRA-2 16px@0.5/same	6.935282e-30	False
	MERRA-2 32px@0.25/same	MERRA-2 16px@0.5/same	1.174539e-01	True
AUPRC	ERA5 32px@0.25/same	ERA5 16px@0.5/same	7.391314e-01	True
	ERA5 32px@0.25/same	MERRA-2 32px@0.25/same	2.082609e-30	False
	ERA5 32px@0.25/same	MERRA-2 16px@0.5/same	2.401478e-35	False
	ERA5 16px@0.5/same	MERRA-2 32px@0.25/same	1.952581e-29	False
	ERA5 16px@0.5/same	MERRA-2 16px@0.5/same	3.139588e-34	False
	MERRA-2 32px@0.25/same	MERRA-2 16px@0.5/same	2.450381e-01	True

365 B2 Youden's index

The Youden's index is defined as:

$$J = \text{sensitivity} + \text{specificity} - 1 = \frac{TP}{TP+FN} + \frac{TN}{TN+FP} - 1 \qquad \text{(B6)}$$

Author contributions. SG designed the study, including the data selection and cross-validation evaluation procedures, developed all codes used in this study, carried out all experiments, and performed the analysis. OB contributed to the design and interpretation of the results. SG 370 and OB both contributed to the writing of the manuscript.

Competing interests. The authors declare no competing interests.

Table A6. The estimation of the values of the metrics based on iterative cross-validation. The train and test datasets come from different image set but have the same image resolution and size (cross-comparisons). The column "Shapiro p on means" refers to the p-value of the Shapiro-Wilk test computed on the mean of each iteration, whereas "Shapiro p on all" refers to the p-value computed on all the values of the metric.

Metric	Train dataset	Test dataset	Estimated mean	Estimated std	Shapiro p on means	Shapiro p on all
Accuracy	ERA5 32px@0.25	MERRA-2 32px@0.25	0.977677	0.003182	0.226578	0.017054
	ERA5 16px@0.5	MERRA-2 16px@0.5	0.977721	0.002321	0.786720	0.013096
	MERRA-2 32px@0.25	ERA5 32px@0.25	0.986523	0.002604	0.073408	0.193609
	MERRA-2 16px@0.5	ERA5 16px@0.5	0.986655	0.002943	0.540750	0.019739
AUC	ERA5 32px@0.25	MERRA-2 32px@0.25	0.995235	0.001789	0.762134	5.846574e-09
	ERA5 16px@0.5	MERRA-2 16px@0.5	0.995413	0.001208	0.871147	2.561682e-12
	MERRA-2 32px@0.25	ERA5 32px@0.25	0.998357	0.000803	0.719871	1.581214e-12
	MERRA-2 16px@0.5	ERA5 16px@0.5	0.998323	0.000929	0.215594	5.414141e-09
AUPRC	ERA5 32px@0.25	MERRA-2 32px@0.25	0.993296	0.001898	0.436782	7.346610e-08
	ERA5 16px@0.5	MERRA-2 16px@0.5	0.993437	0.001331	0.905712	2.634040e-11
	MERRA-2 32px@0.25	ERA5 32px@0.25	0.997537	0.000852	0.825225	2.476906e-09
	MERRA-2 16px@0.5	ERA5 16px@0.5	0.997549	0.001092	0.694913	1.416567e-07

Acknowledgements. This study benefited from the ESPRI computing and data centre (https://mesocentre.ipsl.fr) which is supported by CNRS, Sorbonne Université, Ecole Polytechnique and CNES as well as through national and international grants. The authors acknowledge funding from the French state aid managed by the ANR under the "Investissements d'avenir" programme with the reference ANR-11-IDEX-0004-17-EURE-0006 and from the European Union's Horizon 2020 research and innovation program under grant agreement 101003469 for the "eXtreme events: Artificial Intelligence for Detection and Attribution" (XAIDA) project.

Table A7. The values of the metrics from models trained and tested on the same image set, compared to those from models trained on one image set and tested on the other (cross-comparison). The column "Kruskal p-value" refers to the p-value of the Kruskal-Wallis test computed on all the values of the metrics. The column "Comparable" indicates whether the null hypothesis is accepted for a significance level of 1 % (α).

Metric	Train/test dataset	Train/test dataset	Kruskal p-value	Comparable
Accuracy	ERA5 32px@0.25/same	ERA5 32px@0.25/MERRA-2 32px@0.25	4.857791e-47	False
	ERA5 16px@0.5/same	ERA5 16px@0.5/MERRA-2 16px@0.5	1.005262e-44	False
	MERRA-2 32px@0.25/same	MERRA-2 32px@0.25/ERA5 32px@0.25	1.825678e-11	False
	MERRA-2 16px@0.5/same	MERRA-2 16px@0.5/ERA5 16px@0.5	1.434371e-14	False
AUC	ERA5 32px@0.25/same	ERA5 32px@0.25/MERRA-2 32px@0.25	4.011721e-46	False
	ERA5 16px@0.5/same	ERA5 16px@0.5/MERRA-2 16px@0.5	2.698731e-43	False
	MERRA-2 32px@0.25/same	MERRA-2 32px@0.25/ERA5 32px@0.25	7.645103e-09	False
	MERRA-2 16px@0.5/same	MERRA-2 16px@0.5/ERA5 16px@0.5	2.074129e-12	False
AUPRC	ERA5 32px@0.25/same	ERA5 32px@0.25/MERRA-2 32px@0.25	1.640760e-48	False
	ERA5 16px@0.5/same	ERA5 16px@0.5/MERRA-2 16px@0.5	3.619129e-47	False
	MERRA-2 32px@0.25/same	MERRA-2 32px@0.25/ERA5 32px@0.25	3.864565e-12	False
	MERRA-2 16px@0.5/same	MERRA-2 16px@0.5/ERA5 16px@0.5	1.332830e-16	False

Table A8. Statistics of failed predictions by combinations of training/testing datasets.

Training dataset	Test dataset	Specs	Total failed	False negatives	False positives
ERA5	ERA5	32px@0.25	68 (0.88 %)	44	24
ERA5	MERRA-2	32px@0.25	156 (2.02 %)	125	31
ERA5	ERA5	16px@0.5	73 (0.94 %)	46	27
ERA5	MERRA-2	16px@0.5	155 (2.00 %)	128	27
MERRA-2	MERRA-2	32px@0.25	110 (1.42 %)	73	37
MERRA-2	ERA5	32px@0.25	93 (1.20 %)	58	35
MERRA-2	MERRA-2	16px@0.5	100 (1.29 %)	52	48
MERRA-2	ERA5	16px@0.5	90 (1.16 %)	40	50

References

Bosler, P. A., Roesler, E. L., Taylor, M. A., and Mundt, M. R.: Stride Search: a general algorithm for storm detection in high-resolution climate data, Geosci. Model Dev., 9, 1383–1398, https://doi.org/10.5194/gmd-9-1383-2016, 2016.

Chan, J. C. L.: Comment on "Changes in tropical cyclone number, duration, and intensity in a warming environment", Science, 311, 1713, https://doi.org/10.1126/science.1121522, 2006.

Table A9. Background images wrongly classified as TC-containing images (false positives) for all combinations of training/testing datasets. For each image we also indicate the status of the image according to HURDAT2 if present in the database ("None" if not present), the probability of the classification (with its standard deviation across the combinations of training/testing datasets), and the temporal distance to a cyclone in the past and in the future (with the status of the cyclone).

#index	Status	Mean prob	Past (hours)	Future (hours)	HURDAT2 id
207	SD	0.9905±0.0205	8250 (HU)	90 (TS)	AL061990
3149	None	0.8819±0.0544	1308 (TS)	354 (TS)	AL122008
4163	WV	0.9919±0.0150	174 (TS)	228 (TS)	AL092012
6048	None	0.8240±0.1111	132 (TS)	54 (TS)	AL071998
6059	EX	0.9963±0.0047	228 (TS)	96 (TS)	AL132018
6295	None	0.8918±0.1072	162 (HU)	60 (HU)	AL132003
6836	None	0.9216±0.0542	168 (TS)	90 (TS)	AL162000

Table A10. The single TC-contaning image wrongly classified as background (false negative) for all combinations of training/testing datasets. The status of the image according to HURDAT2, the probability of the classification (with its standard deviation across the combinations of training/testing datasets) are indicated. The columns past and future reflect the cyclonic activity in the geographical area of the image, i.e. the temporal distance to the first and the last tracks of the cyclone (and their HURDAT2 status).

#index	Status	Mean prob	Past (hours)	Future (hours)	HURDAT2 id
290	TS	0.0855±0.0665	72 (bckgrd)	6 (TD)	AL162000

Ebert-Uphoff, I. and Hilburn, K.: Evaluation, tuning, and interpretation of neural networks for working with images in meteorological applications, Bull. Am. Meteorol. Soc., 101, E2149–E2170, https://doi.org/10.1175/BAMS-D-20-0097.1, 2020.

Emanuel, K.: Increasing destructiveness of tropical cyclones over the past 30 years, Nature, 436, 686–688, https://doi.org/10.1038/nature03906, 2005.

Gelaro, R., McCarty, W., Suárez, M. J., Todling, R., Molod, A., Takacs, L., Randles, C. A., Darmenov, A., Bosilovich, M. G., Reichle, R., Wargan, K., Coy, L., Cullather, R., Draper, C., Akella, S., Buchard, V., Conaty, A., da Silva, A. M., Gu, W., Kim, G.-K., Koster, R., Lucchesi, R., Merkova, D., Nielsen, J. E., Partyka, G., Pawson, S., Putman, W., Rienecker, M., Schubert, S. D., Sienkiewicz, M., and Zhao, B.: The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2), J. Clim., 30, 5419–5454, https://doi.org/10.1175/JCLI-D-16-0758.1, 2017.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., de Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.-N.: The ERA5 global reanalysis, Q. J. R. Meteorol. Soc., 146, 1999–2049, https://doi.org/10.1002/qj.3803, 2020.

Hong, S., Kim, S., Joh, M., and kwang Song, S.: GlobeNet: Convolutional neural networks for typhoon eye tracking from remote sensing imagery, in: 7th International Workshop on Climate Informatics, edited by Lyubchich, V., Oza, N. C., Rhines, A., and Szekely, E., vol. NCAR Technical Notes, NCAR/TN536+PROC, pp. 69–72, National Center for Atmospheric Research, 2017.

Horn, M., Walsh, K., Zhao, M., Camargo, S. J., Scoccimarro, E., Murakami, H., Wang, H., Ballinger, A., Kumar, A., Shaevitz, D. A., Jonas, J. A., and Oouchi, K.: Tracking scheme dependence of simulated tropical cyclone response to idealized climate simulations, J. Clim., 27, 9197–9213, 2014.

IPCC: 2021: Summary for Policymakers. In: Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change, Masson-Delmotte, V., P. Zhai, A. Pirani, S. L. Connors, C. Péan, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M. I. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J.B.R. Matthews, T. K. Maycock, T. Waterfield, O. Yelekçi, R. Yu and B. Zhou (eds.), Cambridge University Press, 2021.

Jiaxiang, G., Shoshiro, M., Roberts, M. J., Haarsma, R., Putrasahan, D., Roberts, C. D., Scoccimarro, E., Terray, L., Vannière, B., and Vidale, P. L.: Influence of model resolution on bomb cyclones revealed by HighResMIP-PRIMAVERA simulations, Env. Res. Lett., 15, 084 001, https://doi.org/10.1088/1748-9326/ab88fa, 2020.

Kim, M., Park, M.-S., Im, J., Park, S., and Lee, M.-I.: Machine learning approaches for detecting tropical cyclone formation using satellite data, Remote Sensing, 11, 1195, https://doi.org/10.3390/rs11101195, 2019.

Knapp, K. R., Kruk, M. C., Levinson, D. H., Diamond, H. J., and Neumann, C. J.: The International Best Track Archive for Climate Stewardship (IBTrACS): Unifying tropical cyclone data, Bull. Am. Meteorol. Soc., 91, 363–376, https://doi.org/10.1175/2009BAMS2755.1, 2010.

Knutson, T. R., Sirutis, J. J., Zhao, M., Tuleya, R. E., Bender, M., Vecchi, G. A., Villarini, G., and Chavas, D.: Global projections of intense tropical cyclone activity for the late twenty-first century from dynamical downscaling of CMIP5/RCP4.5 scenarios, J. Clim., 28, 7203–7224, https://doi.org/10.1175/JCLI-D-15-0129.1, 2015.

Kossin, J., Emanuel, K., and Vecchi, G.: The poleward migration of the location of tropical cyclone maximum intensity, Nature, 509, 349–352, https://doi.org/10.1038/nature13278, 2014.

Kurth, T., Zhang, J., Satish, N., Racah, E., Mitliagkas, I., Patwary, M. M. A., Malas, T., Sundaram, N., Bhimji, W., Smorkalov, M., Deslippe, J., Shiryaev, M., Sridharan, S., Prabhat, and Dubey, P.: Deep Learning at 15PF: Supervised and semi-supervised classification for scientific data, in: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC ’17, Association for Computing Machinery, New York, NY, USA, https://doi.org/10.1145/3126908.3126916, 2017.

Landsea, C. W. and Franklin, J. L.: Atlantic hurricane database uncertainty and presentation of a new database format, Mon. Weather Rev., 141, 3576–3592, https://doi.org/10.1175/MWR-D-12-00254.1, 2013.

Landsea, C. W., Vecchi, G. A., Bengtsson, L., and Knutson, T. R.: Impact of duration thresholds on Atlantic tropical cyclone counts, J. Clim., 23, 2508–2519, https://doi.org/10.1175/2009JCLI3034.1, 2010.

Ling, C. X., Huang, J., and Zhang, H.: AUC: a better measure than accuracy in comparing learning algorithms, in: Proceedings of IJCAI’03, pp. 329–341, Springer, 2003.

Liu, Y., Racah, E., Prabhat, Correa, J., Khosrowshahi, A., Lavers, D., Kunkel, K., Wehner, M. F., and Collins, W. D.: Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets, CoRR, abs/1605.01156, http://arxiv.org/abs/1605.01156, 2016.

Lundberg, S. M. and Lee, S.-I.: A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems 30, edited by Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., pp. 4765–4774, Curran Associates, Inc., 2017.

Matsuoka, D., Nakano, M., Sugiyama, D., and Uchida, S.: Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model, Progress in Earth and Planetary Science, 5, 80, https://doi.org/10.1186/s40645-018-0245-y, 2018.

Park, M.-S., Kim, M., Lee, M.-I., Im, J., and Park, S.: Detection of tropical cyclone genesis via quantitative satellite ocean surface wind pattern and intensity analyses using decision trees, Remote Sens. Environ., 183, 205–214, https://doi.org/10.1016/j.rse.2016.06.006, 2016.

Prabhat, Kashinath, K., Mudigonda, M., Kim, S., Kapp-Schwoerer, L., Graubner, A., Karaismailoglu, E., von Kleist, L., Kurth, T., Greiner, A., Yang, K., Lewis, C., Chen, J., Lou, A., Chandran, S., Toms, B., Chapman, W., Dagon, K., Shields, C. A., O'Brien, T., Wehner, M., and Collins, W.: ClimateNet: an expert-labelled open dataset and deep learning architecture for enabling high-precision analyses of extreme weather, Geosci. Model Dev., 14, 107–124, https://doi.org/10.5194/gmd-14-107-2021, 2020.

Provost, F., Fawcett, T., and Kohavi, R.: The case against accuracy estimation for comparing induction algorithms, in: Proceedings of the Fifteenth International Conference on Machine Learning, pp. 445–453, Morgan Kaufmann, 1997.

Racah, E., Beckham, C., Maharaj, T., Kahou, S. E., Prabhat, and Pal, C.: Extreme weather: a large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events, in: NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 3405–3416, 2017.

Ribeiro, M. T., Singh, S., and Guestrin, C.: "Why should I trust you?": Explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, pp. 1135–1144, 2016.

Roberts, M. J., Camp, J., Seddon, J., Vidale, P. L., Hodges, K., Vannière, B., Mecking, J., Haarsma, R., Bellucci, A., Scoccimarro, E., Caron, L.-P., Chauvin, F., Terray, L., Valcke, S., Moine, M.-P., Putrasahan, D., Roberts, C. D., Senan, R., Zarzycki, C., Ullrich, P., Yamada, Y., Mizuta, R., Kodama, C., Fu, D., Zhang, Q., Danabasoglu, G., Rosenbloom, N., Wang, H., and Wu, L.: Projected future changes in tropical cyclones using the CMIP6 HighResMIP multimodel ensemble, Geophys. Res. Lett., 47, e2020GL088 662, https://doi.org/10.1029/2020GL088662, e2020GL088662 2020GL088662, 2020.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D.: Grad-CAM: Visual explanations from deep networks via gradient-based localization, in: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 618–626, 2017.

Singh, S., Singh, C., and Mitra, D.: Detection and tracking of tropical cyclone using NCEP-GFS model analysis and forecasts, J. Earth Syst. Sci., p. 15, https://doi.org/10.1007/s12040-021-01765-1, 2022.

Studholme, J., Fedorov, A. V., Gulev, S. K., Emanuel, K., and Hodges, K.: Poleward expansion of tropical cyclone latitudes in warming climates, Nature Geoscience, https://doi.org/10.1038/s41561-021-00859-1, 2021.

Vecchi, G. A., Delworth, T. L., Murakami, H., Underwood, S. D., Wittenberg, A. T., Zeng, F., Zhang, W., Baldwin, J. W., Bhatia, K. T., Cooke, W., He, J., Kapnick, S. B., Knutson, T. R., Villarini, G., van der Wiel, K., Anderson, W., Balaji, V., Chen, J., Dixon, K. W., Gudgel, R., Harris, L. M., Jia, L., Johnson, N. C., Lin, S.-J., Liu, M., Ng, C. H. J., Rosati, A., Smith, J. A., and Yang, X.: Tropical cyclone sensitivities to CO_2 doubling: roles of atmospheric resolution, synoptic variability and background climate changes, Clim. Dyn., 53, 5999—-6033, https://doi.org/10.1007/s00382-019-04913-y, 2019.

Walsh, K. J. E., Fiorino, M., Landsea, C. W., and McInnes, K. L.: Objectively determined resolution-dependent threshold criteria for the detection of tropical cyclones and reanalyses, J. Clim., 20, 2307–2314, 2007.

https://doi.org/10.5194/egusphere-2022-147
Preprint. Discussion started: 19 April 2022

Webster, P. J., Holland, G. J., Curry, J. A., and Chang, H.-R.: Changes in tropical cyclone number, duration, and intensity in a warming environment, Science, 309, 1844–1846, https://doi.org/10.1126/science.1116448, 2005.

Wu, L., Zhao, H., Wang, C., Cao, J., and Liang, J.: Understanding of the effect of climate change on tropical cyclone intensity: A Review, Adv. Atmos. Sci., pp. 205–221, https://doi.org/10.1007/s00376-021-1026-x, 2022.