Environmental Data Science (2025), 4: e36, 1–17
doi:10.1017/eds.2025.10014

CAMBRIDGE UNIVERSITY PRESS

METHODS PAPER

MoTiF: a self-supervised model for multi-source forecasting with application to tropical cyclones

Clément Dauvilliers[1] and Claire Monteleoni[1,2]

[1]ARCHES, INRIA, Paris, Île-de-France, France
[2]Department of Computer Science, University of Colorado, Boulder, CO, USA
Corresponding author: Clément Dauvilliers; Email: clement.dauvilliers@inria.fr

Received: 30 June 2025; **Revised:** 30 June 2025; **Accepted:** 30 June 2025

Keywords: deep learning; extreme weather; multi-sources; self-supervised training; tropical cyclones

Abstract

We present a deep learning architecture that reconstructs a source of data at given spatio-temporal coordinates using other sources. The model can be applied to multiple sources in a broad sense: the number of sources may vary between samples, the sources can differ in dimensionality and sizes, and cover distinct geographical areas at irregular time intervals. The network takes as input a set of sources that each include values (e.g., the pixels for two-dimensional sources), spatio-temporal coordinates, and source characteristics. The model is based on the Vision Transformer, but separately embeds the values and coordinates and uses the embedded coordinates as relative positional embedding in the computation of the attention. To limit the cost of computing the attention between many sources, we employ a multi-source factorized attention mechanism, introducing an anchor-points-based cross-source attention block. We name the architecture MoTiF (multi-source transformer via factorized attention). We present a self-supervised setting to train the network, in which one source chosen randomly is masked and the model is tasked to reconstruct it from the other sources. We test this self-supervised task on tropical cyclone (TC) remote-sensing images, ERA5 states, and best-track data. We show that the model is able to perform TC ERA5 fields and wind intensity forecasting from multiple sources, and that using more sources leads to an improvement in forecasting accuracy.

Impact Statement

This article presents a deep learning method for extreme weather forecasting designed to merge multiple sources of data. The sources can be of different natures (e.g., satellite images and weather station measurements) and cover different geographical areas and times. We show that by using more sources with a more general artificial intelligence model, one can, for example, improve the forecasting of tropical cyclones (TCs).

1. Introduction

In recent years, machine learning models have rapidly taken over the field of global weather forecasting. Fully data-driven models can now reach better average results on many metrics than the best physics-based ones, such as the High-resolution Integrated Forecasting System (IFS-HRES, Bi et al., 2023; Lam et al., 2023; Nguyen et al., 2024; Zhong et al., 2024). Still, the overall ability of Machine Learning (ML)

This research article was awarded Open Data and Open Materials badges for transparent practices. See the Data Availability Statement for details.

models to forecast extreme weather accurately is less clear. On the one hand, artificial intelligence (AI) models have shown improvements upon IFS-HRES in extreme weather-related tasks, such as forecasting TC tracks, atmospheric rivers (Bi et al., 2023; Kochkov et al., 2023; Lam et al., 2023; Price et al., 2024), as well as heat, cold, and wind extremes (Bi et al., 2023; Lam et al., 2023; Olivetti and Messori, 2024). On the other hand, many of the general weather forecasting ML models are trained on European Center for Medium-range Weather Forecasting's Reanalysis v5 (ERA5, Hersbach et al., 2020), which is limited in its capacity to represent extreme weather events. Even disregarding modeling errors, its 0.25° spatial resolution is insufficient to represent many extreme phenomena accurately, such as convective activity (Taszarek et al., 2021). In the case of TCs and extra-TCs (ETCs), ERA5 tends to underestimate the highest wind speeds, which happen in a small region in the eyewall (Bié and de Camargo, 2023; Chen et al., 2024). Besides, reanalysis and physics-based forecasts are limited in their real-time availability as they require a phase of data assimilation. Consequently, forecast updates are typically available only every 6 h, which is longer than the period over which some extreme phenomena can develop or evolve. For those reasons, extreme weather forecasting ML models would greatly benefit from being able to use multiple sources of data, including observations.

1.1. Contributions

Multi-source architecture. In this work, we present an architecture named MoTiF (multi-source transformer via factorized attention) designed to merge multiple sources of geospatial information, of different types (satellite imagery, reanalysis, best-track estimates), from different instruments (multiple observing frequencies and multiple satellites), covering misaligned geographical areas, and at irregular time intervals. The architecture also allows using a flexible number of sources as input, including within a batch during inference or training. MoTiF takes as input a set of sources that each include values (e.g., pixels for image-like sources), spatio-temporal coordinates, and source characteristics (such as the ground sampling distance for satellite sources). The model also supports partially missing sources. The task performed by the network is to predict the values of a requested source, using its coordinates and the values and coordinates of the other sources. The MoTiF architecture is based on the transformer (Vaswani et al., 2023) and Vision Transformer (ViT; Dosovitskiy et al., 2021), and uses three separate embeddings, for the sources' values (e.g., pixels for image-like sources), the spatio-temporal coordinates, and a conditioning vector, including characteristics of the sources and a landmask. The values contain valuable information, while the embedded coordinates serve as relative positional embedding in the attention computation (inspired by Shaw et al., 2018). To limit the memory cost of cross-source attention, we use a factorized attention system inspired by Couairon et al. (2024). As the sources have different shapes and are generally not aligned geographically, the usual factorized attention does not directly apply. For this reason, we introduce a specific anchor-points-based cross-attention layer. This layer limits the cost of the cross-source attention while allowing information to be exchanged between the sources.

Self-supervised framework. We wrap this architecture within a framework that leverages self-supervised learning (SSL), which means that no training labels are required, as the same data point can serve as both input and target. Given a set of available sources in a sample, one of the sources is fully masked, and the model is tasked to reconstruct it using the other available sources. In order to reconstruct the masked source, the model only has access to its spatio-temporal coordinates and information characterizing the source. This requires the model to understand the relationship between the masked source's coordinates and characteristics and those of all other sources. While this source reconstruction task is self-supervised, it has direct applications, contrary to pure pretext tasks such as in the masked auto-encoder setting (He et al., 2021), for example, inferring the wind fields of a cyclone from satellite images. Finally, the reconstruction task can be viewed differently depending on which source is masked: if the masked source is chronologically between two unmasked ones, then the task is temporal interpolation coupled with domain adaptation. If the masked source is chronologically last, the task becomes forecasting.

Application. We apply the MoTiF architecture and the self-supervised framework to the specific case of TCs and ETCs. We pretrain a model on a mix of satellite images and ERA5 surface fields, and then fine-tune it on different tasks, such as ERA5 fields forecasting, wind intensity forecasting, and satellite images reconstruction.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

1.2. Related work

1.2.1. Learning from multiple sources for weather forecasting

Using multiple sources as input for weather forecasting ML models has been studied before. Met-Net 3 (Andrychowicz et al., 2023) couples sparse weather stations' data with assimilated states to forecast a dense field of precipitation and surface variables. Using observations offers Met-Net the advantage of updating its forecast every 5 min while also limiting the errors linked to the assimilation process. However, Met-Net is limited to two types of sources and does not have the objective of combining a large number of sources. Aardvark (Vaughan et al., 2024) is a global weather forecasting data-driven model that combines many remote sensing and in situ sources to forecast ERA5 states. The model's ability to combine observations into a common latent space via a transformer-based architecture supports the choice of architecture made in this work. Nonetheless, Aardvark works on a constant global grid and maps off-the-grid data to the grid with SetConv layers (Gordon et al., 2020). While we employ the same principle of encoding the sources into a common latent space, we focus on nonglobal weather phenomena, for which the sources cover different geographical areas with varying grids. Finally, McNally et al. (2024) combine multiple sources of global observations, but output observations instead of ERA5 or any assimilated state, and use masked autoencoding (He et al., 2021) as a training task. While we use a similar training task in this work, we fully mask one of the sources instead of masking only regions of it. We detail the motivations in the following section.

1.2.2. SSL for geographical-temporal data

SSL consists of training on a so-called pretext task that allows the model to learn powerful representations of the input data. The model can then be fine-tuned for more specific downstream tasks using the learned representations. The advantage of SSL is that the pretext task can be designed not to require any labeled data, which often allows the use of a large amount of data with little to no human input. Two principles of SSL have mostly been explored in the context of temporal-geographical data. Contrastive learning methods have been used on remote-sensing data, including sources from multiple satellites with different resolutions (Ayush et al., 2021; Guo et al., 2024; Klemmer et al., 2024). This type of SSL trains a model to decide whether two inputs are transformations of a single sample or are two different samples. On the other hand, the masked autoencoding method masks areas of the input (He et al., 2021; Tong et al., n.d.) and tasks the model with reconstructing them. It has also shown success in the context of multi-source remote-sensing data (Cong et al., 2023; Reed et al., 2023; Zhang et al., 2024). The SSL task used in this work is similar to masked autoencoding in that the model is trained to reconstruct masked sources, with the difference that we fully mask one source instead of only masking areas of it. The goal of that change is twofold: first, to avoid having the model learn to inpaint masked patches from their borders instead of fetching the information from the other sources; and second, because reconstructing a source from simply its coordinates and the other sources can be a useful application on its own. Another difference with the aforementioned works is that they focus on collocated images, whereas the sources used here cover different geographical areas.

2. Methods

We first present the source reconstruction task used to train the model in a self-supervised manner. We then cover the specific network architecture designed to use multiple sources as input and output data.

2.1. Definition of a sample

We consider a set $\mathcal{S}$ of sources of data $\mathcal{S} = \{\mathcal{S}^1, \mathcal{S}^2, \ldots, \mathcal{S}^{|\mathcal{S}|}\}$. Each source $\mathcal{S}^i$ is associated with a source type $T(\mathcal{S}^i)$, which is defined by the nature of the information contained in the source. Examples of source types are infrared satellite imagery, passive microwave (PMW) imagery, assimilated state, station measurement, and so forth. Several sources can be of the same source type if they measure the same

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

variable(s) but with varying characteristics, for example, two PMW sensors on different satellites that observe nearly the same frequency but differ in ground sampling distance.

A sample $\boldsymbol{X}_n$ is defined as a set of elements from a subset $S = \{S^1, S^2, \ldots, S^K\}$ of $\mathcal{S}$. Thus, within a sample, either zero or one element for each source may be included.

A data point $\boldsymbol{X}_n^k$ from a source S^k in the sample of the index n is made up of multiple elements:

(a) The values $\boldsymbol{v}_n^k$, which can be zero-dimensional ($\boldsymbol{v}_n^k \in \mathbb{R}^{C^k}$, where C^k is the number of channels) or two-dimensional (2D) ($\boldsymbol{v}_n^k \in \mathbb{R}^{C^k \times H \times W}$, where H and W are, respectively, the height and width of the element). The values represent the data of interest in the source, for example, the values of the pixels for 2D sources. Since sources of the same source type by definition contain the same variables, they must have the same channels (and, therefore, the same number of channels C^k). Two data points from the same source can have different spatial shapes.

(b) The spatial coordinates $\boldsymbol{c}_n^k$, whose dimensions are the same as those of the values, except that there are exactly four channels: latitude, sin(longitude), cos(longitude), and a land–sea mask. The latitudes and longitudes are given for every point (that is, for every pixel in images), which theoretically allows the model to work with any geometry.

(c) The relative time delta $\delta_n^k \in \mathbb{R}$, which is computed as $\delta_n^k = \frac{t_{0,n} - t_n^k}{\delta_{max}}$, where $t_{0,n}$ is the absolute time of the latest source in the sample n, t_n^k is the absolute time of the observation $\boldsymbol{v}_n^k$, and δ_{max} is the maximum time delta allowed between two observations in a single sample, that is, the maximum time period between the earliest and the latest elements. δ_{max} is a constant fixed before training.

(d) The source characteristics $\boldsymbol{s}^k \in \mathbb{R}^{N\left(T\left(S^k\right)\right)}$, where $N\left(T\left(S^k\right)\right)$ is the number of characteristic variables for the source type $T\left(S^k\right)$. Those are variables that characterize a source within its source type; therefore, all sources of a common type share the same characteristic variables, but with potentially different values. The source characteristics depend on the source and are therefore constant across all elements of a source.

As parts of the sources may be missing (e.g., sea surface temperature over land), an availability mask is added to the original channels in the values. Its value is one at points where the values are available and zero where they are missing.

2.2. Self-supervised multi-source training

We first briefly describe the separate values-coordinates embeddings, which are needed to detail the training task. The details of the embeddings are given in Section 2.3.1.

Given a sample $\boldsymbol{X}_n = \{\boldsymbol{X^1}_n, \boldsymbol{X^2}_n, \ldots, \boldsymbol{X}_n^K\}$, each source is embedded into three sequences of vectors as follows: the embedded values, coordinates, and a conditioning vector.

$$\tilde{\mathrm{v}}_n^k = \mathrm{ValuesEmbedding}\left(\boldsymbol{v}_n^k, \mathrm{lm}_n^k\right) \in \mathbb{R}^{L^k \times d_V}$$
$$\tilde{\boldsymbol{c}}_n^k = \mathrm{CoordsEmbedding}\left(\boldsymbol{c}_n^k, \delta_n^k\right) \in \mathbb{R}^{L^k \times d_C}$$
$$\tilde{\boldsymbol{s}}_n^k = \mathrm{ConditionEmbedding}\left(\boldsymbol{s}_n^k, \mathrm{lm}_n^k\right) \in \mathbb{R}^{L^k \times d_V}$$

where L^k is the length of the sequences, d_V and d_C are the embedding dimensions of the values and coordinates, respectively, and $\mathrm{lm}_n^k \in \mathbb{R}^{L^k \times d_V}$ is a binary land–sea mask. The conditioning vector $\tilde{\boldsymbol{s}}_n^k$ is used to pass information from the source characteristics and the land–sea mask to every layer of the model, through adaptive conditional normalization. As the land–sea mask is crucial for the applications explored later, we found that including it in the conditioning at every layer significantly improved the performance.

At each training step, a source $\boldsymbol{X}_n^m$ is chosen randomly with uniform probability to be masked. The embedded values of that source are then replaced with a learned [MASK] token:

$$\tilde{\boldsymbol{v}}_n^m = \mathrm{repeat}([\mathrm{MASK}], L^m \times d_V) \in \mathbb{R}^{L^m \times d_V}$$

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

The embedded coordinates and conditioning are unaltered by the masking and serve as the information from which the model reconstructs the masked values. Since the values, coordinates, and characteristics are embedded independently, there is no leakage of information between the embedded values and the data received by the model after masking.

The set of all embedded values (including the masked ones), embedded coordinates, and conditioning are passed to the model. For each source, the model produces an output of the same shape as the original values. We use the mean square error as a loss function between the original and reconstructed values of the masked source. Points at which the true values are missing are excluded from the loss computation. Figure 1 shows a diagram of the overall mask-and-reconstruct pipeline. In the context of batched training, the masked source is drawn independently between the samples of the batch, and the loss is averaged over the samples. Since which source is masked varies between the samples, only outputting reconstructions for masked sources would lead to different batch sizes across the samples. To mitigate this, the model produces outputs for both masked and unmasked sources. However, the reconstructions for unmasked sources are ignored in the loss.

2.3. Multi-source architecture

In this section, the subscript $_n$ indicating the sample will be omitted for simplicity. The superscript ^{k} designates the k-th source.

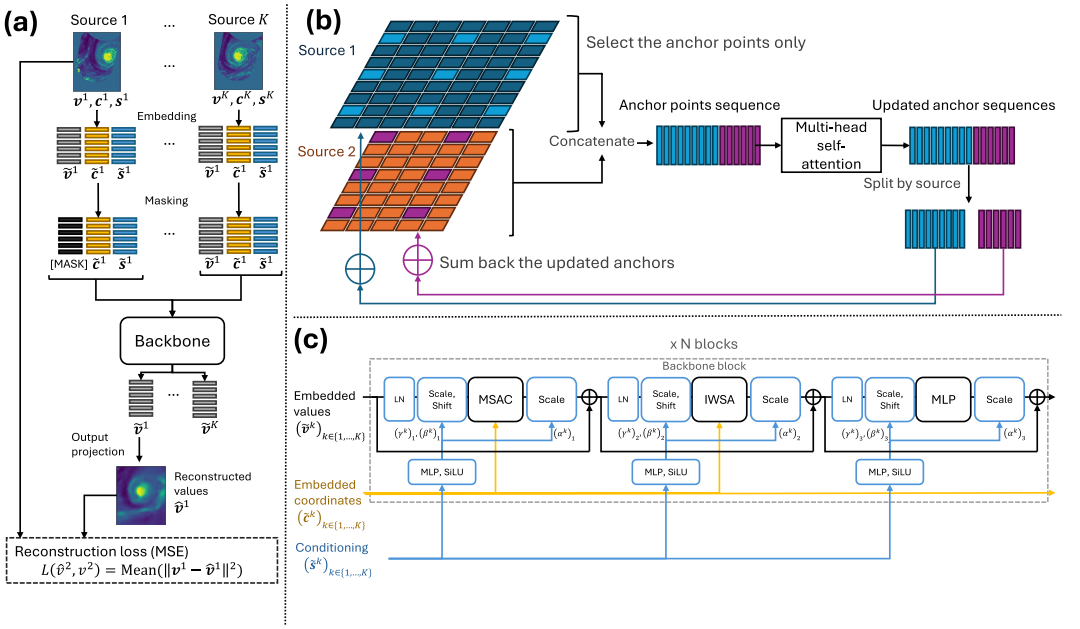

Figure 1. *Overall diagram of the architecture. (a) Overall view of the masking-and-reconstruction pipeline. In this case, Source 1 is chosen to be masked. (b) Multi-source anchored cross-attention mechanism (MSCA) is shown for an example with two 2D sources. In this case, each embedded vector is a patch from the original image, as usual in ViT-based architectures. The embedded vectors that are not anchor points remain unchanged through this layer. (c) Diagram of the backbone. IWSA, individual windowed self-attention. The MSCA block lets information travel across the sources, while the IWSA block lets information travel within each source. The embedded coordinates serve as positional encoding for the attention layers.*

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

2.3.1. Embeddings

The embedding layers project the sources' components into two latent spaces that are common to all sources, one for the values and one for the coordinates. The model includes three embedding modules for each source type (values, coordinates, and characteristics), which process all sources of the type in question. The layers in the embedding modules depend on the dimensionality of the source type.

For 2D sources, the values $\boldsymbol{v}^k \in \mathbb{R}^{C^k \times H^k \times W^k}$ are split into square patches of side length p, which are then embedded separately, following the ViT architecture (Dosovitskiy et al., 2021). This is done with a convolutional layer of kernel size and stride p and d_V filters. The same is done for the geographical coordinates $\boldsymbol{c}^k \in \mathbb{R}^{4 \times H^k \times W^k}$ with d_C filters. This results in two sequences of L^k vectors $\tilde{\boldsymbol{v}}^k$ and $\boldsymbol{c'}^k$ with L^k being the number of patches in the input.

For 0D sources, the values $\boldsymbol{v}^k \in \mathbb{R}^{C^k}$ are embedded with a dense layer with output dimension d_V. The same is done for the geographical coordinates of the shape $\boldsymbol{c}^k \in \mathbb{R}^4$ with an output dimension of d_C. This results in two vectors: one for the values and one for the geographical coordinates.

For all source types, the coordinates embedding module then applies a treatment to integrate the time information: the time delta δ^k is embedded with a dense layer into a vector of dimension d_C, which is then summed to $\boldsymbol{c'}^k$ to obtain a sequence $\boldsymbol{c''}^k$. For 2D sources, the embedded time vector is repeated along the sequence dimension before being summed to $\boldsymbol{c'}^k$. $\boldsymbol{c'}^k$ is finally fed into a small Multi-Layer Perceptron (MLP) with two dense layers, Gaussian Error Linear Unit (GELU) activation, and layer normalization to obtain the final coordinates embedding $\tilde{\boldsymbol{c}}^k$.

Finally, for all sources, another layer embeds the conditioning information: for 2D sources, the 2D land–sea mask is embedded with a convolutional layer as is done for the values and coordinates. The sources' characteristics are then embedded with a linear layer and summed to the embedded land–sea mask. For 0D sources, the land–sea mask is just a scalar, which is concatenated to the characteristic variables, before being passed through a linear layer. The conditioning tensor is used in the adaptive conditional normalization layer described in the next section.

2.3.2. Backbone

The MoTiF backbone is based on the transformer architecture (Vaswani et al., 2023), modified to work with multiple sources. Instead of processing a single sequence of vectors, for each source S_k two sequences are processed throughout the backbone: the values $\tilde{\boldsymbol{v}}^k$ and coordinates $\tilde{\boldsymbol{c}}^k$. Throughout the backbone, only the values sequences are modified and constitute the backbone's output. The coordinate sequences remain unchanged, but still influence the computation of the updated value sequences via conditioning and relative positional attention (as detailed thereafter).

The backbone is a chain of blocks. Each block includes three sub-blocks in succession: (a) a multi-source anchored cross-attention block (MSCA), (b) an individual windowed spatial attention block (IWSA), and (c) an MLP with two dense layers, GELU activation. Each layer is wrapped in an adaptive conditional normalization layer, which lets the information in the coordinates influence the computation on the values in the wrapped layer. The details of the MSCA, IWSA, and conditional normalization layers are given below. Figure 1 describes the successive blocks in the backbone.

Factorized multi-source attention. As computing the attention over the full sequences of all sources at once would be intractable with a large number of sources, we choose to decompose the attention into two steps: (a) cross-sources attention and (b) spatial attention individual to each source. This system is usually referred to as factorized attention, and has notably been used for global weather forecasting with a three-dimensional (3D) representation of the atmosphere (Couairon et al., 2024). That 3D factorization first computes the attention across the vertical column at each geographical point individually, and then the attention across each 2D field separately. This mechanism applies to global weather forecasting since two fields at different altitudes/pressure levels in the atmosphere are well aligned geographically (same latitudes and longitudes). In our case, the sources are, in general, not geographically aligned and are even of different sizes, so the notion of "cross-sources column" is ill-defined. For this reason, we use an anchor-point attention mechanism in the MSCA blocks, which allows points of any areas of different sources to attend to one another. The IWSA blocks then compute the spatial self-attention over each source individually.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

MSCA block. Given sequences of embedded values $\{\tilde{\boldsymbol{v}}^k\}_{k\in\{1,\dots,K\}}$ and coordinates $\{\tilde{\boldsymbol{c}}^k\}_{k\in\{1,\dots,K\}}$, a subset of the vectors (referred to as *anchor points*) is gathered from each sequence. For 0D and 1D sources, the sequence contains a single vector, which is chosen as the anchor. For 2D sources, every Λ vector along each axis (height and width) is selected as an anchor, where Λ is a hyperparameter set to 4 in our experiments. This forms the anchor sequences $\{\overline{\boldsymbol{v}}^k\}_{k\in\{1,\dots,K\}}$ and $\{\overline{\boldsymbol{c}}^k\}_{k\in\{1,\dots,K\}}$. The anchor value sequences are concatenated into a single sequence $\boldsymbol{V}$, and the same is done for the coordinates to obtain a sequence $\boldsymbol{C}$. An attention map $\boldsymbol{A}$ is then computed over $\boldsymbol{V}$ in the traditional sense, with $\boldsymbol{C}$ playing the role of relative positional embedding (as in Shaw et al., 2018): $\boldsymbol{V}$ is projected through a linear layer to a triplet of queries $\boldsymbol{Q}_V \in \mathbb{R}^{L_\Lambda \times d_I}$, keys $\boldsymbol{K}_V \in \mathbb{R}^{L_\Lambda \times d_I}$, and values $\boldsymbol{V}_V \in \mathbb{R}^{L_\Lambda \times d_I}$, where d_I is the inner dimension of the attention layer and L_Λ is the total number of anchor points. The same is done for the coordinates to obtain a triplet $\boldsymbol{Q}_C, \boldsymbol{K}_C, \boldsymbol{V}_C$. The keys and queries are normalized with an RMSNorm layer following Esser et al. (2024). The attention map is then computed as:

$$\mathbf{A} = \frac{\mathrm{Softmax}\left(\mathbf{Q}_V(\mathbf{K}_V)^T + \mathbf{Q}_C(\mathbf{K}_C)^T\right)}{\sqrt{d_I}}$$

In practice, multi-head self-attention (Vaswani et al., 2023) is used. The anchor values sequence is then multiplied with the attention map as in the traditional transformer and then linearly projected back to d_V: $\boldsymbol{V}_{out} = \mathrm{Linear}(\boldsymbol{AV})$. Then, $\boldsymbol{V}_{out}$ is split back into the anchors vectors, and the updated anchor values are summed to their values before computing the attention. The vectors that were not selected as anchor points are unchanged throughout the process. Figure 1 shows an example of this layer for two 2D sources.

IWSA block. The IWSA block computes the attention between vectors of the same source, for each source separately. For 2D sources, this is done using spatial attention windows as in the Swin Transformer (Liu et al., 2021). For odd-indexed blocks, the windows are shifted by half the window size. The window size is set to 8. For 0D sources, this block does nothing (identity), as 0D sources are sequences of a single vector.

Adaptive conditional normalization layer. This layer is the same modulation mechanism used in the Diffusion Transformer (DiT) model (Peebles and Xie, 2023), and is used to let each layer receive the information embedded in the conditioning $\tilde{\boldsymbol{s}}$, including the source characteristics and land–sea mask. Let F be a block $\{\tilde{\boldsymbol{w}}^k\}_{k\in\{1,\dots,K\}} = F\left(\{\tilde{\boldsymbol{v}}^k\}_{k\in\{1,\dots,K\}}, \{\tilde{\boldsymbol{c}}^k\}_{k\in\{1,\dots,K\}}\right)$ that takes as input sequences of embedded values and coordinates and outputs updated embedded values. The conditioning is projected to a triplet $\gamma^k, \beta^k, \alpha^k$ by a dense layer followed by a Sigmoid Linear Unit (SiLU). The embedded values are then shifted and scaled as $\boldsymbol{\omega}^k = \gamma^k \tilde{\boldsymbol{v}}^k + \beta^k$. The block is then applied to the shifted-scaled values as $\left\{\tilde{\boldsymbol{w}}'^k\right\}_{k\in\{1,\dots,K\}} = F\left(\{\boldsymbol{\omega}^k\}_{k\in\{1,\dots,K\}}, \{\tilde{\boldsymbol{c}}^k\}_{k\in\{1,\dots,K\}}\right)$. Finally, the output of the block is multiplied by the gate and summed to a skip connection: $\tilde{\boldsymbol{\omega}}^k = \alpha^k \tilde{\boldsymbol{w}}'^k + \tilde{\boldsymbol{v}}^k$.

2.3.3. Output projection layers

The output projection layers receive the value sequence $\{\tilde{\boldsymbol{v}}^k\}_{k\in\{1,\dots,K\}}$ output by the backbone and project them to the original source spaces. Similar to the embedding layers, sources of the same type share the same output layer. For 2D sources, the output layer is made up of a layer normalization and a strided deconvolution followed by a two-hidden-layer ResNet. To reduce checkerboard artifacts due to the patching, the strided deconvolutions are initialized with the ICNR scheme (Aitken et al., 2017). For 0D sources, the output layers are made up of a layer normalization followed by a dense layer.

3. Experiments on TCs and ETCs

3.1. Data

Dataset. We use the TC-PRIMED dataset (Razin et al., 2023). This dataset includes multiple sources of data specific to TCs and ETCs. The first type of source used from TC-PRIMED is PMW satellite images

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

around the 37 GHz band. PMW observations have proven useful in the forecasting of TCs (Wimmers et al., 2019; Ma et al., 2023), as a proxy for liquid precipitations (Razin et al., 2023). The PMW images are taken from 11 sensors on 11 different satellites, with different characteristics (exact frequency, scan type, ground sampling distance, and orbit). Since the satellites were in orbit over different periods of time, and only a fraction of the satellites orbited over any given storm, only a fraction of the sources is eventually available for each pair (storm and time), with the maximum being 6.

The PMW images in TC-PRIMED cover at least 50% of the 750-km disc centered on the storm's center. However, the images are not, in general, centered on the cyclone. All sources of PMW images are grouped in the source type "passive microwave" (in the sense defined in Section 2). We also use ERA5 surface fields cropped to a 20° × 20° box around the cyclone. The ERA5 fields are centered on the cyclone's center of circulation according to best-track estimations. The included ERA5 fields are 10-m u and v components of the wind, mean sea-level pressure, and sea surface temperature. Finally, we use best-track data included with TC-PRIMED, which gives the best retrospective estimates of the maximum sustained wind (MSW) speeds within the cyclones.

We refer the reader to the original article about TC-PRIMED (Razin et al., 2023) and its documentation for the details of the preprocessing applied to the sources to build the datasets, as well as the exact characteristics of each sensor/satellite. The PMW images are products at level 1C, which means they could realistically be used in real-time applications.

Preprocessing. While the architecture can be applied to the PMW images in their original satellite geometry (as the spatial coordinates of every point are fed to the model), we found that resampling the images to a regular lat-lon grid improved the training. We use a spatial resolution of 0.15°, which is the finest resolution across the PMW sensors used in our experiment. Since all PMW images have a native spatial resolution coarser than 0.15°, no information is lost by the resampling operation: all images use the same regular grid, but coarser sources are blurrier. All sources are finally normalized by subtracting their means and dividing by their standard deviations.

Train-validation-test split. The list of samples is divided into three splits: training, validation, and test, with proportions of 0.8, 0.15, and 0.15, respectively. Two samples with the same storm ID (which give information about the same cyclone) are necessarily put in the same split. The test split is not used or looked at until making the final evaluation of the model.

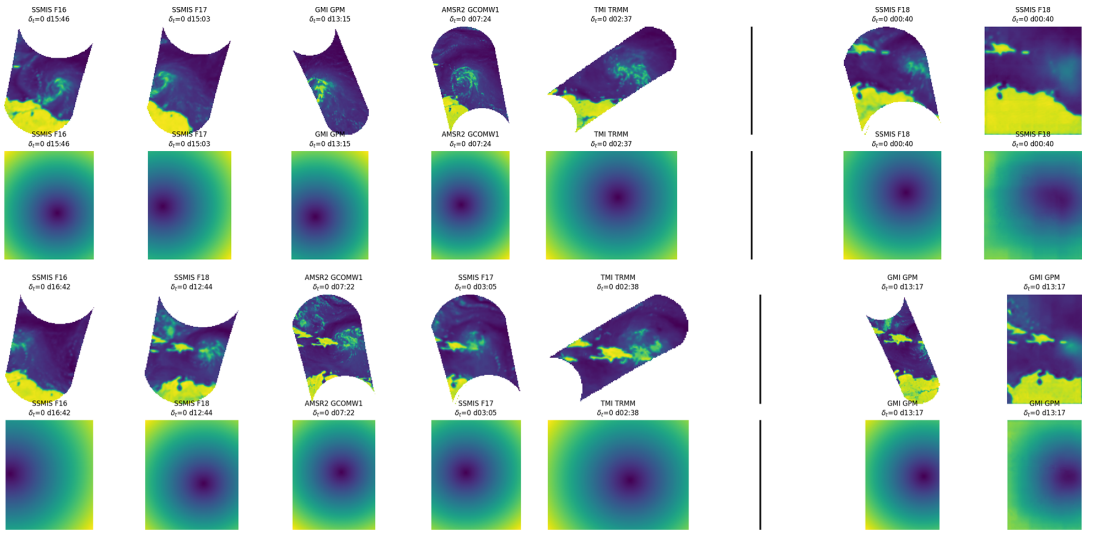

Figure 2. *Two examples (the first two rows and the last two rows, respectively) of reconstructions from the same storm, but with different available sources and targets. The model is able to correctly identify the area in the other sources that is closest in space and time to its target.*

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

3.2. Pretraining

We pretrain the MoTiF model in the self-supervised setting described in Section 2.2, using the PMW and ERA5 data. For each training sample, either ERA5 or one of the satellite sources is masked, and the model is tasked to reconstruct it using the other sources. For the ERA5 and PMW data, we add a "distance-to-center" channel: this channel is an image of the same dimensions as the other fields in the source, and gives at each pixel the distance from the center of the storm (given by the best-track estimates). This channel is not given to the model as input, but the model is asked to predict it alongside the other channels; this channel will be used to quantitatively evaluate how well the model locates the cyclone.

To limit the size of the batches, a sample can only include at most one observation from each satellite source and two ERA5 states. The time window is set to $\delta_{max} = 24h$. When several observations of a source are available within the time window (for example, ERA5 always has four time points in 24 h), one is selected with uniform probability.

We then fine-tune the model on three different tasks: forecasting ERA5 fields, forecasting the storms' MSW speed, and interpolating or extrapolating PMW images. The following sections describe the results of those experiments.

The pretraining uses a cosine annealing learning rate schedule with a maximum of 3e−5 and a minimum of 1e−8 with five epochs of warmup. The fine-tuning also uses a cosine annealing schedule but from 3e−6 to 1e−9, and over 50 epochs. In practice, all fine-tuning experiments converged to their optima in at most five epochs, making it much shorter than the pretraining by using an early stopping policy.

3.3. Results

3.3.1. ERA5 forecasting

Reconstructing the wind fields from remote-sensing observations is among the most desirable areas of research for applying AI to TCs (Duong et al., 2023). The architecture described here allows to use ERA5 data as both input and/or output, coupled with a flexible amount of remote-sensing images, given at any point in time. In this experiment, the model is tasked with forecasting the surface fields of ERA5 $t_0 + 24h$, given sources that are in the time window $[t_0 - 24h, t_0]$. Like in the pretraining, the model receives at most one observation per PMW sensor and one ERA5 state. Contrary to the pretraining, when several data points of a source exist in the time window, the most recent is selected. This notably means that the model always receives the ERA5 state at exactly t_0. An example of a sample is shown in Figure 2. Since the ERA5 fields are centered on the storm's center in the TC-PRIMED dataset, their spatial coordinates naturally give information on the location of the storm. To limit that information and force the model to detect the location of the storm on its own, the ERA5 fields are randomly cropped during training: a rectangular subset of the fields is selected, with a width and height randomly sampled between 60 and 100% of the original size. As a result, during training, the ERA5 fields are generally not centered on the storm.

We compare the performance of the architecture in different settings: the pretrained model without fine-tuning, the fine-tuned model with both ERA5 and PMW inputs, the fine-tuned model with only ERA5 as input, and finally, the fine-tuned model with only PMW images as input. We use the root mean squared error and mean absolute error (MAE) as evaluation metrics. Except for the distance-to-center channel, the pixels outside of the 1000-km radius around the center of the cyclone are excluded from the error computation, in order to evaluate the forecast in the relevant area specifically. The radius of 1,000 km is chosen to roughly fit the maximum diameter of tropical storms (2,220 km for Typhoon Tip). We also evaluate the model on the minimum sea-level pressure, which is a strong indicator of a storm's intensity. The results are indicated in Table 1 and Figures 3 and 4. The fine-tuning operation, as expected, improves the performance on all variables. Interestingly, using only PMW images results in a lower error for the distance to the storm's center than using only ERA5 as input, suggesting PMW images are more effective for forecasting the storm's position at $+24h$. A potential reason for this is the error in the locations of storms in ERA5 (Bié and de Camargo, 2023), which can be corrected by the model with the PMW images, which are direct observations and do not contain a location error. On the other hand, ERA5 is a more useful input for forecasting the future ERA5 fields, although coupling both types of input yields the best results.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

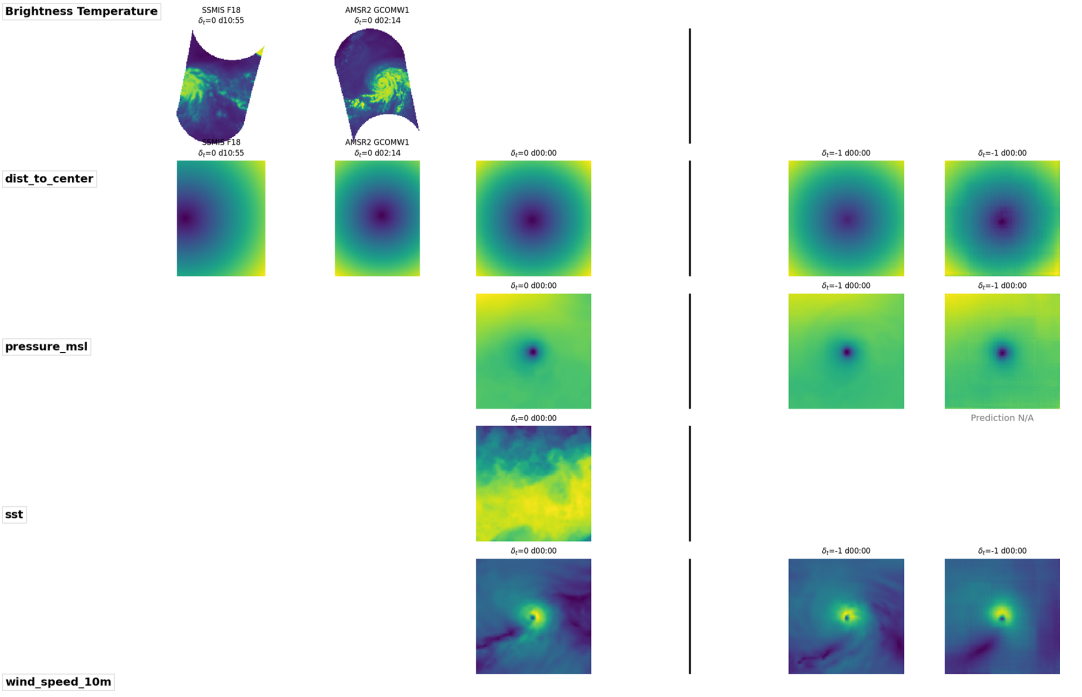

Figure 3. *Example of ERA5 surface fields reconstruction from a previous ERA5 state and PMW images. Left of the vertical bar: available sources, given as input to the model. Right of the vertical bar: targets (left) and predictions (right). Each row corresponds to a variable. The distance-to-center variable is displayed for every source, but is actually not given as input to the model, being only required as output. The sea-surface temperature (SST) field is only used as input. The annotations $\delta_t = Ddhh : mm$ indicate the time delta between the reference time t0 and the time of the observation, in the format days–hours–minutes. The forecast time is at $t_0 + 1$ day; thus, the predictions are at $\delta_t = -1day$.*

Table 1. *RMSE values for different channels and sets of input sources. Lower is better. SLP, sea-level pressure*

Setting	Mean SLP (hPa)	V-wind 10 m (kt)	U-wind 10 m (kt)	Min SLP (hPa)	Wind speed 10 m (kt)
No fine-tuning	3.18	3.09	3.02	7.07	2.80
Fine-tuned (PMW only)	3.56	3.31	3.39	8.78	3.03
Fine-tuned (ERA5 only)	2.53	2.97	2.89	6.66	2.62
Fine-tuned (ERA5 + PMW)	2.28	2.66	2.64	5.62	2.34

3.3.2. Intensity forecasting

In this second experiment, the model is tasked with forecasting the MSW speed of the storm at $t_0 + 24h$. The time window is again set to $[t_0 - 24h, t_0]$. Since this data source was not included during the pretraining, the embedding and output projection layers for the MSW source are fully learned during the fine-tuning phase. The sources available in the samples to make the forecast are the same as for the ERA5 forecasting experiment: at most, one observation per PMW sensor, and one ERA5 state. The results

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

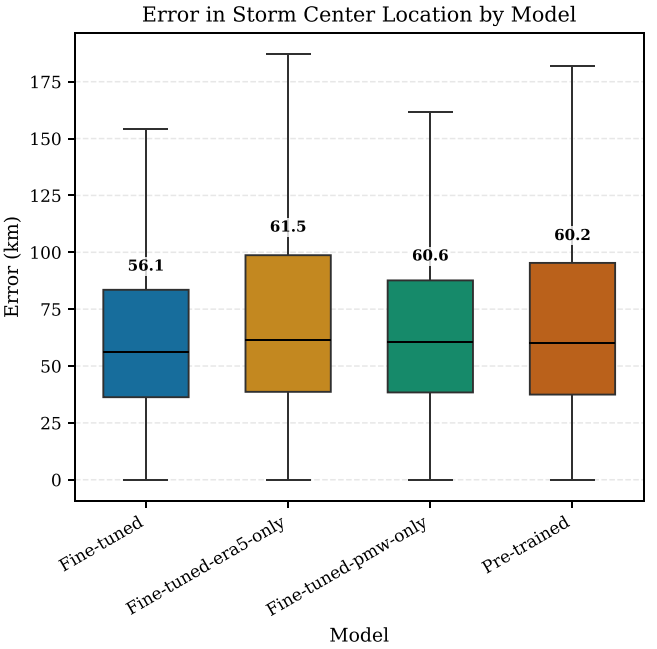

Figure 4. *Comparison of error in forecast storm location (km). Text annotations indicate the median values.*

are given in Table 2 and Figure 5. As for the ERA5 forecasting, combining both types of input sources yields the lowest average error. While the architecture was not designed with the specific objective of doing intensity forecasting, the overall MAE for the model using both ERA5 and PMW images as input is comparable to those of forecasting agencies (12.9 to 22.0 kt) and of other deep learning techniques such as DeepTC (Kim et al., 2024).

3.3.3. PMW reconstruction

This experiment uses the same self-supervised task as the pretraining, while only keeping the PMW sources: one PMW image is randomly masked, and the model is tasked with reconstructing it from the other available PMW images. Figures 6–10 show examples of reconstructions, and more are included in the Supplementary Material. Overall, the model is able to well locate the core of the storm in the reconstructed images (as shown by the distance-to-center channel), and reproduces the structure to some extent. It is noteworthy that the distance-to-center channel is not given as input to the model, but is required to be predicted alongside the brightness temperature. It is still included in the figures to help the reader identify the center of the storm in the PMW images. The model suffers from an obvious lack of low frequencies, even in the case where the image to reconstruct is very close in time to an available source. This limitation is expected as the model is trained with the mean squared error as the objective, whose optimal point is the mean of the target distribution. Therefore, the model tries to predict the mean of the distribution, which, by averaging all possible realizations, smoothes out all of the local, high-frequency components.

4. Discussion

4.1. Potential impacts

Using multiple sources of data for data-driven weather forecasting is a field that is rapidly gaining attention, notably due to its potential for extreme weather. This work aims to be a step towards a general ML method for multi-source weather interpolation and forecasting by tackling the challenge of fusing

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

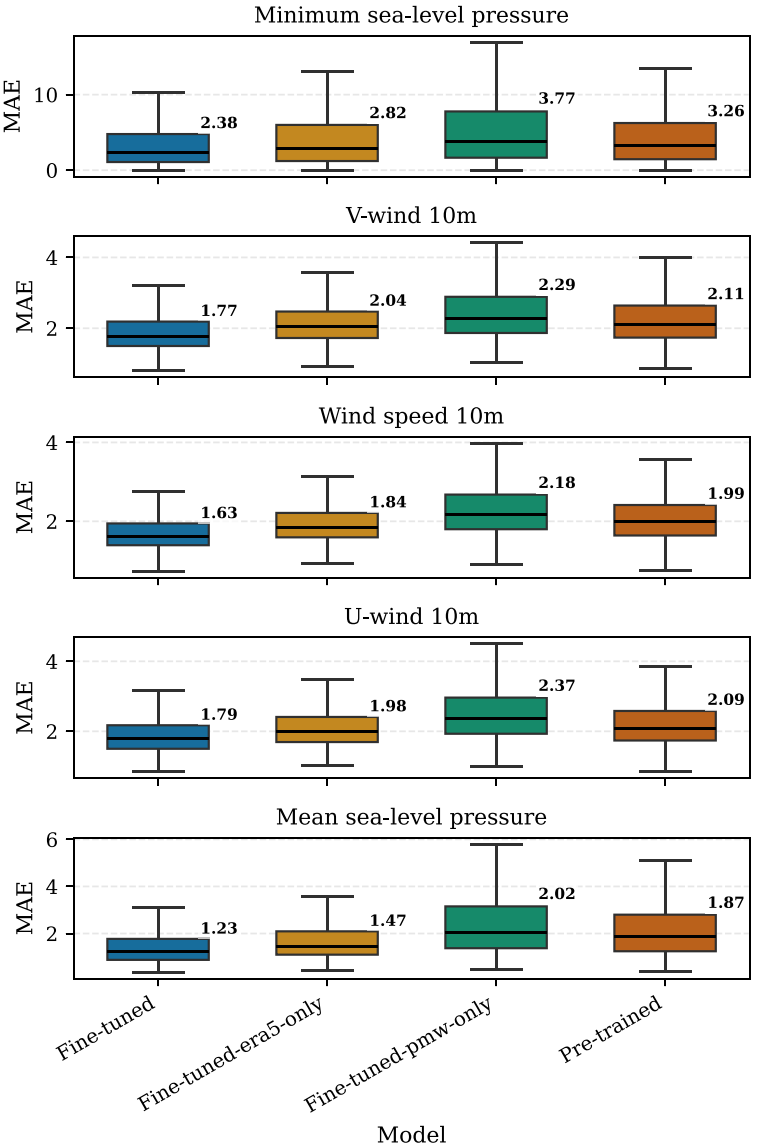

Figure 5. *Comparison of mean absolute error (MAE) for different sets of input sources, as well as the non-fine-tuned model. Lower is better. Text annotations indicate the median values.*

Table 2. *Overall RMSE and MAE for intensity forecasting for different input sources (kt). Lower is better*

Setting	RMSE	MAE
PMW only	23.07	17.37
ERA5 only	19.54	14.39
ERA5 + PMW	17.28	12.51

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

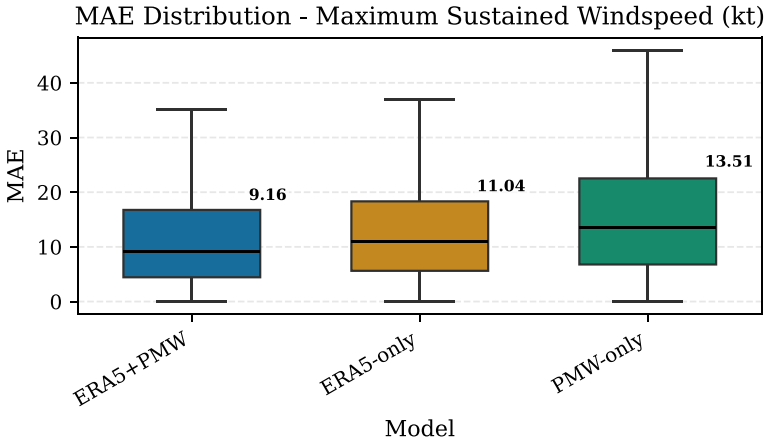

Figure 6. *Distribution of the mean absolute error at the intensity forecasting task, for different subsets of input sources. Lower is better.*

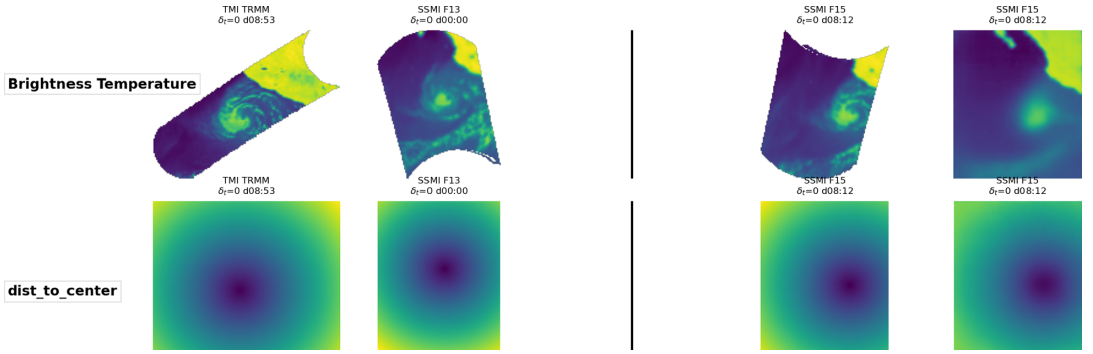

Figure 7. *Example of reconstruction of a passive microwave image. Left of the vertical bar, top row (brightness temperature): available sources, fed to the model as input. Right, top row: target and prediction. Bottom row: "dist_to_center" channel, which is the distance between each pixel and the storm's center according to best-track data. This channel is not given as input, but is required to be predicted by the model to judge its ability to locate the cyclone. The annotations* $\delta_t = Ddhh : mm$ *indicate the time delta between the reference time* t_0 *and the time of the observation, in the format days–hours–minutes. In this example, the closest source (41 min apart, left) in time does not capture the tail of the cyclone, while another image, further away chronologically (8 h 53 min apart), does show it. The model is able to merge the sources, in the sense that it fetches information from the closer image for the center and from the other one for the cyclone's tail.*

sources that are misaligned in space and time. Regarding the application of TCs and ETCs intensity forecasting, this work goes beyond its predecessors: while TC multi-sources intensity forecasting with ML models has been studied many times before (Chen et al., 2018; Giffard-Roisin et al., 2020; Ma et al., 2023), previous works used images aligned in space (centered on the storm's center) and at regular time intervals (generally, every 6 h). However, the potential applications go further: examples of problems which can theoretically be addressed with this architecture are: (a) nowcasting precipitation fields using radar data as target and assimilated states, previous radar data, station measurements, and microwave images as input; (b) reconstructing the wind fields around the eye of TCs using Synthetic-aperture Radar (SAR) data as target.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

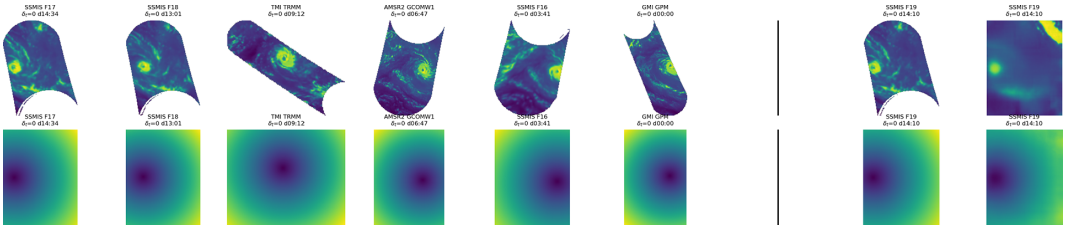

Figure 8. *Example of reconstruction using images from six different satellites as input to reconstruct a seventh. The multi-source cross-attention lets the model process many different sources at a reasonable cost. In particular, examples such as this one can be processed in the same batch as examples with only one or two sources, both during training and inference. While cases with five or more available satellites are a minority of the training data, the model does not collapse, and shows an ability to select information from the sources that are closest in time and space. Every image has a specific aspect ratio and size.*

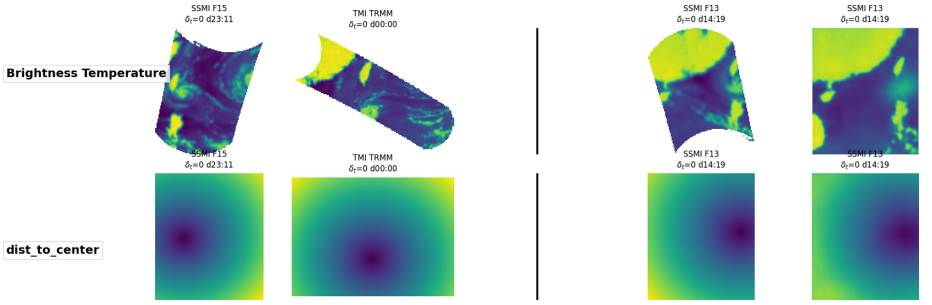

Figure 9. *Example of temporal interpolation of a passive microwave image where the image to reconstruct is in between the two other available sources chronologically. While the only the low frequencies are reconstructed, the distance-to-center channel shows that the location of the storm's core was well interpolated.*

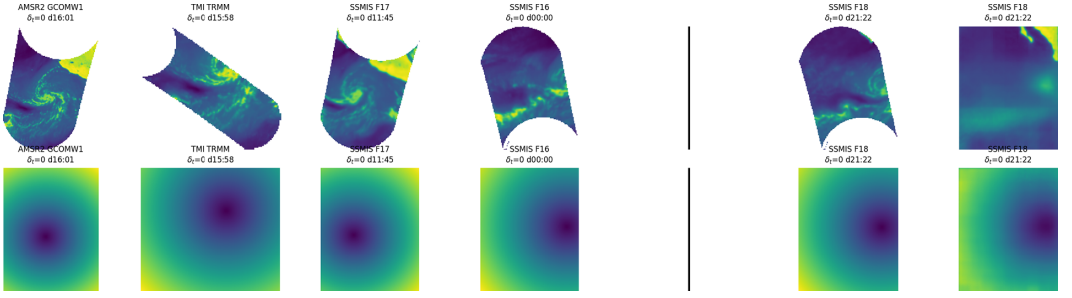

Figure 10. *Example of reconstruction where the image to reconstruct is anterior to the other available images.*

4.2. Limitations

A clear limitation of this work is that the model is trained with a deterministic objective, using the mean squared error as a loss function. In this setting, the model learns to avoid predicting high-frequency features when reconstructing a source, as those are not fully determined by the other observations. This greatly limits the model's applicability to tasks requiring the prediction of small-scale features, such as the example problems mentioned in Section 4.1.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

4.3. Future work

4.3.1. Using a generative model

Since using a deterministic training objective is the main limitation of the model, the main direction to improve it is to use a generative objective instead. Generative models, such as those based on diffusion, learn to sample from the target distribution. Thus, a generative model trained well enough would produce samples with high-frequency features if they appear in the target distribution. Diffusion models have notably used this advantage to better forecast extreme events in global weather forecasting (Zhong et al., 2023.

Since training a diffusion model is costlier than a deterministic model, a possibility is to pretrain a deterministic model on the self-supervised reconstruction task, and then fine-tune on a specific task with the diffusion objective.

4.4. Testing the model on more types of extreme weather

As mentioned in Section 4.1, there are more potential applications over which the model can be tested; however, most of those (such as precipitation nowcasting and wind field reconstruction) first require the model to produce high-frequency features, and could therefore serve as benchmarks for a potential generative version.

5. Conclusion

This work presents an architecture that allows learning from sources of different natures, dimensionalities, and characteristics, as input and/or output. Contrary to previous works that combine multiple sources for weather forecasting, this architecture can use sources that are misaligned in space and at irregular time intervals, and can work with a flexible set and number of sources. We propose a self-supervised reconstruction task to train the model, which forces the model to learn the relationship between the spatiotemporal coordinates of all sources. We apply that self-supervised pretraining on a mix of assimilated states (ERA5 reanalysis surface fields) and remote-sensing images (PMW satellite imagery). We then explore different possibilities of the architecture by fine-tuning the model on different downstream tasks, in which the predicted sources differ in nature and dimensionality. While this work leaves many questions open regarding the applicability of the architecture for directly useful problems, it represents a step forward compared to the previous literature for combining machine learning and the diversity of data types available in geospatial sciences.

Open peer review. To view the open peer review materials for this article, please visit http://doi.org/10.1017/eds.2025.10014.

Supplementary material. The supplementary material for this article can be found at http://doi.org/10.1017/eds.2025.10014.

Acknowledgments. The authors are grateful to Anastase Charantonis for the fruitful discussions about the training objective. Many thanks to Guillaume Couairon for the advice on the factorized attention, as well as to David Landry for his technical advice.

Author contribution. Conceptualization: C.D.; Data curation: C.D.; Funding acquisition: C.M.; Investigation: C.D.; Methodology: C.D.; Project administration: C.D. and C.M.; Resources: C.M.; Software: C.D.; Supervision: C.M.; Validation: C.D.; Visualization: C.D. and C.M.; Writing – original draft: C.D.; Writing – review and editing: C.M. All authors approved the final submitted draft.

Competing interests. The authors declare none.

Data availability statement. All of the data used in this work is part of the TC-PRIMED dataset (https://doi.org/10.25921/dmy1-0595). The code is fully available on its repository or Zenodo: https://doi.org/10.5281/zenodo.15782083.

Ethical standard. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Provenance statement. This article is part of the Climate Informatics 2025 proceedings and was accepted in Environmental Data Science on the basis of the Climate Informatics peer review process.

Funding statement. This research was supported by the French government through the Choose France Chair in AI program. This work was granted access to the HPC resources of IDRIS under the allocation AD011014682R1 made by GENCI.

References

Aitken A, **Ledig C**, **Theis L**, **Caballero J**, **Wang Z and Shi W** (2017) Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize. http://arxiv.org/abs/1707.02937.

Andrychowicz M, **Espeholt L**, **Li D**, **Merchant S**, **Merose A**, **Zyda F**, **Agrawal S and Kalchbrenner N** (2023) Deep learning for day forecasts from sparse observations. http://arxiv.org/abs/2306.06079.

Ayush K, **Uzkent B**, **Meng C**, **Tanmay K**, **Burke M**, **Lobell D and Ermon S** (2021) Geography-aware self-supervised learning. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada: IEEE, pp. 10161–10170. https://doi.org/10.1109/ICCV48922.2021.01002.

Bi K, **Xie L**, **Zhang H**, **Chen X**, **Gu X and Tian Q** (2023) Accurate medium-range global weather forecasting with 3D neural networks. *Nature 619*(7970), 533–538. https://doi.org/10.1038/s41586-023-06185-3.

Bié AJ and de Camargo R (2023) Tropical cyclones position and intensity in the Southwest Indian Ocean as represented by CFS and ERA5 atmospheric reanalysis datasets. *International Journal of Climatology 43*(10), 4532–4551. https://doi.org/10.1002/joc.8101.

Chen B, **Chen B-F and Lin H-T** (2018) Rotation-blended CNNs on a new open dataset for tropical cyclone image-to-intensity regression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18. New York: Association for Computing Machinery, pp. 90–99. https://doi.org/10.1145/3219819.3219926.

Chen T-C, **Collet F and Di Luca A** (2024) Evaluation of ERA5 precipitation and 10-m wind speed associated with extratropical cyclones using station data over North America. *International Journal of Climatology 44*(3), 729–747. https://doi.org/10.1002/joc.8339.

Cong Y, **Khanna S**, **Meng C**, **Liu P**, **Rozi E**, **He Y**, **Burke M**, **Lobell DB and Ermon S** (2023) SatMAE: Pre-training transformers for temporal and multi-spectral satellite imagery. http://arxiv.org/abs/2207.08051.

Couairon G, **Singh R**, **Charantonis A**, **Lessig C and Monteleoni C** (2024) ArchesWeather & ArchesWeatherGen: A deterministic and generative model for efficient ML weather forecasting. http://arxiv.org/abs/2412.12971.

Dosovitskiy A, **Beyer L**, **Kolesnikov A**, **Weissenborn D**, **Zhai X**, **Unterthiner T**, **Dehghani M**, **Minderer M**, **Heigold G**, **Gelly S**, **Uszkoreit J and Houlsby N** (2021) An image is worth 16x16 words: Transformers for image recognition at scale. http://arxiv.org/abs/2010.11929.

Duong Q-P, **Wimmers A**, **Herndon D**, **Tan Z-M**, **Zhuo J-Y**, **Knaff J**, **Al Abdulsalam I**, **Horinouchi T**, **Miyata R and Avenas A** (2023) Objective satellite methods including AI algorithms reviewed for the tenth international workshop on tropical cyclones (IWTC-10). *Tropical Cyclone Research and Review 12*(4), 259–266. https://doi.org/10.1016/j.tcrr.2023.11.001.

Esser P, **Kulal S**, **Blattmann A**, **Entezari R**, **Müller J**, **Saini H**, **Levi Y**, **Lorenz D**, **Sauer A**, **Boesel F**, **Podell D**, **Dockhorn T**, **English Z**, **Lacey K**, **Goodwin A**, **Marek Y and Rombach R** (2024) Scaling rectified flow transformers for high-resolution image synthesis. http://arxiv.org/abs/2403.03206.

Giffard-Roisin S, **Yang M**, **Charpiat G**, **Bonfanti CK**, **Kégl B and Monteleoni C** (2020) Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data. *Frontiers in Big Data 3*:1. https://doi.org/10.3389/fdata.2020.00001.

Gordon J, **Bruinsma WP**, **Foong AYK**, **Requeima J**, **Dubois Y and Turner RE** (2020) Convolutional conditional neural processes. http://arxiv.org/abs/1910.13556.

Guo X, **Lao J**, **Dang B**, **Zhang Y**, **Yu L**, **Ru L**, **Zhong L**, **Huang Z**, **Wu K**, **Hu D**, **He H**, **Wang J**, **Chen J**, **Yang M**, **Zhang Y and Li Y** (2024) SkySense: A multi-modal remote sensing foundation model towards universal interpretation for earth observation imagery. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA: IEEE, pp. 27662–27673. https://doi.org/10.1109/CVPR52733.2024.02613.

He K, **Chen X**, **Xie S**, **Li Y**, **Dollár P and Girshick R** (2021) Masked autoencoders are scalable vision learners. http://arxiv.org/abs/2111.06377.

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 RJ**, **Hólm E**, **Janisková M**, **Keeley S**, **Laloyaux P**, **Lopez P**, **Lupu C and Radnoti G** (2020) Patricia de Rosnay, Iryna Rozum, Freja Vamborg, Sebastien Villaume, and Jean-Noël Thépaut. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society 146*(730), 1999–2049. https://doi.org/10.1002/qj.3803.

Kim J-H, **Ham Y-G**, **Kim D**, **Li T and Ma C** (2024) Improvement in forecasting short-term tropical cyclone intensity change and their rapid intensification using deep learning. *Artificial Intelligence for the Earth Systems 3*, e230052. https://doi.org/10.1175/AIES-D-23-0052.1.

Klemmer K, **Rolf E**, **Robinson C**, **Mackey L and Rußwurm M** (2024) SatCLIP: global, general-purpose location embeddings with satellite imagery. http://arxiv.org/abs/2311.17179.

Kochkov D, **Yuval J**, **Langmore I**, **Norgaard P**, **Smith J**, **Mooers G**, **Lottes J**, **Rasp S**, **Düben P**, **Klöwer M**, **Hatfield S**, **Battaglia P**, **Sanchez-Gonzalez A**, **Willson M**, **Brenner MP and Hoyer S** (2023) Neural general circulation models. http://arxiv.org/abs/2311.07222.

Lam R, **Sanchez-Gonzalez A**, **Willson M**, **Wirnsberger P**, **Fortunato M**, **Alet F**, **Ravuri S**, **Ewalds T**, **Eaton-Rosen Z**, **Hu W**, **Merose A**, **Hoyer S**, **Holland G**, **Vinyals O**, **Stott J**, **Pritzel A**, **Mohamed S and Battaglia P** (2023) GraphCast: Learning skillful medium-range global weather forecasting. http://arxiv.org/abs/2212.12794.

https://doi.org/10.1017/eds.2025.10014 Published online by Cambridge University Press

Liu Z, **Lin Y**, **Cao Y**, **Hu H**, **Wei Y**, **Zhang Z**, **Lin S and Guo B** (2021) Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada. IEEE, pp. 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986.

Ma D, **Wang L**, **Fang S and Lin J** (2023) Tropical cyclone intensity prediction by inter- and intra-pattern fusion based on multi-source data. *Environmental Research Letters 18*(1), 014020. https://doi.org/10.1088/1748-9326/aca9e2.

McNally A, **Lessig C**, **Lean P**, **Boucher E**, **Alexe M**, **Pinnington E**, **Chantry M**, **Lang S**, **Burrows C**, **Chrust M**, **Pinault F**, **Villeneuve E**, **Bormann N and Healy S** (2024) Data driven weather forecasts trained and initialised directly from observations. http://arxiv.org/abs/2407.15586.

Nguyen T, **Shah R**, **Bansal H**, **Arcomano T**, **Maulik R**, **Kotamarthi V**, **Foster I**, **Madireddy S and Grover A** (2024) Scaling transformer neural networks for skillful and reliable medium-range weather forecasting, October 2024. http://arxiv.org/abs/2312.03876. arXiv:2312.03876 [physics].

Olivetti L and Messori G (2024) Do data-driven models beat numerical models in forecasting weather extremes? A comparison of IFS HRES, Pangu-weather, and GraphCast. *Geoscientific Model Development 17*(21), 7915–7962. https://doi.org/10.5194/gmd-17-7915-2024.

Peebles W and Xie S (2023) Scalable diffusion models with transformers. http://arxiv.org/abs/2212.09748.

Price I, **Sanchez-Gonzalez A**, **Alet F**, **Andersson TR**, **El-Kadi A**, **Masters D**, **Ewalds T**, **Stott J**, **Mohamed S**, **Battaglia P**, **Lam R and Willson M** (2024) GenCast: Diffusion-based ensemble forecasting for medium-range weather. http://arxiv.org/abs/2312.15796.

Razin MN, **Slocum CJ**, **Knaff JA**, **Brown PJ and Bell MM** (2023) Tropical cyclone precipitation, infrared, microwave, and environmental dataset (TC PRIMED). *Bulletin of the American Meteorological Society 104*(11), E1980–E1998. https://doi.org/10.1175/BAMS-D-21-0052.1.

Reed CJ, **Gupta R**, **Li S**, **Brockman S**, **Funk C**, **Clipp B**, **Keutzer K**, **Candido S**, **Uyttendaele M and Darrell T** (2023) Scale-MAE: A scale-aware masked autoencoder for multiscale geospatial representation learning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. Paris, France: IEEE, pp. 4065–4076. https://doi.org/10.1109/ICCV51070.2023.00378.

Shaw P, **Uszkoreit J and Vaswani A** (2018) Self-attention with relative position representations. http://arxiv.org/abs/1803.02155.

Taszarek M, **Pilguj N**, **Allen JT**, **Gensini V**, **Brooks HE and Szuster P** (2021) Comparison of convective parameters derived from ERA5 and MERRA-2 with Rawinsonde data over Europe and North America. https://doi.org/10.1175/JCLI-D-20-0484.1.

Tong Z, **Song Y**, **Wang J and Wang L** (n.d.) VideoMAE: Masked autoencoders are data-efficient learners for self-supervised video pre-training.

Vaswani A, **Shazeer N**, **Parmar N**, **Uszkoreit J**, **Jones L**, **Gomez AN**, **Kaiser L and Polosukhin I** (2023) Attention is all you need. http://arxiv.org/abs/1706.03762.

Vaughan A, **Markou S**, **Tebbutt W**, **Requeima J**, **Bruinsma WP**, **Andersson TR**, **Herzog M**, **Lane ND**, **Chantry M**, **Hosking JS and Turner RE** (2024). Aardvark weather: End-to-end data-driven weather forecasting. http://arxiv.org/abs/2404.00411.

Wimmers A, **Velden C and Cossuth JH** (2019) Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery. https://doi.org/10.1175/MWR-D-18-0391.1.

Zhang, L, **Zhao Y**, **Dong R**, **Zhang J**, **Yuan S**, **Cao S**, **Chen M**, **Zheng J**, **Li W**, **Liu W**, **Zhang W**, **Feng L and Fu H** (2024) A^2-MAE: A spatial-temporal-spectral unified remote sensing pre-training method based on anchor-aware masked autoencoder. http://arxiv.org/abs/2406.08079.

Zhong X, **Chen L**, **Liu J**, **Lin C**, **Qi Y and Li H** (2023) FuXi-extreme: Improving extreme rainfall and wind forecasts with diffusion model. http://arxiv.org/abs/2310.19822.

Zhong X, **Chen L**, **Fan X**, **Qian W**, **Liu J and Li H** (2024) FuXi-2.0: Advancing machine learning weather forecasting model for practical applications. http://arxiv.org/abs/2409.07188

Cite this article: Dauvilliers C and Monteleoni C (2025). MoTiF: a self-supervised model for multi-source forecasting with application to tropical cyclones. *Environmental Data Science*, 4: e36. doi:10.1017/eds.2025.10014