Pangu-Weather: A 3D High-Resolution System for Fast and Accurate Global Weather Forecast

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian✉, *Fellow, IEEE*

Abstract—In this paper, we present Pangu-Weather, a deep learning based system for fast and accurate global weather forecast. For this purpose, we establish a data-driven environment by downloading 43 years of hourly global weather data from the 5th generation of ECMWF reanalysis (ERA5) data and train a few deep neural networks with about 256 million parameters in total. The spatial resolution of forecast is $0.25^\circ \times 0.25^\circ$, comparable to the ECMWF Integrated Forecast Systems (IFS). More importantly, for the first time, an AI-based method outperforms state-of-the-art numerical weather prediction (NWP) methods in terms of accuracy (latitude-weighted RMSE and ACC) of all factors (*e.g.*, geopotential, specific humidity, wind speed, temperature, *etc.*) and in all time ranges (from one hour to one week). There are two key strategies to improve the prediction accuracy: (i) designing a 3D Earth Specific Transformer (3DEST) architecture that formulates the height (pressure level) information into cubic data, and (ii) applying a hierarchical temporal aggregation algorithm to alleviate cumulative forecast errors. In deterministic forecast, Pangu-Weather shows great advantages for short to medium-range forecast (*i.e.*, forecast time ranges from one hour to one week). Pangu-Weather supports a wide range of downstream forecast scenarios, including extreme weather forecast (*e.g.*, tropical cyclone tracking) and large-member ensemble forecast in real-time. Pangu-Weather not only ends the debate on whether AI-based methods can surpass conventional NWP methods, but also reveals novel directions for improving deep learning weather forecast systems.

Index Terms—Numerical Weather Prediction, Deep Learning, Medium-range Weather Forecast.

1 Introduction

Weather forecast is one of the most important scenarios of scientific computing. It offers the ability of predicting future weather changes, especially the occurrence of extreme weather events (*e.g.*, floods, droughts, hurricanes, *etc.*), which has large values to the society (*e.g.*, daily activity, agriculture, energy production, transportation, industry, *etc.*). In the past decade, with the bloom of high-performance computational device, the community has witnessed a rapid development in the research field of numerical weather prediction (NWP) [1]. Conventional NWP methods mostly follow a simulation-based paradigm which formulates the physical rules of atmospheric states into partial differentiable equations (PDEs) and solves them using numerical simulations [2], [3], [4]. Due to the high complexity of solving PDEs, these NWP methods are often very slow, *e.g.*, with a spatial resolution of $0.25^\circ \times 0.25^\circ$, a single simulation procedure for 10-day forecast can take hours of computation using hundreds of nodes in a supercomputer [5]. This largely reduces the timeliness in daily weather forecast and the number of ensemble members that can be used for probabilistic weather forecast. In addition, conventional NWP algorithms largely rely on the parametric numerical models, but these models, albeit being very complex [1], are often considered inadequate [6], [7], *e.g.*, errors will be introduced by parameterization of unresolved processes.

To address the above issues, a promising direction lies in data-driven weather forecast with AI, in particular, deep learning[1]. The methodology is to use a deep neural network to capture the relationship between the input (observed data) and output (target data to be predicted). On specialized computational device (*e.g.*, GPUs), AI-based methods run very fast and easily achieve a tradeoff between model complexity, prediction resolution, and prediction accuracy [9], [10], [11], [12], [13], [14], [15]. As a recent example, FourCastNet [14] increased the spatial resolution to $0.25^\circ \times 0.25^\circ$, comparable to the ECMWF Integrated Forecast Systems (IFS), yet it takes only 7 seconds on four GPUs for making a 100-member, 24-hour forecast, which is orders of magnitudes faster than the conventional NWP methods. However, the forecast accuracy of FourCastNet is still below satisfaction, *e.g.*, the RMSE of 5-day Z500 forecast using a single model and a 100-member ensemble are 484.5 and 462.5, respectively, which are much worse than 333.7 reported by operational IFS of ECMWF [16]. In [8], researchers conjectured that 'a number of fundamental breakthroughs are needed' before AI-based methods can beat NWP.

The breakthrough comes much earlier than they thought. In this paper, we present Pangu-Weather, a powerful AI-based weather forecast system that, **for the first time**, surpasses existing NWP methods (and, of course, AI-based methods) in terms of prediction accuracy of all factors. The test is performed on the 5th generation of ECMWF reanalysis (ERA5) data. We download 43 years (1979–2021) of global weather data, among which we use the 1979–2017 data for training, the 2019 data for validation, and the 2018,

- *All authors are with Huawei Cloud Computing, Shenzhen, Guangdong 518129, China.*
  E-mail: {bikaifeng1,tian.qi1}@huawei.com, 198808xc@gmail.com
- *Qi Tian is the corresponding author.*

1. Throughout this paper, we will use 'conventional NWP' or simply 'NWP' to refer to the numerical simulation methods, and use 'AI-based' or 'deep learning based' to specify data-driven forecast systems. We understand that, verbally, AI-based methods also belong to NWP, but we follow the convention [8] to use these terms.

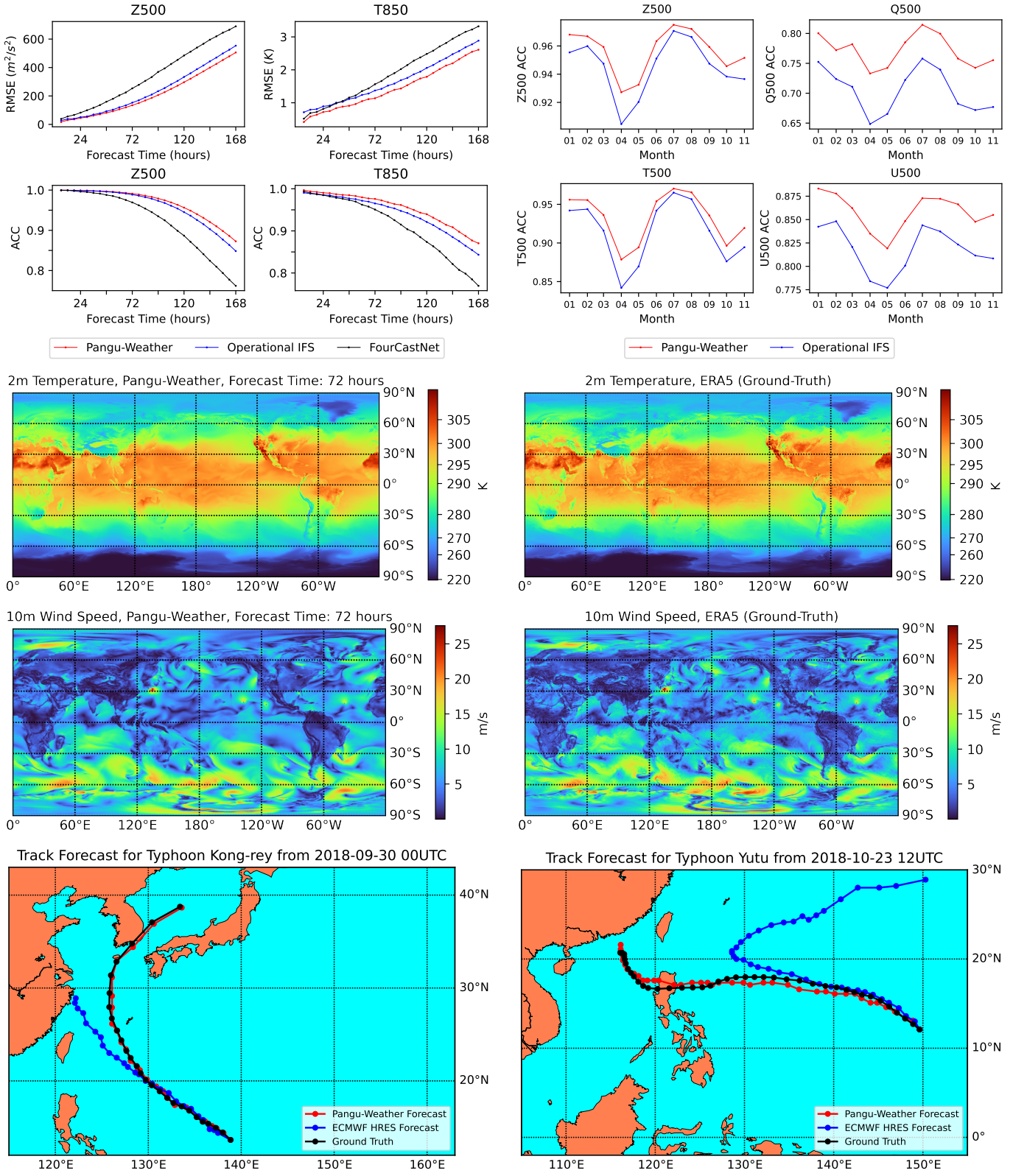

Fig. 1: A showcase of Pangu-Weather's forecast results. **Top**: Pangu-Weather claims significant advantages over operational IFS (NWP) and FourCastNet (AI-based) in terms of forecast accuracy (i) of different factors (500hPa geopotential, Z500, and 850hPa temperature, T850) and (ii) with respect to different months in year. **Middle**: visualization of Pangu-Weather's 3-day forecast of 2m temperature (T2M) and 10m wind speed at 00:00 UTC, September 1st, 2018, with comparison to the ERA5 ground-truth. **Bottom**: Pangu-Weather produces more accurate tracking for two tropical cyclones in 2018, *i.e.*, Typhoon Kong-rey (2018-25) and Yutu (2018-26). Specifically, Pangu-Weather predicts the correct path of Yutu (*i.e.*, it goes to the Philippines) 48 hours earlier than the ECMWF-HRES forecast.

2020, 2021 data for testing. We choose 13 pressure levels, each with 5 important variables (*i.e.*, geopotential, specific humidity, temperature, u-component and v-component of wind speed), and the surface level with 4 variables (*i.e.*, $2\mathrm{m}$ temperature, u-component and v-component of $10\mathrm{m}$ wind speed, and mean sea-level pressure).

Some key results are summarized in Figure 1. **Quantitatively**, Pangu-Weather outperforms all existing weather forecast systems. In particular, with a single-member forecast, Pangu-Weather reports an RMSE of 5-day Z500 forecast of 296.7, significantly better than the operational IFS [16] and the previous best AI-based method (*i.e.*, FourCastNet [14]) which reported 333.7 and 462.5, respectively. In addition, the inference cost of Pangu-Weather is merely $1{,}400\mathrm{ms}$ on a single GPU, more than $10000\times$ faster than operational IFS and on par with FourCastNet [14]. **Qualitatively**, Pangu-Weather not only shows high-resolution ($0.25^\circ \times 0.25^\circ$) visualization maps (*e.g.*, for temperature and wind speed), but also offers high-quality extreme weather forecast (*e.g.*, for tropical cyclone tracking).

The technical contribution of Pangu-Weather is two-fold. **First**, we integrate height information (offered by different pressure levels) into a new dimension, so that the input and output of deep neural networks are in 3D forms. We further design a 3D Earth-specific transformer (3DEST) architecture to process 3D data. Our experiments show that, although 3D data require heavier computational overhead (in particular, the large memory costs obstacle us from using full observation elements and very deep network architectures), 3D models can better capture the intrinsic relationship between different pressure levels and thus yield significant accuracy gain beyond the 2D counterparts. **Second**, we apply a hierarchical temporal aggregation algorithm that involves training a series of models with increasing forecast lead times (*i.e.*, 1-hour, 3-hour, 6-hour, and 24-hour forecast). Hence, in the testing stage, the number of iterations needed for medium-range (*e.g.*, 5-day) forecast is largely reduced and, consequently, the cumulative forecast errors are alleviated. Compared to previous methods (*e.g.*, FourCastNet [14] applied plain temporal aggregation with recurrent optimization), our strategy is easier to implement, more stable during training, and achieves much higher medium-range forecast accuracy.

The Pangu-Weather system is built upon a GPU cluster of Huawei Cloud with 192 NVIDIA Tesla-V100 GPUs. Each single forecast model is trained for 100 epochs which take around 15 days. To maximally support large neural networks, we use a batch size of 1 on each GPU, *i.e.*, the overall batch size is 192. With diagnostic studies, we notice that the forecast accuracy continuously goes up with a larger amount of training data and/or a longer training procedure – 100 epochs, the maximum budget that we can use, are actually insufficient for the training procedure to arrive at full convergence. That said, the community can wait for more data (including increasing the time and/or spatial resolutions) or use more powerful computational device to improve AI-based weather forecast. The trend is similar to establishing large-scale pre-trained models in other AI scopes, *e.g.*, computer vision [17], [18], natural language processing [19], [20], cross-modal understanding [21], and beyond.

Overall, the contribution of this paper are summarized in the following three aspects:

- We end the debate on whether AI-based methods can surpass NWP for global weather forecast. We establish a deep learning framework that, for the first time, surpasses operational IFS in terms of all weather factors and all forecast times from one hour to one week, meanwhile enjoying a very fast inference speed and a high spatial resolution of $0.25^\circ \times 0.25^\circ$.
- Technically, we reveal several key issues that significantly improve forecast accuracy, namely, (i) using a 3D deep network to integrate height information, and (ii) applying hierarchical temporal aggregation to alleviate cumulative forecast errors. Arguably, these techniques will be more effective in the future with more powerful computational device and higher-quality training data.
- We show that Pangu-Weather can easily transfer the ability of deterministic forecast to downstream scenarios such as extreme weather forecast and large-member ensemble forecast, where timeliness is guaranteed by its fast inference speed.

The remainder of this paper is organized as follows. Section 2 formulates the problem and briefly reviews previous work, based on which we demonstrate our technical insights. Section 3 elaborates the Pangu-Weather system with algorithmic designs and implementation details. Section 4 shows generic forecast results and investigates two specific scenarios, namely, extreme weather forecast and large-member ensemble forecast. Section 5 draws conclusions and reveals future directions.

2 Preliminaries and Insights

2.1 Problem Setting and Notations

Most weather forecast systems were built upon the analysis or reanalysis beyond observation data. The reanalysis datasets are considered the best known estimation [22], [23] for most atmospheric variables except for some factors like precipitation. Throughout this paper, we make use of the ERA5 dataset, *i.e.*, the 5th generation of ECMWF reanalysis data [24]. The ERA5 data have four dimensions, namely, latitude and longitude, pressure levels (for height) and time. We can choose an arbitrary number of weather factors (*e.g.*, geopotential, temperature, *etc.*), but do not count them toward a new dimension. With a total size of over 2PB, the dataset is split into 2D (latitude and longitude) slices to ease downloading. That said, given a time point (hourly within the past 60 years), a pressure level (or Earth's surface), and a weather factor, one can download a matrix representing the specified global reanalysis data. We denote the overall ERA5 data as $\mathbf{A}$, and we use superscripts to refer to specific weather factors and pressure levels, and subscripts to indicate spatiotemporal coordinates, *e.g.*, $\mathbf{A}_t^{\mathrm{T}850}$ stands for the global temperature data (a matrix) at time t and a height of $850\mathrm{hPa}$ and $\mathbf{A}_{x,y,t}^{\mathrm{Z}500}$ the geopotential data at position (x, y), time t, and a height of $500\mathrm{hPa}$ – note that $\mathbf{A}_{x,y,t}^{\mathrm{Z}500}$ is a single number.

Based on the ERA5 data, the problem of weather forecast is clearly defined: given an initial time t_0, the algorithm

Terms	Definition in this paper
system	The entire algorithm for end-to-end weather forecast
model	A deep network that produces one-time prediction
initial time	The time point that weather forecast is made at
forecast time	The time gap between observation and desired forecast
lead time	The time gap between input and output of one model
spacing	The minimum forecast time in a forecast system
range	The maximum forecast time in a forecast system
variable	An observed weather factor, *e.g.*, 2m temperature
parameter	A learnable value in deep networks
x, y	Horizontal coordinate (latitude & longitude)
t	Temporal coordinate (time point)
Δt	Lead time added to t
h	Height (in pressure level, *e.g.*, 500hPa)
$\mathbf{A}$	The overall weather data (*e.g.*, ERA5)
$\mathbf{A}^{\mathrm{T}h}_{x,y,t}$	Temperature at position (x, y), time t, height h
$\mathbf{A}^{*}_{t}$	All variables (all positions and heights) at time point t
$\hat{\mathbf{A}}^{*}_{t+\Delta t}$	The forecast results at time point $t + \Delta t$

TABLE 1: A summary of terminologies and notations used in this paper. In this work, we name the proposed system as Pangu-Weather and the proposed model architecture as 3D Earth-specific transformers (3DEST).

shall make use of all historical weather data (*i.e.*, $\mathbf{A}^*_t$ for $t \leqslant t_0$) to predict future weather data (*i.e.*, $\mathbf{A}^*_t$ for $t > t_0$), where $*$ stands for all factors. Before starting a survey on existing methods, we first note that the resolution of weather data is large due to the following facts. First, there are $37 \times 21 + 262$ observation factors in total (37 pressure levels, each of which has 21 weather variables, and a surface that has 262 variables), and it is believed that different elements can impact each other (*e.g.*, temperature is highly correlated to geopotential). Second, ERA5 provides about 60 years of hourly observation data, *i.e.*, the scale of time axis is over 10^5. Third, the spatial resolution is $0.25^\circ \times 0.25^\circ$, implying that each frame of global weather data is of 1440×720 numbers (*i.e.*, 'pixels' or 'voxels' if the data is to be processed by deep neural networks). As we shall see later, the high complexity has raised serious concerns on computational costs for both NWP and AI-based methods.

Based on the above definition, a weather forecast system is described as a mathematical function $f(\cdot)$ applied on $\mathbf{A}^*_t$. There are mainly two lines of research for weather forecast, which we follow the convention [8] to refer to them as NWP and AI-based methods.

2.2 NWP Methods

The first line is the conventional numerical weather prediction (NWP) methods that approximate $f(\cdot)$ using simulation. Starting with initial weather states, a set of partial differential equations (PDEs) are established to simulate different physical processes such as thermodynamics equations, N-S equations, continuous equations, *etc* [1], [25], [26]. To solve the PDEs, the atmospheric states are partitioned into discrete grids. Intuitively, reducing the spacing of grids leads to a larger number of grids and a higher spatial resolution of weather forecast, and also increases the computational costs of simulation. Currently, the spatial resolution is highly limited by the power of supercomputers. To accelerate computation, more approximation approaches were introduced, including (i) interpolation, which first performs low-resolution simulation and then estimates in-grid weather states, and (ii) parameterization [27], which uses an approximate function to solve very complex weather processes – typical examples include the parameterization for cloud [28], [29], [30] and convection [31], [32].

Prior to this work, NWP methods contribute overall the highest prediction accuracy, but they are still troubled by the super-linearly increasing computational overhead [1], [5], especially when the amount of observation data keeps growing and it is difficult to perform efficient parallelization for NWP methods [33]. The slowness of NWP not only weakens the timeliness of operational IFS systems (*e.g.*, most such systems can only update prediction several times a day), but also restricts the number of ensemble members (*i.e.*, a set of individual prediction results for ensemble), hence weakening the diversity and accuracy of probabilistic weather forecast. In addition, the formulae used by NWP methods inevitably introduce approximation and computational errors [6], [34] which can augment with either iteration or incomplete or inaccurate analysis data [35]. It thus brings major challenges to maintain NWP methods with a complex PDE system that takes more and more factors into consideration.

2.3 AI-based Methods

To alleviate the above burden, researchers started the second line that investigates AI-based methods for weather forecast. The cutting edge technology of AI lies in deep learning [36], a branch of machine learning, assuming that the complex function (*i.e.*, $f(\cdot)$) can be directly learned from abundant training data without knowing the actual physical procedure and/or formulae. Most often, $f(\cdot)$ appears as a deep neural network which is often written as $f(\cdot; \boldsymbol{\theta})$ where $\cdot$ is a placeholder for input data and $\boldsymbol{\theta}$ denotes the learnable parameters. The network often contains a number of layers. Each of these layers has a large amount of learnable parameters, and these parameters are initialized as white noise and optimized by back-propagating prediction errors of the deep network. The most similar field to weather forecast is computer vision (CV) where image data appears in 2D/3D cubes. In the past decade, the CV community developed many effective network architectures (*e.g.*, [37], [38], *etc.*), and recently, they transplanted a kind of powerful architectures named transformers from natural language processing [39] and developed the variants [40], [41] that are capable of dealing with image data.

In the scope of weather forecast, AI-based methods were first applied in the scenarios where it is difficult to predict future weather data using the NWP methods, *e.g.*, precipitation forecasting based on radar data [42], [43], [44], [45] or satellite data [46], [47]. The powerful expressive ability of deep neural networks led to the success in these data-driven environments, which further encouraged researchers to delve into the scenarios that the NWP methods are troubled by enormous computational overhead, *e.g.*, direct medium-range weather forecast [10], [12], [13], [14], [15] that consumed most of the computational resources of weather forecast centers in the past decade.

This paper investigates medium-range weather forecast. NWP and AI-based methods have been competing in this scenario, where NWP methods led in forecast accuracy [8]

and resolution, while AI-based methods showed their advantages in efficiency (*e.g.*, the inference speed is orders of magnitude faster than the NWP methods [8], [14]). Prior to 2022, AI-based methods cannot achieve the horizontal resolution of $0.25^\circ \times 0.25^\circ$ as NWP methods can. Recently, FourCastNet [14] improved the resolution to $0.25^\circ \times 0.25^\circ$, but the forecast accuracy (*e.g.*, in terms of RMSE or ACC), is still inferior to operational IFS even after a large-member ensemble was performed. The disadvantages in forecast accuracy and interpretability, especially in extreme weather events, hinder the applications of AI-based methods. Consequently, AI-based methods mostly play the role of fast surrogate models for medium-range weather forecast.

2.4 Insights

We briefly analyze the reasons why AI-based (specifically, deep learning based) methods fell behind NWP methods in terms of prediction accuracy. There are mainly two aspects, summarized as follows.

First, weather forecast shall take high-dimensional (*e.g.*, 3D spatial with 1D time), anisotropic data into consideration, yet existing AI-based methods [10], [12], [13], [14], [15] often worked on 2D (latitude and longitude) data. This brings two-fold disadvantages. On the one hand, the spacing and distribution of atmospheric states and the relationship between atmospheric patches change rapidly across pressure levels, making it difficult for 2D models to adapt to different situations. On the other hand, many weather processes (*e.g.*, radiation, convection, *etc.*) can only be completely formulated in the 3D space, and thus 2D models cannot make use of such important patterns.

Second, medium-range weather forecast can suffer from cumulative forecast errors when the model is called too many times. As an example, FourCastNet [14] trained a base model for 6-hour forecast, so that performing a 7-day forecast required executing the model 28 times iteratively. Compared to the case in NWP methods, such errors can grow rapidly because AI-based methods often do not consider real-world constraints (*e.g.*, formulated by the PDEs). According to the results in FourCastNet, the forecast error often grows super-linearly with time. Note that FourCastNet applied a specialized method for reducing iteration error, but the actual gain is somewhat limited.

Summarizing the above factors, we come up with the insights that one shall try to **increase the dimensionality of data** and **reduce the number of iterations** for more accurate medium-range weather forecast. However, this encounters difficulties in computational overhead because the weather data is very large (see Section 2.1). In the next part, we will elaborate a method built upon a tradeoff between accuracy and efficiency – in brief, we use 3D (latitude, longitude, and height) data as input and output, and train a few individual models for different prediction time gaps to maximally reduce the maximum number of iterations called for medium-range forecast.

3 Methodology

3.1 Overview

Based on the above insights, we present our system, termed Pangu-Weather, for fast and accurate global weather forecast. As an AI-based method, it surpasses the accuracy of conventional NWP methods for the first time, meanwhile enjoying a very fast inference speed.

The core part of Pangu-Weather is a set of deep neural networks trained on 39 years of global weather data – we elaborate data preparation and the pre-training task in Section 3.2. The key to reduce the accuracy loss is two-fold, namely (i) using a 3D Earth-specific transformer (3DEST) to model the 3D atmosphere effectively – see Section 3.3, and (ii) applying a hierarchical temporal aggregation strategy (*i.e.*, training a few models with various lead times) to alleviate cumulative forecast errors – see Section 3.4. The Pangu-Weather system can be applied to generic or specific forecast scenarios, as we shall see in Section 4.

3.2 Data Preparation and the Pre-training Task

We download the ERA5 dataset [24], [48], [49] from the official website[2] for training and evaluating Pangu-Weather. It contains global, hourly reanalysis data for the past 60 years. The observation data and the prediction of numerical models are blended into reanalysis data using numerical assimilation methods, providing a high-quality benchmark for global weather forecast. Following the existing methods [10], [13], [14], we train our models on a subset of ERA5 – in particular, we use the 1979–2017 (39 years of) data for training, the 2019 data for validation, and the 2018, 2020, 2021 data for testing.

We make use of observation data of every single hour so that the algorithm can perform hourly prediction. We keep the highest spatial resolution available in ERA5, namely, $0.25^\circ \times 0.25^\circ$ on Earth's sphere, resulting in an input resolution of 1440×721 (1440 for longitude and 721 for latitude – note that the northmost and southmost data do not overlap). The largest difference between our method and the prior works lies in that we formulate height information (represented as pressure levels) into the 3rd spatial dimension. To reduce computational costs, we follow [10] to choose 13 pressure levels (*i.e.*, 50hPa, 100hPa, 150hPa, 200hPa, 250hPa, 300hPa, 400hPa, 500hPa, 600hPa, 700hPa, 850hPa, 925hPa, and 1000hPa), from a total of 37 levels, plus Earth's surface. To fairly compare with the online version of ECMWF control forecast, we choose to predict the factors published in the TIGGE dataset [16], namely, five upper-air atmospheric variables (*i.e.*, geopotential, specific humidity, temperature, u-component and v-component of wind speed) and four surface weather variables (*i.e.*, 2m temperature, u-component and v-component 10m wind speed, and mean sea level pressure). In addition, three constant masks (*i.e.*, the topography mask, land-sea mask and soil type mask) are added to the input of surface variables.

The pre-training task is straightforward, *i.e.*, asking the model to predict the future weather given historical observation data. Technically, this involves sampling a time point t (*i.e.*, date and hour) from the dataset and specifying a prediction gap Δt, so that the model, $f(\cdot;\boldsymbol{\theta})$, takes $\mathbf{A}_t^*$ as input and predicts $\hat{\mathbf{A}}_{t+\Delta t}^*$, with the goal of approaching $\mathbf{A}_{t+\Delta t}^*$. In the context of deep learning, $f(\cdot;\boldsymbol{\theta})$ appears as a

2. https://cds.climate.copernicus.eu/ offered by Copernicus Climate Data (CDS).

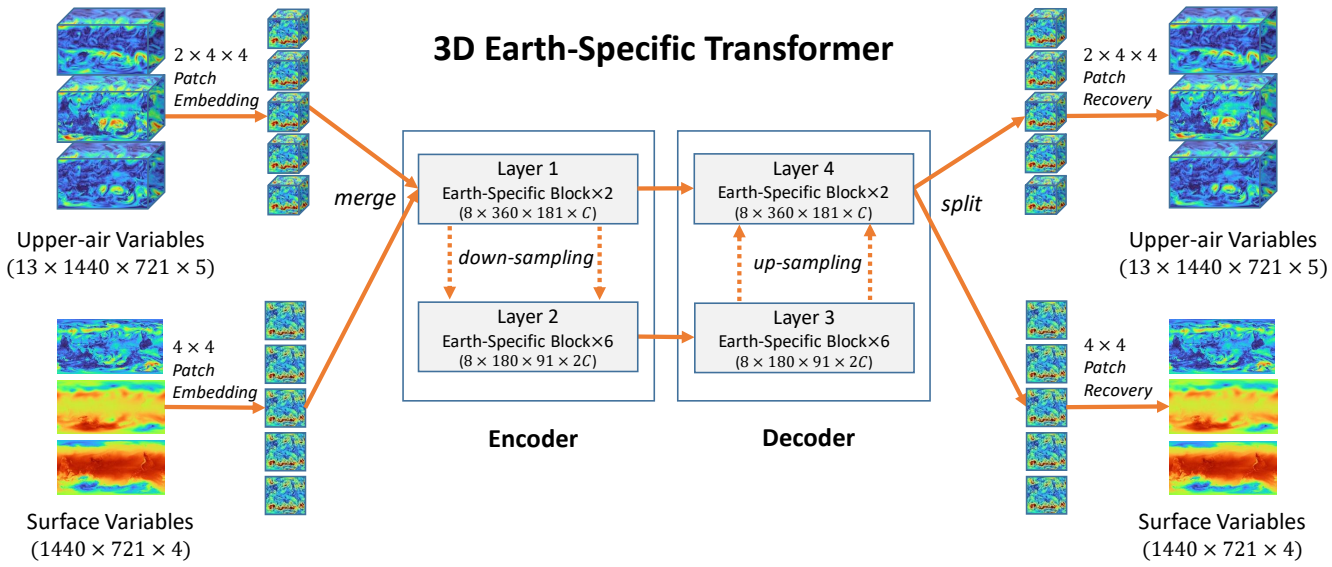

Fig. 2: An overview of the 3D Earth-specific transformer (3DEST). Based on the standard encode-decoder design, we (i) adjust the shifted-window mechanism and (ii) apply an Earth-specific positional bias – see the main texts for details.

differentiable function so that the difference between $\mathbf{A}^*_{t+\Delta t}$ and $\hat{\mathbf{A}}^*_{t+\Delta t}$ is computed and back-propagated to update the parameters, $\boldsymbol{\theta}$. The technical details, including the design of $f(\cdot;\boldsymbol{\theta})$ and the choice of Δt values, are to be elaborated in the following subsections.

3.3 3D Earth-Specific Transformer

This part describes the design of $f(\cdot,\boldsymbol{\theta})$. We name it as a 3D Earth-specific transformer (3DEST). The overall architecture of 3DEST is illustrated in Figure 2. It is a variant of vision transformer [40] with input and output being 3D weather states at a specified time point. For a single model, the lead time between input and output states is fixed, *e.g.*, Δt equals to 6 hours. We achieve any-time weather forecast by aggregating multiple models with different lead times, as elaborated in the next subsection.

There are two sources of input and output data, namely, upper-air variables and surface variables. The former involves 13 pressure levels, and they combined offer a $13 \times 1440 \times 721 \times 5$ data cube. The latter contains a $1440 \times 721 \times 4$ cube. These parameters are first embedded from the original space into a C-dimensional latent space. A common technique in computer vision named patch embedding is used for dimensionality reduction. For the upper-air part, the patch size is $2 \times 4 \times 4$, so that the embedded data has a shape of $7 \times 360 \times 181 \times C$. For the surface variables, the patch size is 4×4, so that the embedded data has a shape of $360 \times 181 \times C$. These two data cubes are then concatenated along the first (height) dimension to yield a $8 \times 360 \times 181 \times C$ cube. The cube is then propagated through a standard encoder-decoder architecture with 8 encoder layers and 8 decoder layers. The output of decoder is still a $8 \times 360 \times 181 \times C$ cube, which is projected to the original space with patch recovery, producing the desired output. Below, we describe the technical details of each component.

Patch embedding and patch recovery. We follow the standard vision transformer to use a linear layer with GeLU activation for this purpose. In our implementation, a patch has $2 \times 4 \times 4$ pixels for upper-air variables and 4×4 for surface variables. The stride of sliding windows is the same as patch size, and necessary zero-value padding is added when the data size is indivisible by the patch size. The number of parameters for patch embedding is $(4 \times 4 \times 2 \times 5) \times C$ for upper-air variables and $(4 \times 4 \times 4) \times C$ for surface variables. Patch recovery performs the opposite operation, but it does not share parameters with patch embedding.

The encoder-decoder architecture. The data size remains unchanged ($8 \times 360 \times 181 \times C$) for the first 2 encoder layers, while for the next 6 layers, the horizontal dimensions are reduced by a factor of 2 and the number of channels is doubled, resulting in a data size of $8 \times 180 \times 91 \times 2C$. The decoder part is symmetric to the encoder part, with the first 6 decoder layers sized $8 \times 180 \times 91 \times 2C$ and the next 2 layers sized $8 \times 360 \times 181 \times C$. The outputs of the 2nd encoder layer and the 7th decoder layer are concatenated along the channel dimension. Down-sampling and up-sampling operations connect the adjacent layers of different resolutions, and we follow the implementation of Swin transformers [41]. For down-sampling, we merge four tokens into one (the feature dimensionality increases from C to $4C$) and perform a linear layer to reduce the dimensionality to $2C$. For up-sampling, the reverse operations are performed.

3D Earth-specific transformer blocks. Each encoder and decoder layer is a 3D Earth-specific transformer (3DEST) block. It is similar to the standard vision transformer block [40] but specifically designed to align with Earth's geometry. To further reduce computational costs, we inherit the window-attention mechanism [41] to partition the feature maps (either $8 \times 360 \times 181$ or $8 \times 180 \times 91$ – the last dimension is omitted) into windows, and each window contains up to $2 \times 12 \times 6$ tokens. The standard self-attention mechanism is applied within each window. The shifted-window attention mechanism is applied, so that for

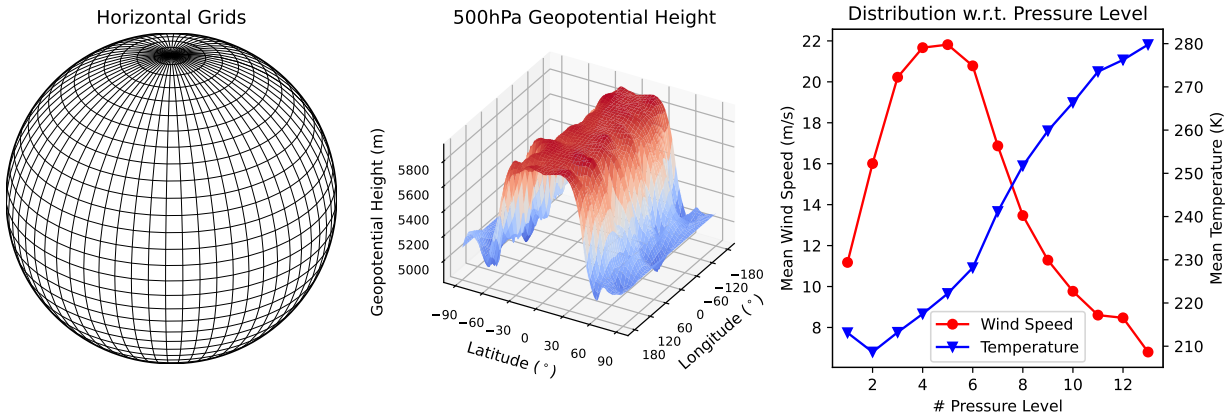

Fig. 3: The motivation of using an Earth-specific positional bias. **Left**: the horizontal map corresponds to an uneven spatial distribution on Earth's sphere. **Middle**: the geopotential height is closely related to the latitude. **Right**: the mean wind speed and temperature are closely related to the height (formulated as pressure levels).

every layer, the grid partition differs from the previous one by half window size[3]. We refer the reader to the original paper [41] for more details. The standard self-attention formula is written below:

$$\text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{SoftMax}(\mathbf{Q}\mathbf{K}^\top/\sqrt{D}+\mathbf{B})\mathbf{V}, \quad (1)$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are the query, key, and value vectors produced by the transformer block, respectively, D is the feature dimensionality of $\mathbf{Q}$ and $\mathbf{K}$ (*i.e.*, C or $2C$), and $\mathbf{B}$ is the positional bias term.

Earth-specific positional bias. Swin transformer used a relative positional bias to represent the translation invariant component of attentions, where the bias is computed upon the relative coordinate of each window. For global weather forecast, however, the situation is a bit different. Each token corresponds to an absolute position on Earth's coordinate system and, since the map is a projection of Earth's sphere, the spacing between neighboring tokens can be different – see Figure 3. More importantly, some weather states are closely related to the absolute position. Examples of geopotential, wind speed, and temperature are shown in Figure 3. To capture these properties, we modify $\mathbf{B}$ into an Earth-specific positional bias, termed $\mathbf{B}_{\text{ESP}}$, adding a positional bias to each token based on its absolute (rather than relative) coordinate.

Mathematically, let the entire feature map have a spatial resolution of $N_{\text{pl}} \times N_{\text{lat}} \times N_{\text{lon}}$ where N_{pl}, N_{lat}, and N_{lon} indicate the size along the axes of height (by pressure levels), latitude, and longitude, respectively. Swin transformer partitions these neurons into $M_{\text{pl}} \times M_{\text{lat}} \times M_{\text{lon}}$ windows, and each window has a size of $W_{\text{pl}} \times W_{\text{lat}} \times W_{\text{lon}}$. The Earth-specific position bias matrix contains $M_{\text{pl}} \times M_{\text{lat}}$ sub-matrices (M_{lon} does not appear because different longitudes share the same bias – the longitude indices are cyclic and spacing is evenly distributed along this axis), each of which contains $W_{\text{pl}}^2 \times W_{\text{lat}}^2 \times (2W_{\text{lon}}-1)$ learnable parameters. When the attention is computed between two units within the same window (Swin does not compute inter-window attentions), we use the window coordinate $(m_{\text{pl}}, m_{\text{lat}}, m_{\text{lon}})$ to locate the corresponding bias sub-matrix (m_{lon} is not used), and then use the intra-window coordinates, $(h'_1, \phi'_1, \lambda'_1)$ and $(h'_2, \phi'_2, \lambda'_2)$, to call for the bias value at $(h'_1 + h'_2 \times W_{\text{pl}}, \phi'_1 + \phi'_2 \times W_{\text{lat}}, \lambda'_1 - \lambda'_2 + W_{\text{lon}} - 1)$ of the sub-matrix.

Applying the Earth-specific positional bias brings two-fold differences. **First**, it enables a better formulation of Earth's atmosphere: In every attention block, the Earth-specific positional bias learns different spatial relationship between tokens for different latitudes and heights, hence correcting the non-uniformity brought by the uneven spatial distribution. **Second**, compared to the original version where all grids share the same bias, the number of learnable parameters of each transformer layer is largely increased from $(2W_{\text{pl}}-1) \times 2(W_{\text{lat}}-1) \times (2W_{\text{lon}}-1)$ to $M_{\text{pl}} \times M_{\text{lat}} \times W_{\text{pl}}^2 \times W_{\text{lat}}^2 \times (2W_{\text{lon}}-1)$. In the first block, the latter quantity is about $527\times$ larger than the former one. The huge amount of bias parameters allows each block to flexibly learn specific patterns for each variable, such as the relationship shown in Figure 3. In practice, we do not observe any difficulties in optimizing the large amount of parameters. Instead, the model converges faster in the training process since useful priors have been introduced. In addition, the Earth-specific positional bias does not increase the FLOPs of the model.

Design choices. We briefly discuss other design choices. Due to the large computational overhead, we do not perform exhaustive ablative or diagnostic studies on the hyper-parameters and we believe there exist configurations that lead to higher accuracy. **First**, we use 8 $(2+6)$ encoder and decoder layers, which is significantly fewer than the standard Swin transformer. This is to reduce the complexity in both time and memory. If one has a larger GPU memory and a more powerful cluster, increasing the network depth can lead to higher accuracy. **Second**, it is possible to reduce the number of parameters used in the Earth-specific positional bias by parameter sharing or other techniques. However, we do not consider it as a key issue,

3. Note that, along the longitude dimension, the leftmost and rightmost indices are actually close to each other. In the shifted-window mechanism, if half windows appear at both leftmost and rightmost positions, they are directly merged into one window.

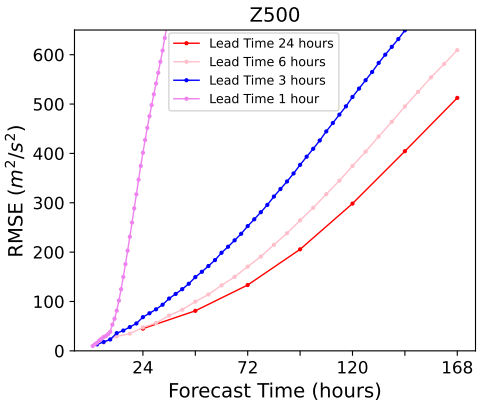

Fig. 4: The curves showing cumulative forecast errors when one performs up to 7-day forecast with the base lead time being 1 hour, 3 hour, 6 hours, and 24 hours, respectively. The statistics are performed in the March 2018 subset.

because it is unlikely to deploy the weather forecast model to edge device with limited storage. **Third**, it is possible and promising to feed the weather states of more time points into the model, which changes all tensors from 3D to 4D. While we believe such a modification can lead to accuracy gain, the limited computational budget prevents us from this trial.

3.4 Hierarchical Temporal Aggregation

When the goal is to make medium-range weather forecast (*e.g.*, the forecast time is up to 5 days) yet the lead time of the basic forecast model is relatively short (*e.g.*, FourCastNet trained a model with a lead time of 6 hours), the system must execute the model many times iteratively, and the cumulative forecast errors can grow continuously. As shown in Figure 4, we mimic FourCastNet [14] to execute the 6-hour model 28 times to achieve up to 7-day forecast, and we find that the forecast accuracy rapidly goes down as iteration goes on. Not surprisingly, the forecast accuracy drop becomes dramatic if the basic lead time is set to be 1 hour (*i.e.*, the model is executed 168 times), yet the drop is largely alleviated if the lead time is 24 hours (*i.e.*, executed 7 times). This implies that, for medium-range and even long-range forecast, the system can benefit much from suppressing the cumulative forecast errors.

For this purpose, we exploit a straightforward yet effective strategy named hierarchical temporal aggregation. We train four individual models for 1-hour, 3-hour, 6-hour, and 24-hour prediction, respectively. We do not continue enlarging the lead time, because it largely increases the difficulty of training the base model[4]. At the testing stage, given a forecast goal, we use the greedy algorithm to guarantee the minimal number of iterations. For example, for 7-day forecast, we execute 24-hour forecast 7 times, while for a 23-hour forecast, we execute 6-hour forecast 3 times, followed by 3-hour forecast 1 time and 1-hour forecast 2 times.

4. We find that, based on the current deep network, it is difficult to perform long-term (say, 28-day) forecast. We conjecture that, if more powerful methods are used (*e.g.*, using time-aware inputs, increasing the computational complexity, *etc.*), the model may gain such abilities.

We point out that hierarchical temporal aggregation makes both the training and testing stages more efficient. For training, it avoids performing recursive optimization as many existing works [11], [12], [14] did, *e.g.*, FourCastNet [14] computed both $f(\mathbf{A})$ and $f(f(\mathbf{A}))$ and produced two loss terms – although the iterative errors are indeed suppressed, it requires $2\times$ GPU memory for the same model and thus reduced the model size which is one of the critical factors of improvement. In addition, it avoids training a recursive neural network which may be unstable. For testing, especially when the forecast range is large, it reduces the number of forecasts as well as the time complexity.

The four individual models are trained for 100 epochs using the Adam optimizer. Each full training procedure takes 16 days on 192 NVIDIA Tesla-V100 GPUs. We find that all models have not yet arrived at full convergence at the end of 100 epochs, but the limited computational budget prevents us from continuing the training procedure. A weight decay of 3×10^{-6} and a scheduled DropPath with a drop ratio of 0.2 are adopted to avoid over-fitting.

4 Results

We report the forecast results of Pangu-Weather on two datasets. The first one is the held-out part of ERA5 for an overall evaluation of global, deterministic weather forecast. The second one is the 4th version of International Best Track Archive for Climate Stewardship (IBTrACS) dataset for evaluating the ability at tracking tropical cyclones, a special case of extreme weather forecast.

We compare Pangu-Weather to the strongest methods in both worlds of NWP and AI, namely, operational IFS offered by ECMWF (downloaded from the TIGGE archive [16])[5] and FourCastNet [14]. For tropical cyclones tracking, we also download ECMWF-HRES forecast as a stronger competitor against Pangu-Weather in IBTrACS. To the best of our knowledge, no prior AI-based methods have ever reported quantitative results for tropical cyclones tracking.

4.1 Deterministic Forecast

The deterministic forecast of Pangu-Weather is performed on the unperturbed initial states from ERA5. The forecast resolution of Pangu-Weather is determined by the training data (*i.e.*, ERA5), where the spatial resolution is $0.25^\circ \times 0.25^\circ$, comparable to the control forecast of ECMWF ENS product [3] and same as FourCastNet [14], yet the spacing of forecast (the minimal forecast time) is 1 hour (*i.e.*, Pangu-Weather can provide hour-by-hour forecast), $6\times$ smaller than that of FourCastNet [14].

Following the prior AI-based methods, the accuracy of deterministic forecast is computed by two quantitative metrics, namely, the latitude-weighted Root Mean Square Error (RMSE) and latitude-weighted Anomaly Correlation Coefficient (ACC). For a specified time point t, the RMSE

5. We failed to download part of forecast results of the surface variables from TIGGE, due to the unavailability of ECMWF's Data Handling Systems from September to November, so we compare our results to operational IFS by (i) fetching the numbers reported in WeatherBench [10] and (ii) extracting the quantities from the plots in the FourCastNet paper [14].

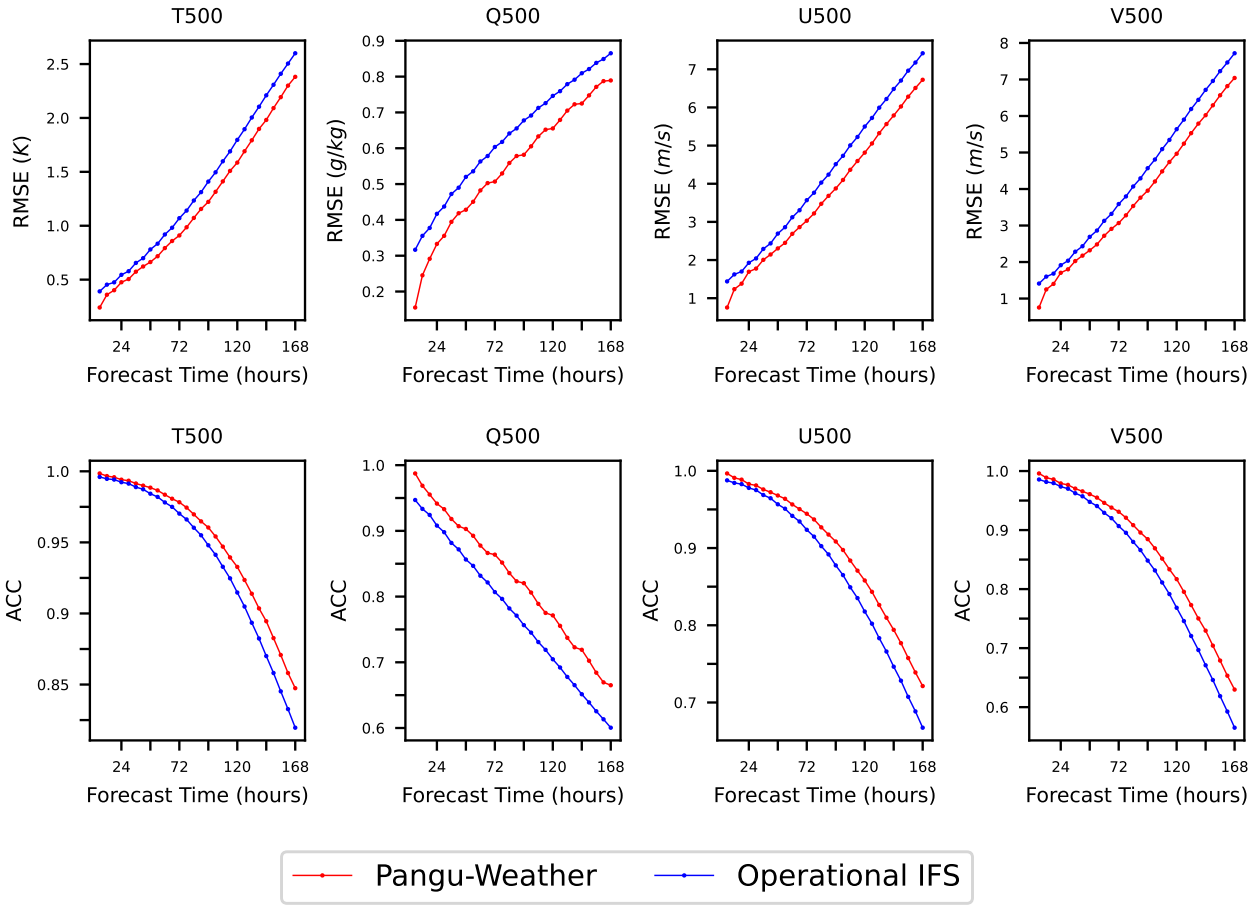

Fig. 5: The comparison of forecast accuracy in terms of latitude-weighted RMSE (lower is better) and ACC (higher is better) of four upper-air variables at the pressure level of 500hPa. Here, T, Q, U and V stand for temperature, specific humidity, u-component and v-component of wind speed, respectively.

and ACC of any variable v (*e.g.*, 2m temperature or 500hPa geopotential) are defined as follows:

$$\mathrm{RMSE}(v,t)=\sqrt{\frac{\sum_{i=1}^{N_{\mathrm{lat}}}\sum_{j=1}^{N_{\mathrm{lon}}}L(i)(\hat{\mathbf{A}}_{i,j,t}^{v}-\mathbf{A}_{i,j,t}^{v})^2}{N_{\mathrm{lat}}\times N_{\mathrm{lon}}}}, \quad (2)$$

$$\mathrm{ACC}(v,t)=\frac{\sum_{i,j}L(i)\hat{\mathbf{A}}_{i,j,t}^{\prime v}\mathbf{A}_{i,j,t}^{\prime v}}{\sqrt{\sum_{i,j}L(i)(\hat{\mathbf{A}}_{i,j,t}^{\prime v})^2\times\sum_{i,j}L(i)(\mathbf{A}_{i,j,t}^{\prime v})^2}}, \quad (3)$$

where $L(i)=N_{\mathrm{lat}}\times\frac{\cos\phi_i}{\sum_{i'=1}^{N_{\mathrm{lat}}}\cos\phi_{i'}}$ stands for the weight at latitude ϕ_i and $\mathbf{A}'$ denotes the difference between $\mathbf{A}$ and the climatology (*i.e.*, long-term mean of weather states, which is estimated on the training data over 39 years). Note that we omitted the range of summation in Eqn (3) for simplicity. In what follows, we report these two metrics on upper-air atmospheric variables and surface weather variables to show the superiority of Pangu-Weather. We also provide extensive visualization and diagnostic results for qualitative studies.

4.1.1 Upper-air Atmospheric Variables

As in the training procedure (see Section 3.2 for data preparation), Pangu-Weather forecasts five important upper-air variables (*i.e.*, geopotential, specific humidity, temperature, u-component and v-component of wind speed) at 13 pressure levels (*i.e.*, 50hPa, 100hPa, 150hPa, 200hPa, 250hPa, 300hPa, 400hPa, 500hPa, 600hPa, 700hPa, 850hPa, 925hPa, and 1000hPa), with a spatial resolution of $0.25^\circ\times 0.25^\circ$. This is to maximally ease the comparison to operational IFS [16] and FourCastNet [14], the best NWP and AI-based methods.

The testing environment is established on the weather data in 2018[6]. Following the protocol of operational IFS, we choose 2 time points (00:00 UTC and 12:00 UTC) each day as the initial time[7] and produce hourly forecast for the upcoming week, namely, forecast time being $1\mathrm{h}, 2\mathrm{h}, \ldots, 168\mathrm{h}=7\mathrm{d}$. Quantitative comparisons are mainly made between Pangu-Weather and operational IFS, while the comparisons against other AI-based methods (*e.g.*, [10], [11], [12], [13], [14], [15]) are incomplete due to the difference in spatial resolutions,

6. We also test our system in the 2020 and 2021 data, while we cannot provide comparative results since no prior works have reported results on these data. The property of forecast results is mostly similar to that observed in the 2018 data.

7. The test points on Jan 1st, 2018 are excluded due to the overlap with training data. All test points in December 2018 are unavailable due to a server error of ECMWF. In addition, for the T850 variable, all test points in October 2018 are not used due to an unexpected error of data download from the TIGGE archive.

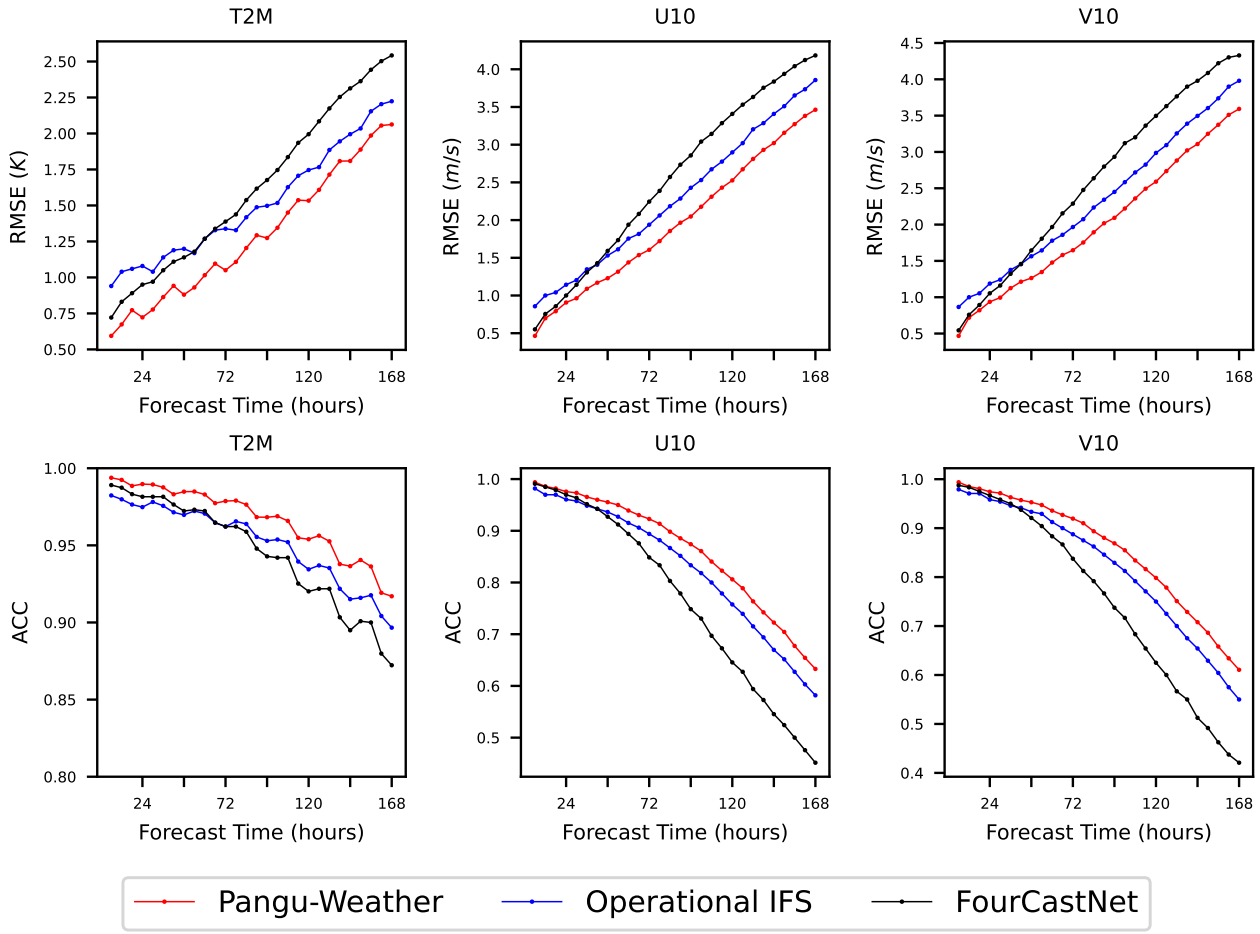

Fig. 6: The comparison of forecast accuracy in terms of latitude-weighted RMSE (lower is the better) and ACC (higher is the better) of three surface variables. Here, T2M, U10, V10 stand for 2m temperature, u-component and v-component of 10m wind speed, respectively.

test subsets, post-processing methods, *etc*. Note that all prior AI-based methods reported inferior forecast accuracy compared to operational IFS, while our method significantly outperforms operational IFS, claiming clear advantages over these candidates. For two specific variables, Z500 and T850, we directly compare Pangu-Weather against FourCastNet by fetching the numerical values from the plots in the paper – this introduces some errors which are negligible compared to the accuracy gap between Pangu-Weather and FourCastNet.

The comparative results between Pangu-Weather and operational IFS are shown in Figure 1 (top) and Figure 5, where Pangu-Weather enjoys consistent gains (in **all** forecast times and for all variables) in forecast accuracy compared to operational IFS. The advantage becomes more significant as forecast time increases, implying that AI-based methods are better at capturing effective (though non-interpretable) patterns for medium-range weather forecast. Specifically, we note that the 'forecast time gain' of Pangu-Weather over operational IFS (*i.e.*, the difference between forecast times at the same forecast accuracy) is more than 12 hours for all variables and more than 24 hours for specific humidity – this implies that AI-based methods are significantly better at forecasting specific variables.

Specifically, we investigate 500hPa geopotential (Z500) and 850hPa temperature (T850), the variables that were widely reported in prior AI-based methods. The quantitative comparison for these two variables is shown in Figure 1. As shown, the forecast accuracy of Pangu-Weather is consistently higher than that of operational IFS and FourCastNet, the previous best AI-based method (yet weaker than operational IFS). Quantitatively, for Z500, the 3-day and 5-day RMSEs (in m^2/s^2) of operational IFS are 152.8 and 333.7, respectively, and Pangu-Weather reduces them to 134.5 and 296.7 (133.9 and 294 if the December 2018 data are included). For T850, the 3-day and 5-day RMSEs (in K) of operational IFS are 1.37 and 2.06, respectively, and Pangu-Weather reduces them to 1.14 and 1.79 (1.13 and 1.77 if the December 2018 data are included), claiming an over 10% relative error drop. The relative drop of RMSE is more than 10% in all scenarios, which also reflects in a 'forecast time gain' of 10–15 hours. When compared to FourCastNet, we observe even more significant accuracy gains – the relative reduction of RMSE is more than 30% in the above scenarios, and the 'forecast time gain' is also enlarged to more than 36 hours.

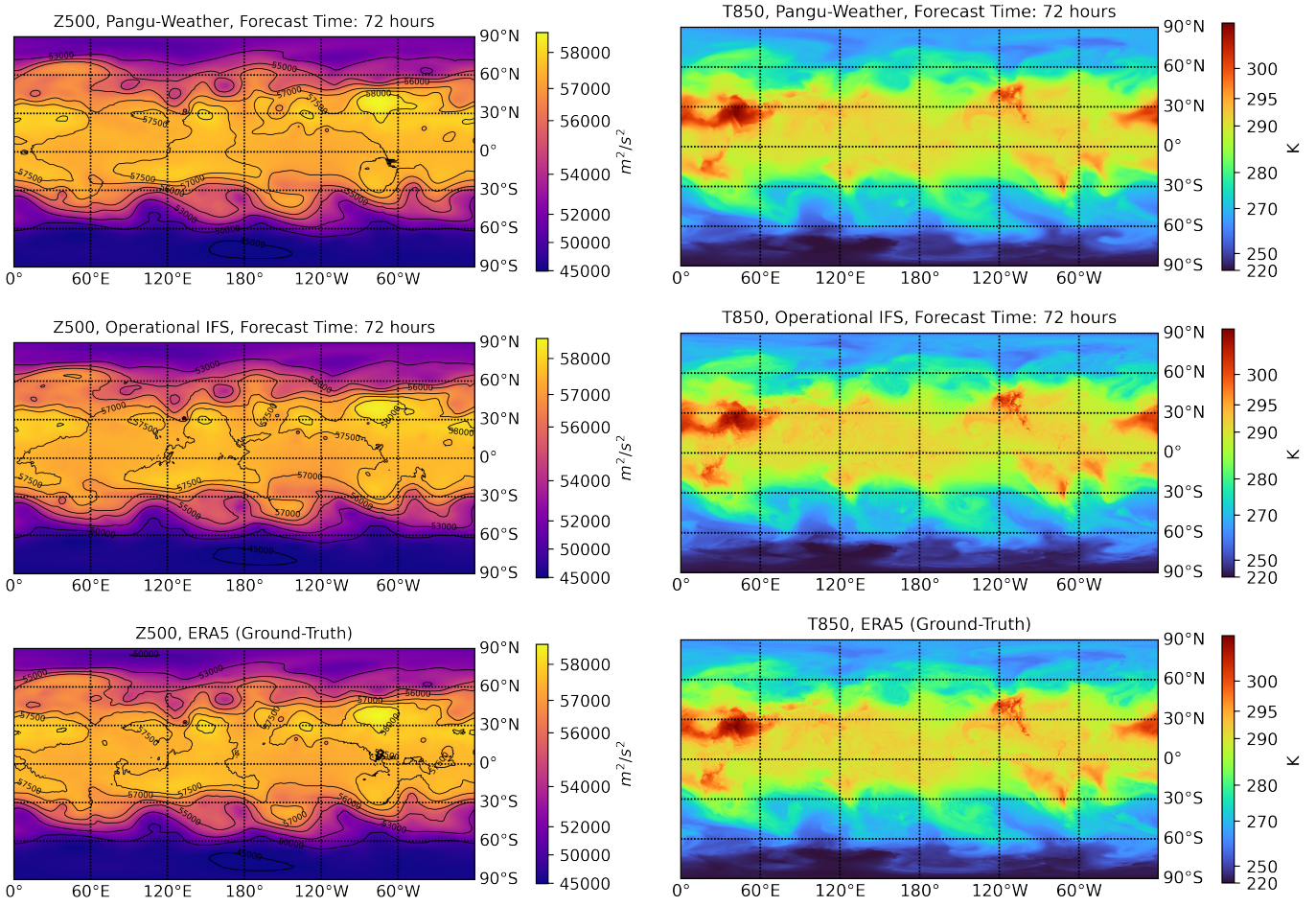

Fig. 7: Visualization of 3-day weather forecast produced by Pangu-Weather (top), operational IFS (middle), and the ERA5 ground-truth (bottom). The left and right columns show the maps of 500hPa geopotential (Z500) and 850hPa temperature (T850), respectively. The input time point (*i.e.* the forecast is performed on) is 00:00 UTC, September 1st, 2018.

4.1.2 Surface Weather Variables

Pangu-Weather forecasts four important surface variables, *i.e.*, 2m temperature, u-component and v-component of 10m wind speed, and mean sea level pressure. Compared to the upper-air variables, these surface variables have close and complex relationship to topography and human activities (*e.g.*, urban heat island effect) and thus are more difficult to forecast. The testing environment is established in a similar way of forecasting upper-air variables[8]. Quantitative comparisons are made between Pangu-Weather and previous best NWP method (*i.e.*, operational IFS) and AI-based method (*i.e.*, FourCastNet [14]), where the numerical results of FourCastNet and operational IFS are fetched from the plots in the paper[9]. The comparative results are shown in Figure 6. Again, Pangu-Weather outperforms both competitors in terms of forecast accuracy in **all** forecast times, for all variables, and the advantage becomes more significant as forecast time increases. The 'forecast time gain' of all these variables is about 18 hours, slightly longer than the gain in forecasting upper-air variables.

We investigate the forecast accuracy of separate variables. For 2m temperature (T2M), the 3-day and 5-day RMSEs (in K) are 1.34 and 1.75 for operational IFS, 1.39 and 2.00 for FourCastNet, and Pangu-Weather reduces them to 1.05 and 1.53 (1.06 and 1.52 with a 6-hour test interval), respectively. For u-component of 10m wind speed (U10), the 3-day and 5-day RMSEs (in m/s) are 1.94 and 2.90 for operational IFS, 2.24 and 3.41 for FourCastNet, and Pangu-Weather reduces them to 1.61 and 2.53 (1.61 and 2.55 for the test data in 2018 with a 6-hour interval), respectively. We omit the numerical comparison for v-component of 10m wind speed (V10) since it is almost the same as that of U10.

We have the forecast results for the variable of mean sea level pressure (MSLP) but we cannot provide the quantitative comparison due to the unavailability of data from the ECMWF server. We believe that our forecast is the best candidate because of two reasons. On the one hand, According to prior experiences [50], a model that better forecasts on other surface variables (*i.e.*, T2M, U10, V10) also enjoys a higher forecast accuracy on MSLP. On the other hand, we make use of our forecast of MSLP for tracking tropical cy-

8. The test points on Jan 1st, 2018 are excluded due to the overlap with training data.

9. For a fair comparison to FourCastNet, we follow the protocol to set the test interval (the gap between neighborhood test time points) to be 9 days for T2M and 2 days for U10 and V10, albeit we can produce forecast results every single hour. We also report the RMSE values using a fixed 6-hour interval in the following part.

clones – as shown in Section 4.2.2, Pangu-Weather achieves much better results, quantitatively and qualitatively, than operational IFS in forecasting 88 named tropical cyclones in the year of 2018.

In addition, we evaluate Pangu-Weather on WeatherBench [10], a benchmark for low-resolution weather forecast. For this purpose, we simply down-sample the forecast results of Pangu-Weather by $22.5\times$ into a coarse grid with a spatial resolution of $5.625^\circ \times 5.625^\circ$, and compare the results to the down-sampled ERA5 ground-truth. Quantitatively, operational IFS reported 3-day/5-day RMSEs of T2M[10] being $1.35/1.77$ on WeatherBench, and Pangu-Weather improves the results to $1.04/1.51$, respectively.

4.1.3 Visualization

In Figure 7, we first visualize the 72-hour forecast of Pangu-Weather on two upper-air variables, namely, Z500 and T850, and compare the results to operational IFS and the ERA5 ground-truth. Both forecast results are sufficiently close to the ground-truth, yet one can detect the differences between them. Pangu-Weather produce smoother contour lines, implying that the model tends to forecast similar values for neighboring regions – this is a typical property of deep neural networks in learning from large-scale datasets. In comparison, operational IFS tends to preserve small-scale structures, yet such predictions are not guaranteed to be correct. As shown in the previous part, Pangu-Weather enjoys the advantage of overall forecast accuracy.

In Figure 1 (top left), we also visualize the 72-hour forecast of Pangu-Weather on two surface variables, namely, 2m temperature (T2M) and 10m wind speed ($\sqrt{u^2+v^2}$). As can be seen, Pangu-Weather produces high-resolution forecasts that are (i) very close to the ERA5 ground-truth (also refer to the previous part for quantitative results) and (ii) sufficient to preserve most of small-scale structures of surface variables.

4.1.4 Diagnostic Studies

We investigate the monthly averaged 5-day latitude-weighted ACC of four upper-air variables, namely, geopotential (Z), specific humidity (Q), temperature (T), and u-component of wind speed (U), all at the pressure level of $500\mathrm{hPa}$. The comparison between Pangu-Weather and operational IFS is shown in Figure 1 (top right). Pangu-Weather outperforms operational IFS in every single month, demonstrating the stability of forecast. More importantly, the advantage of Pangu-Weather becomes more significant in the worst performed months (*e.g.*, April and May), implying that AI-based methods have learned useful and complementary knowledge from large data. We conjecture that such knowledge may correspond to (i) unknown or unformulated atmospheric procedures or (ii) better manipulations with missing factors. Studying these factors may be an interesting topic for meteorologists.

A clear advantage of Pangu-Weather lies in its ability of performing hourly weather forecast. We plot the hourly RMSE and ACC values for two upper-air variables (Z500 and Q500) and two surface variables (T2M and U10) in Figure 8. Note that we have applied a greedy algorithm based on hierarchical temporal aggregation as elaborated in Section 3.4. For some variables (*e.g.*, Q500 and T2M), we observe a clear trend that forecast accuracy drops with the number of iterations, *e.g.*, 3 calls are required for a 72-hour forecast, while 8 calls are required for a 71-hour forecast (*i.e.*, $71 = 24 + 24 + 6 + 6 + 6 + 3 + 1 + 1$), and thus the 71-hour accuracy is much lower due to cumulative forecast errors. While we can easily improve the accuracy by moving the time point back (*e.g.*, performing 72-hour forecast using 1-hour-earlier weather states for 71-hour forecast), we just offer the original forecast results here to show this important phenomenon. This calls for an advanced temporal aggregation algorithm in the future – one possibility lies in integrating the time axis into input data and making use of 4D deep neural networks, but this implies much heavier computational overheads.

4.1.5 Computational Costs

A clear advantage of AI-based methods lies in the inference speed. FourCastNet [14] claimed a $45{,}000\times$ speedup over the traditional NWP method, and Pangu-Weather is comparable with FourCastNet (see the next paragraph). Considering the advantage in forecast accuracy, Pangu-Weather has the potentials of replacing conventional NWP, enabling real-time weather forecast to be performed any time (*e.g.*, once a second), rather than the current status that weather forecast is performed merely a few times per day. A few side benefits are expected. (i) It largely increases the timeliness of short-range weather forecast which is important in warning about short-term extreme weathers, *e.g.*, cloudbursts. (ii) It enables large-member ensemble forecast which is important for meteorologists to pay attentions to the sensitive weather factors or variables.

The inference speed of Pangu-Weather is comparable to that of FourCastNet [14], implying that using holistic 3D deep neural networks for inference is slightly more costly than using 2D counterparts, yet the accuracy is much higher. In a system-level comparison, FourCastNet requires $280\mathrm{ms}$ for inferring a 24-hour forecast on an NVIDIA Tesla-A100 GPU (312 TeraFLOPS), while Pangu-Weather needs $1{,}400\mathrm{ms}$ on an NVIDIA Tesla-V100 GPU (120 TeraFLOPS). Taking GPU performance into consideration, Pangu-Weather is about 50% slower than FourCastNet, while still being one of the fastest systems for high-resolution, global weather forecast.

To further accelerate Pangu-Weather, we can train models with larger lead times (*e.g.*, 72 hours) so as to reduce the number of temporal aggregations. We expect to explore this direction in the near future.

4.2 Results on Extreme Weather Events

Extreme weather forecast plays a vital role of global weather forecast. Despite rare occurrence, extreme weather events like hurricanes can bring tremendous casualty and economical loss. Therefore, it is expected that weather forecast systems can warn about upcoming extreme weather events that complement daily weather reports.

In this subsection, we investigate the ability of Pangu-Weather in forecasting extreme weather events and compare

10. We only show the comparison on T2M, because WeatherBench evaluated the forecast results of T850 and Z500 on the 2017 subset which is part of our training data.

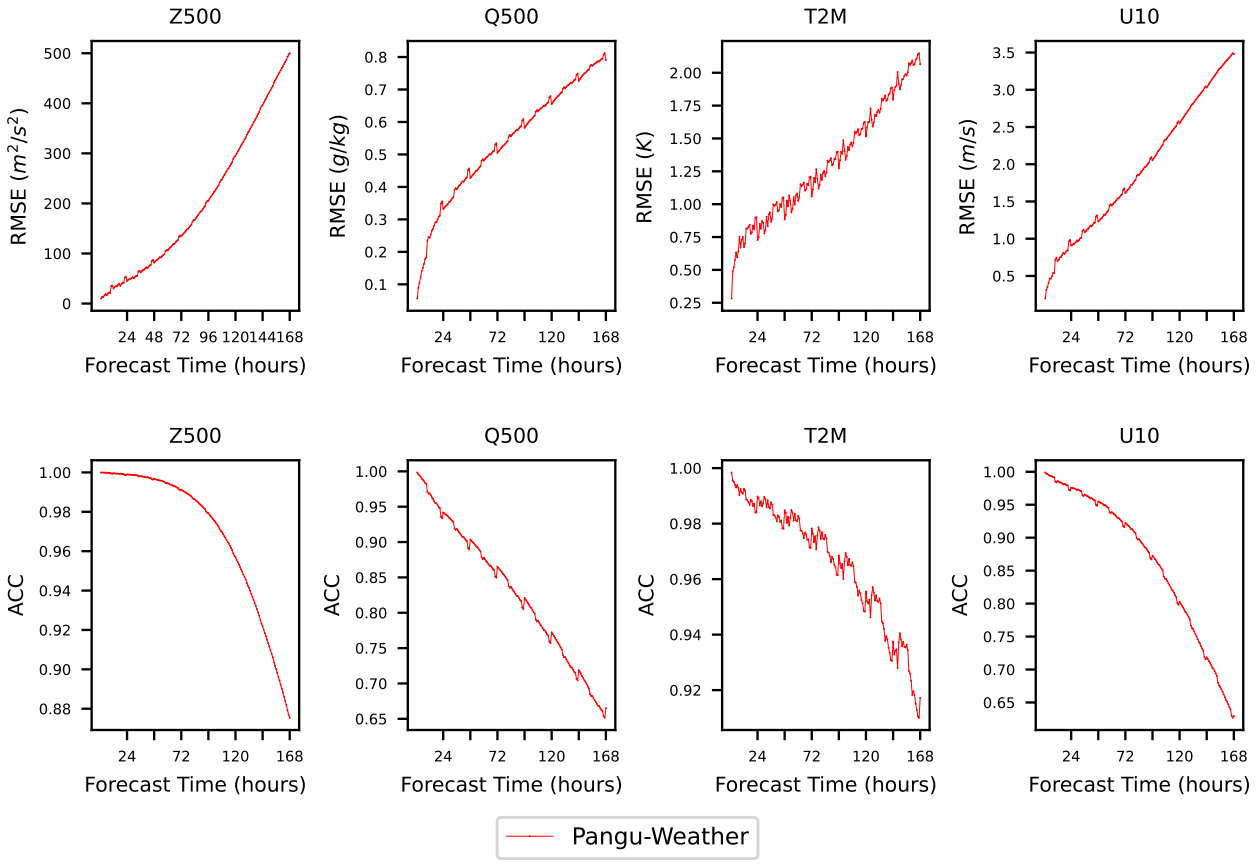

Fig. 8: Hourly forecast results for two upper-air variables (500hPa geopotential, Z500, and 500hPa specific humidity, Q500) and two surface variables (2m temperature, T2M, and u-component of 10m wind speed, U10). The forecast time ranges from 1 hour to 7 days (168 hour), and both latitude-weighted RMSE (lower is better) and ACC (higher is better) are reported. The input time points are chosen from the 2018 data – since no comparison is made, we only exclude the data on January 1st due to the overlaps with the training set.

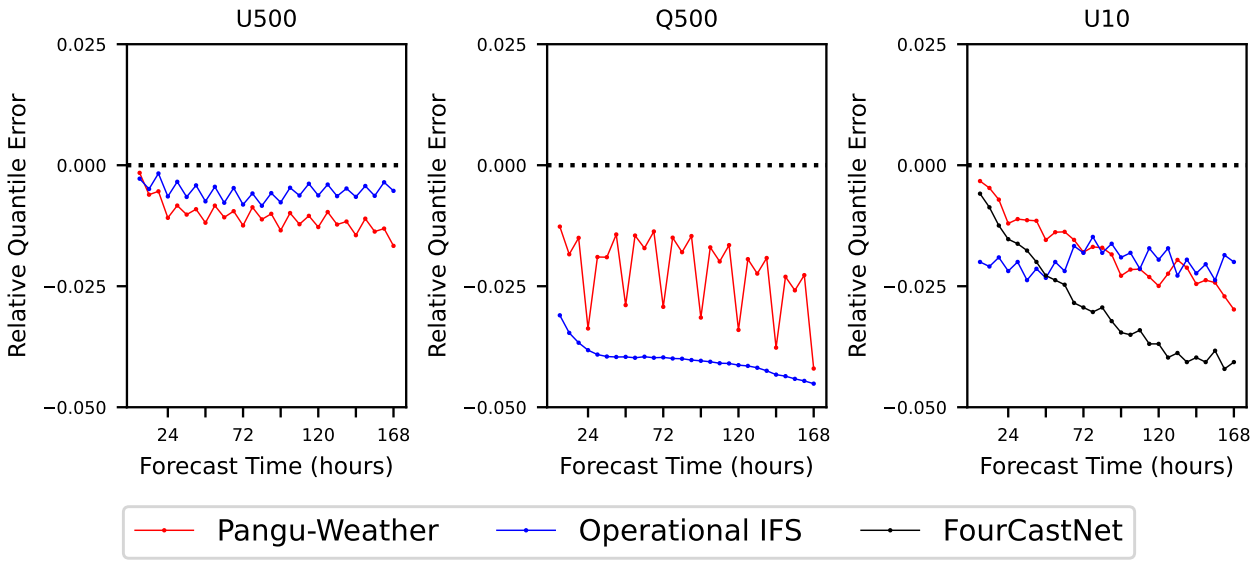

Fig. 9: Plots of RQE values with respect to forecast time for two upper-air variables (500hPa u-component of wind speed, U500, and 500hPa specific humidity, Q500) and one surface variables (u-component of 10m wind speed, U10).

the ability to that of conventional NWP methods, *i.e.*, operational IFS. We first introduce a quantitative metric named relative quantile error to measure the overall tendency in Section 4.2.1, and then study a special and important case, *i.e.*, tracking tropical cyclones, in Section 4.2.2.

4.2.1 Overall Tendency in Predictions of Extremes

We use a similar approach to [51] to compare the values of top-level quantiles calculated on the forecast result and ground-truth. Mathematically, we set $D = 50$ percentiles, denoted as $q_1, \ldots, q_D$. Following FourCastNet [14], we set $q_1 = 90\%$, $q_D = 99.99\%$, and the intermediate ones are linearly distributed between q_1 and q_D in the logarithmic scale. Then, the corresponding quantiles, denoted as $Q_1, \ldots, Q_D$, are computed individually for each pair of weather variable and forecast time, *e.g.*, for all 3-day forecasts of U10, pixel-wise values are summarized from all frames for statistics. Finally, the relative quantile error (RQE) is used for measuring the difference between the ground-truth and any weather forecast system:

$$\mathrm{RQE} = \sum_{d=1}^{D} \frac{\hat{Q}_d - Q_d}{Q_d}, \tag{4}$$

where Q_d and $\hat{Q}_d$ are different versions of the d-th quantile calculated on the ERA5 ground-truth and the system being investigated, *e.g.*, Pangu-Weather. RQE can measure the overall tendency, where $\mathrm{RQE} < 0$ and $\mathrm{RQE} > 0$ imply that the forecast system tends to underestimate and overestimate the intensity of extremes, respectively.

In Figure 9, we plot the RQE values for two upper-air variables (U500, V500) and one surface variable (U10) with respect to forecast time. Pangu-Weather is compared to both operational IFS and FourCastNet (only for U10). As seen, all the three methods tend to underestimate extremes, *i.e.*, the RQE values are consistently smaller than 0. The absolute RQE values reported by AI-based methods generally grow (*i.e.*, heavier underestimation) with forecast time, while that of operational IFS remains mostly unchanged. We attribute the above observation to the cumulative forecast errors of AI-based methods – compared to FourCastNet, Pangu-Weather significantly alleviates such errors with hierarchical temporal aggregation (see Section 3.4). Compared to operational IFS, Pangu-Weather shows higher absolute RQE values (*i.e.*, heavier underestimation) for U500 and lower absolute RQE values (*i.e.*, lighter underestimation) for Q500. Regarding U10, Pangu-Weather is much better than operational IFS for up to 3 days (72 hours) and then becomes slightly worse due to cumulative forecast errors.

4.2.2 Tracking Tropical Cyclones

We study a special case of extreme weather forecast, namely, tracking tropical cyclones. Note that we follow the conventions to focus on forecasting the eye of tropical cyclones rather than the intensity[11]. Hence, in this part, we report the averaged distance between the ground-truth and predicted cyclone eyes.

To track the eye of a tropical cyclone, we follow [55] to find the local minimum of mean sea level pressure (MSLP).

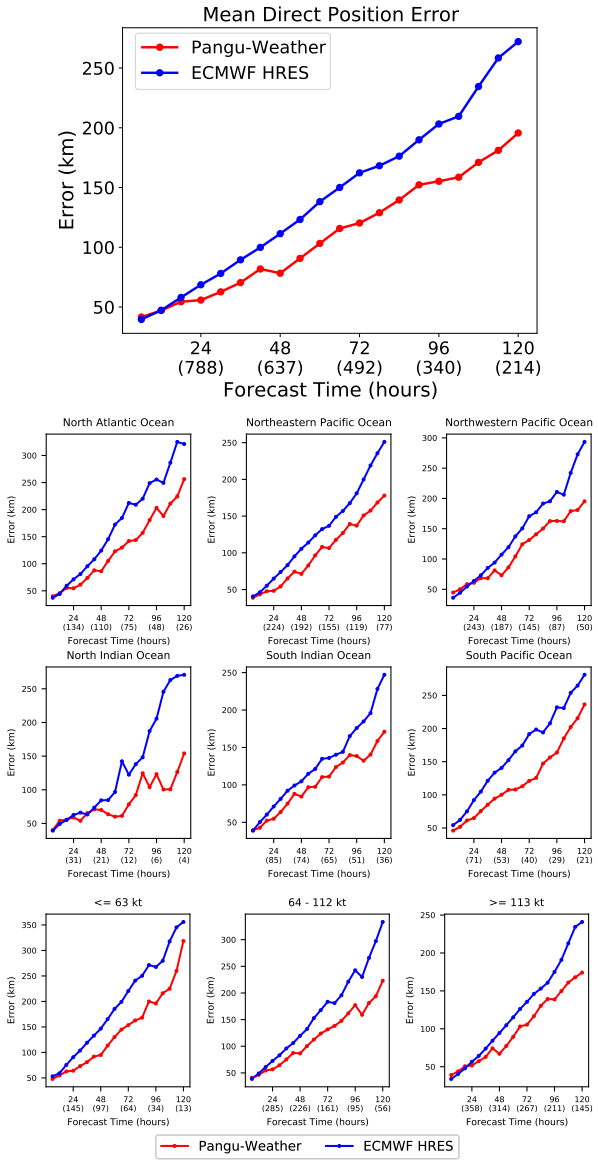

Fig. 10: The comparison of mean direct position errors of tropical cyclone tracking between Pangu-Weather and ECMWF-HRES, where the results are obtained by averaging 88 named tropical cyclones in 2018. We show the overall results (top), the results with respect to different basins, and the results with respect to different intensities (bottom).

11. Due to the limited resolution of EDA systems, reanalysis data like ERA5 always underestimate cyclones intensity (*e.g.*, minimum pressure and maximum wind speed) significantly [52], [53], [54]. Trained on ERA5, it is difficult for Pangu-Weather to forecast the intensity accurately (*e.g.*, the predicted minimum pressure is often 50hPa higher than the ground-truth), while the path tracking accuracy is reasonable (see the later results). In the future, if higher-resolution, unbiased weather data (especially tropical cyclone data) are provided, it is very likely that Pangu-Weather can be directly trained or fine-tuned on these data for more accurate intensity forecast.

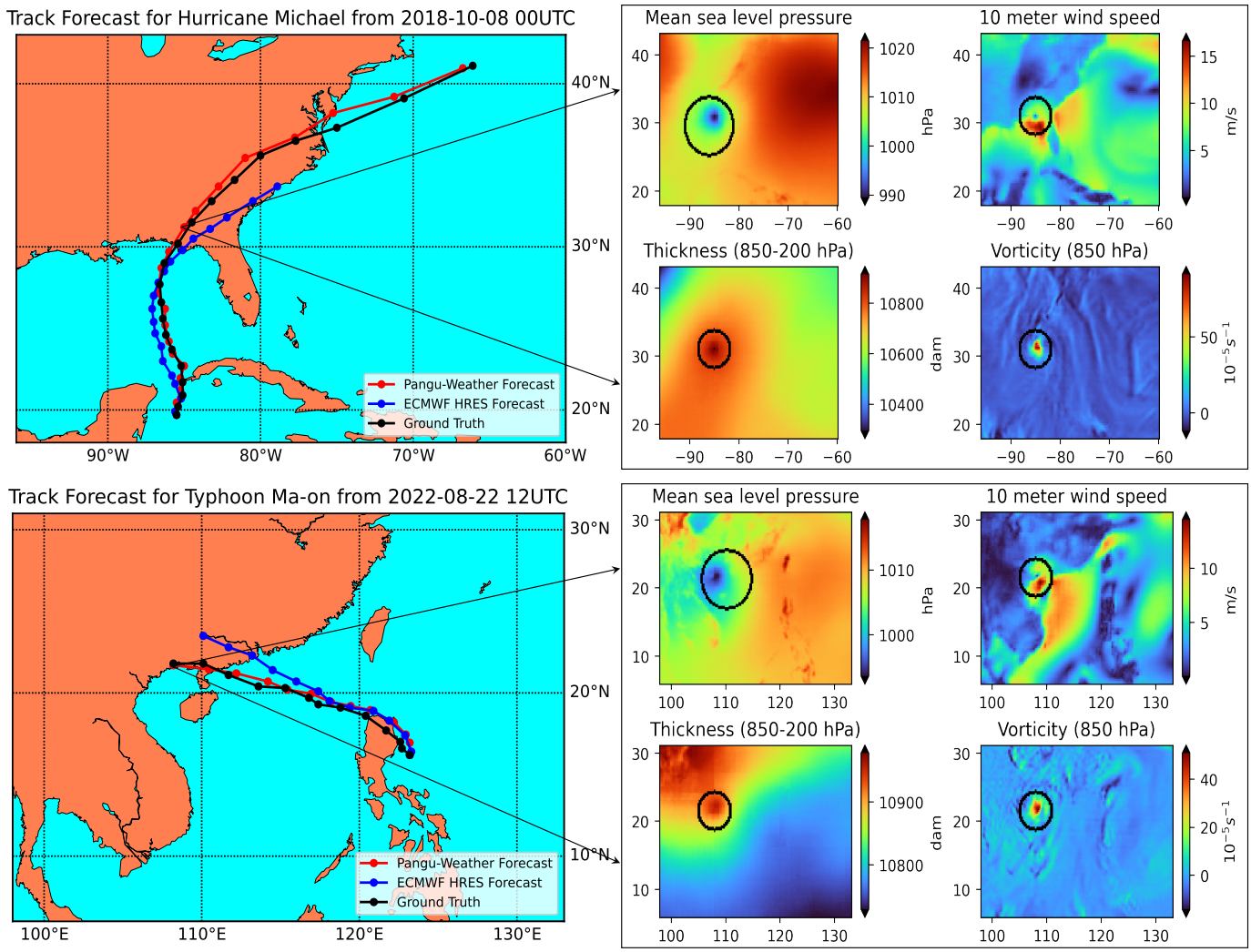

Fig. 11: **Left**: the tracking of cyclone eyes for Hurricane Michael (2018-13) and Typhoon Ma-on (2022-09) by Pangu-Weather and ECMWF-HRES, with a comparison to the ground-truth (by IBTrACS). **Right**: an illustration of the tracking process, where we use Pangu-Weather as an example. It locates cyclone eye by checking four variables (from forecast results), namely, mean sea level pressure, 10m wind speed, thickness between 850hPa and 200hPa, and vorticity of 850hPa). The displayed figures correspond to the forecast of these variables at a forecast time of 72 hours, and the forecast of cyclone eye is indicated using the tail of arrows.

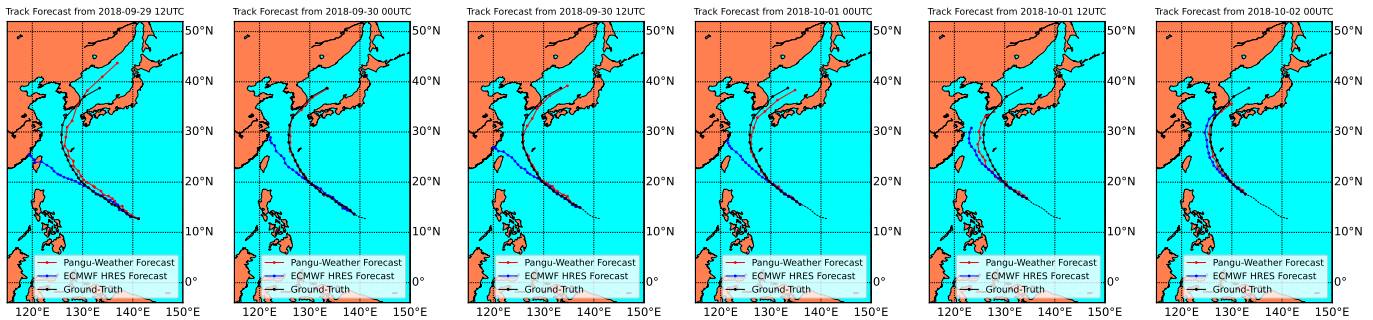

Fig. 12: The dynamic tracking results of cyclone eyes for Typhoon Kong-rey (2018-25) by Pangu-Weather and ECMWF-HRES, with a comparison to the ground-truth (by IBTrACS). We show six time points with the first one being 12:00 UTC, September 29th, 2018, and the time gap between neighboring points being 12 hours. The historical (observed) path of cyclone eyes is shown in dashed. Mind the significant difference between Pangu-Weather and ECMWF-HRES (Pangu-Weather is significantly better) at the middle four time points.

Specifically, we follow [54] to set the lead time to be 6 hours. Given the starting time point and the corresponding (initial) position of a cyclone eye, we iteratively call for forecasting the 6-hour-later weather states and look for a local minimum of MSLP that satisfies the following conditions:

- There is a maximum of $850\mathrm{hPa}$ relative vorticity that is larger than 5×10^{-5} within a radius of $278\mathrm{km}$ for the Northern Hemisphere, or a minimum that is smaller than 5×10^{-5} for the Southern Hemisphere.
- There is a maximum of thickness between $850\mathrm{hPa}$ and $200\mathrm{hPa}$ within a radius of $278\mathrm{km}$ when the cyclone is extratropical.
- The maximum 10m wind speed is larger than $8\mathrm{m/s}$ within a radius of $278\mathrm{km}$ when the cyclone is on land.

Once the cyclone eye is located, the tracking algorithm continues to find the next position in a vicinity of $445\mathrm{km}$. The tracking algorithm terminates when no local minimum of MSLP is found to satisfy the above conditions.

We refer to the International Best Track Archive for Climate Stewardship (IBTrACS) project [56], [57] which contains the best available estimations for tropical cyclones. We directly apply the above tracking algorithm on the deterministic forecast results of Pangu-Weather. We compare the tracking results to ECMWF-HRES, a strong competitor of cyclone tracking based on high-resolution ($9km \times 9km$) operational weather forecast – clearly, a higher-resolution forecast is more accurate in locating tropical cyclone eyes. The ECMWF-HRES forecast of cyclone eyes are directly downloaded from the TIGGE archive [16]. For a fair comparison, we choose the tropical cyclones in 2018 (the year of the above quantitative study of deterministic forecasts) that appeared in both the IBTrACS project and the ECMWF-HRES forecasts. This results in a dataset (which we call TC2018) with 88 named tropical cyclones.

We quantitatively compare the forecast accuracy of Pangu-Weather and ECMWF-HRES in TC2018. The 3-day and 5-day mean direct position errors (for cyclone eyes) of Pangu-Weather are $120.29\mathrm{km}$ and $195.65\mathrm{km}$, respectively, which are significantly smaller than $162.28\mathrm{km}$ and $272.10\mathrm{km}$ reported by ECMWF-HRES. Figure 10 plots the mean direct position errors with respect to forecast time. One can see a clear advantage of Pangu-Weather over ECMWF-HRES over the entire dataset and within subsets of different basins or different intensities. Inheriting the property of deterministic forecast, the advantage becomes more significant when forecast time gets larger.

The tracking results of some representative cases are shown in Figures 1 and 11. We study four representative cases, three in 2018 and one in 2022. For Michael and Ma-on, we set the starting time point to be the earliest one in the ECMWF-HRES forecast, while for Kong-rey and Yutu, the starting points are postponed for a few days for better visualization. We use Pangu-Weather to forecast the entire cyclone path (*i.e.*, until the cyclone dissipates), and compare the tracking results to ECMWF-HRES and the ground-truth. Again, Pangu-Weather produces much more accurate tracking results compared to ECMWF-HRES, and the advantage becomes large as forecast time increases. Below, we analyze these cases one-by-one.

- **Typhoon Kong-rey (2018-25)**[12] is one of the most powerful tropical cyclones worldwide in 2018. It caused 4 fatalities and $\$171.5$ million damage. As shown in Figure 1, ECMWF-HRES forecasts that Kong-rey would land on China, but it actually did not. Pangu-Weather, instead, produces accurate tracking results which almost coincide with the ground-truth. Also, Figure 12 shows the tracking results of Pangu-Weather and ECMWF-HRES at different time points – the forecast of Pangu-Weather barely changes with time, yet ECMWF-HRES arrives at the conclusion that Kong-rey would not land on China more than 48 hours later than Pangu-Weather.
- **Typhoon Yutu (2018-26)**[13] is an extremely powerful tropical cyclone that caused catastrophic destruction in the Mariana Islands and the Philippines. It also ties Kong-rey as the most powerful tropical cyclone worldwide in 2018, resulting in 30 fatalities and $\$854.1$ million damage. As shown in Figure 3.1, Pangu-Weather makes the correct forecast (Yutu goes to the Philippines) as early as 6 days before landing, while the forecast of ECMWF-HRES is dramatically incorrect (Yutu makes a big turn and heads to the northeast). ECMWF-HRES forecasts the correct direction more than 48 hours later than Pangu-Weather.
- **Hurricane Michael (2018-13)**[14] is the strongest hurricane of the 2018 Atlantic hurricane season. Michael became a Category-5 hurricane and landed on Florida on October 10th, 2018, resulting in 74 fatalities and $\$25.5$ billion damage. As shown in Figure 11, with a starting time that is more than 3 days earlier than landing, Both Pangu-Weather and ECMWF-HRES forecast the landfall on Florida, but the delay of predicted landing time is only 3 hours for Pangu-Weather but 18 hours for ECMWF-HRES. In addition, Pangu-Weather shows great advantages in tracking Michael after it landed, while the tracking of ECMWF-HRES is much shorter and shifts to the east obviously.
- **Typhoon Ma-on (2022-09)**[15] is a severe tropical storm that impacted the Philippines and China. Ma-on landed over Maconacon, Philippines on August 23rd and made a second landfall over Maoming, China on August 25th, resulting in 7 fatalities and $\$9.13$ million damage. As shown in Figure 11, when the starting time point is about 3 days earlier than landing, ECMWF-HRES produces a wrong forecast that Ma-on would land on Zhuhai, China, while the forecast of Pangu-Weather is about right.

The much better tracking results are directly owed to the higher deterministic forecast accuracy of Pangu-Weather. In the right part of Figure 11, we show how Pangu-Weather tracks Hurricane Michael and Typhoon Ma-on following the specified tracking algorithm. Among the four variables, mean sea level pressure and 10m wind speed are directly produced by deterministic forecast, and thickness and vor-

12. https://en.wikipedia.org/wiki/Typhoon_Kong-rey_(2018)
13. https://en.wikipedia.org/wiki/Typhoon_Yutu
14. https://en.wikipedia.org/wiki/Hurricane_Michael
15. https://en.wikipedia.org/wiki/Tropical_Storm_Ma-on_(2022)

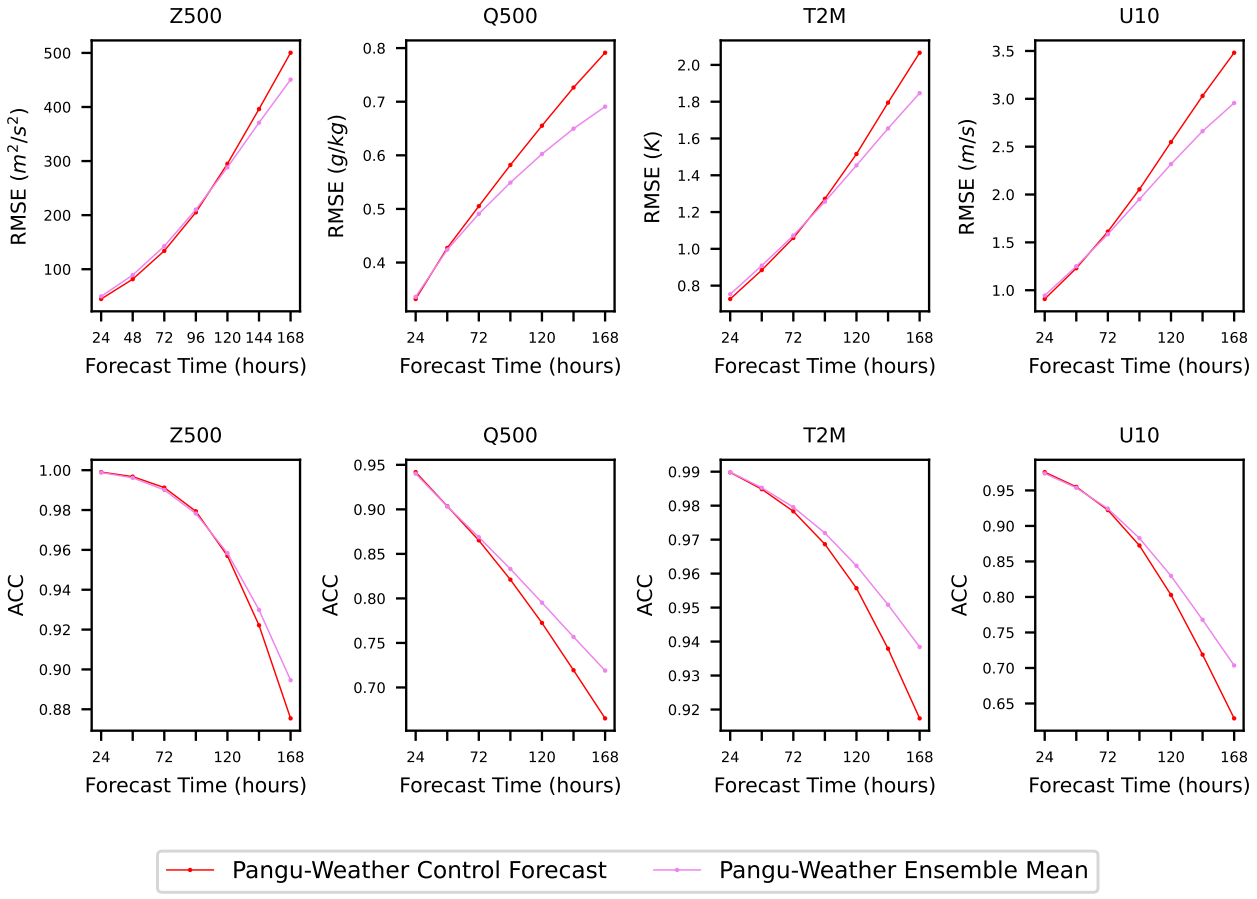

Fig. 13: Comparison of control forecast and ensemble forecast results of Pangu-Weather. We display the latitude-weighted RMSE (lower is better) and ACC (higher is better) for two upper-air variables (500hPa geopotential, Z500, and 500hPa specific humidity, Q500) and two surface variables (2m temperature, T2M, and u-component of 10m wind speed, U10).

ticity are mainly derived from geopotential and wind speed. This indicates that Pangu-Weather can produce intermediate results that support cyclone tracking, which further assists meteorologists in understanding and exploiting the tracking results.

In summary, the advantages of Pangu-Weather in tracking tropical cyclone eyes are mainly inherited from the good practice of deterministic forecast, in particular, the forecast of mean sea level pressure that is critical for locating cyclone eyes. Arguably, the proposed 3DEST architecture incorporates 3D information to improve the accuracy of important variables such as geopotential, wind speed, and mean sea level pressure. On the other hand, we notice that Pangu-Weather still heavily underestimates the intensity of tropical cyclones, arguably due to the same weakness of the ERA5 data. In the future, we expect higher-resolution, more accurate training data can be established for fine-tuning Pangu-Weather for these specific scenarios of extreme weather forecast. We also welcome meteorologists to offer expertise to improve the forecast of intensity.

4.3 Ensemble Forecast

A core goal of ensemble weather forecast is to investigate the uncertainty of forecast systems, *i.e.*, the change of forecast results with respect to small perturbations. Researchers mainly considered two types of uncertainty added to by (i) initial weather states and (ii) model parameters. The purpose is for diagnosing errors in observed or reanalyzed data (*e.g.*, caused by assimilation [35]) and bias of both NWP and AI-based models (*e.g.*, biasing towards false patterns).

The methodology of ensemble forecast mainly involves adding noise to either initial weather states or model parameters and observing the change of forecast results. Either of them requires performing the inference multiple times. Pangu-Weather, as an AI-based method, enjoys a much faster inference speed than conventional NWP methods, *e.g.*, more than $10{,}000\times$ faster than operational IFS. This offers an opportunity of performing large-member ensemble forecast in relatively low computational costs. In what follows, we offer a preliminary study of ensemble forecast based on Pangu-Weather, yet we believe that meteorologists can offer professional knowledge to further utilize the ability of ensemble forecast.

In this paper, we mainly investigate the first line that adds perturbations to initial weather states. For simplicity, we follow FourCastNet [14] to set the perturbations to be random Perlin noise, while we believe that richer meteorologic knowledge can assist us in developing more advanced ensemble methods (*e.g.*, based on singular vectors [58]). Mathematically, let the initial weather state be $\mathbf{A}_t^*$, and we

randomly generate $S = 99$ Perlin noise vectors of the same size of $\mathbf{A}_t^*$, denoted as $\mathbf{P}_1, \ldots, \mathbf{P}_S$. The initial states are perturbed into $\mathbf{A}_t^* + \eta\mathbf{P}_s$, where $\eta = 0.2$ is the coefficient that controls the noise amplitude, and $s = 0, 1, \ldots, S$ where $s = 0$ implies that no noise is added, *i.e.*, $\mathbf{P}_0 \equiv \mathbf{0}$. We feed all the $S + 1$ initial states to the trained model and average the outputs as the final ensemble forecast result. Experiments are still performed on the ERA5 weather data in 2018, where deterministic forecast is taken as a natural baseline. As shown in Figure 13, the accuracy of 100-member ensemble forecast is slightly worse than single-member deterministic forecast in short-range (*e.g.*, 1-day) weather forecast, but is significantly higher than deterministic forecast when forecast time is longer than 5 days. This aligns with our intuition and the observations of prior work [14], indicating that large-member ensemble forecast is especially useful when single-model accuracy becomes lower, yet it risks introducing unexpected noise that may cause accuracy drop when the deterministic forecast is accurate enough. In addition, ensemble forecast brings more benefits to the non-smooth variables such as 500hPa specific humidity (Q500) and 10m surface wind speed (U10), *e.g.*, the latitude-weighted RMSEs of 7-day forecast for Z500 and U10 are reduced from 500.3 and 3.48 to 450.6 and 2.96, with relative drops of 10% and 15%, respectively.

Temporarily, we do not study another line (*i.e.*, adding perturbations to model parameters) because Pangu-Weather is based on deep neural networks that contains hundreds of millions of parameters, unlike conventional NWP models [6], [7] that contain much fewer yet physically meaningful parameters. In addition, the learned parameters are highly sensitive to a few random factors during the training procedure (*e.g.*, random seeds, data sampling strategies, *etc.*) and thus are difficult to be perturbed for specific purposes. In the future, with the guidance from meteorologists, we expect to fine-tune the base Pangu-Weather models into a series of 'child models' for manipulating different factors.

5 Conclusions and Future Remarks

In this paper, we present Pangu-Weather, an AI-based system for numerical weather forecast. The technical contribution involves (i) designing the 3D Earth-specific transformer (3DEST) architecture and (ii) applying the hierarchical temporal aggregation strategy. By training deep neural networks on 39 years of global weather data, Pangu-Weather, for the first time, surpasses the conventional NWP methods in terms of both accuracy and speed. Being efficient in inference, Pangu-Weather opens a window for meteorologists to integrate their knowledge to AI-based methods for more exciting applications.

Looking into the future, we expect that computational resource is the key to further improving the accuracy of weather forecast. According to our experiments, the training procedure has not yet arrived at full convergence, and there is much room left in terms of (i) incorporating more observation factors, (ii) integrating the time dimension and training 4D deep networks, and (iii) simply using deeper and/or wider networks. All of these require more powerful GPUs with larger memory and higher FLOPs.

Acknowledgments

We would like to thank ECMWF for offering the ERA5 dataset and the TIGGE archive. Without such selfless dedication, this research would never become possible. We thank NOAA National Centers for Environmental Information for the IBTrACS dataset. We thank other members of the Pangu team for instructive discussions and support in computational resource. Our appreciation also goes to the Integration Verification team of Huawei Cloud EI that offers us a platform of high-performance parallel computing.

References

[1] P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction," *Nature*, vol. 525, no. 7567, pp. 47–55, 2015.

[2] W. C. Skamarock, J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, W. Wang, and J. G. Powers, "A description of the advanced research wrf version 2," National Center For Atmospheric Research Boulder Co Mesoscale and Microscale . . . , Tech. Rep., 2005.

[3] F. Molteni, R. Buizza, T. N. Palmer, and T. Petroliagis, "The ecmwf ensemble prediction system: Methodology and validation," *Quarterly journal of the royal meteorological society*, vol. 122, no. 529, pp. 73–119, 1996.

[4] H. Ritchie, C. Temperton, A. Simmons, M. Hortal, T. Davies, D. Dent, and M. Hamrud, "Implementation of the semi-lagrangian method in a high-resolution version of the ecmwf forecast model," *Monthly Weather Review*, vol. 123, no. 2, pp. 489–514, 1995.

[5] P. Bauer, T. Quintino, N. Wedi, A. Bonanni, M. Chrust, W. Deconinck, M. Diamantakis, P. Düben, S. English, J. Flemming *et al.*, *The ecmwf scalability programme: Progress and plans*. European Centre for Medium Range Weather Forecasts, 2020.

[6] T. Palmer, G. Shutts, R. Hagedorn, F. Doblas-Reyes, T. Jung, and M. Leutbecher, "Representing model uncertainty in weather and climate prediction," *Annual Review of Earth and Planetary Sciences*, vol. 33, no. 1, pp. 163–193, 2005.

[7] M. R. Allen, J. Kettleborough, and D. Stainforth, "Model error in weather and climate forecasting," in *ECMWF Predictability of Weather and Climate Seminar*. European Centre for Medium Range Weather Forecasts, Reading, UK, 2002, pp. 279–304.

[8] M. G. Schultz, C. Betancourt, B. Gong, F. Kleinert, M. Langguth, L. H. Leufen, A. Mozaffari, and S. Stadtler, "Can deep learning beat numerical weather prediction?" *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2194, p. 20200097, 2021.

[9] S. Scher and G. Messori, "Weather and climate forecasting with neural networks: using general circulation models (gcms) with different complexity as a study ground," *Geoscientific Model Development*, vol. 12, no. 7, pp. 2797–2809, 2019.

[10] S. Rasp, P. D. Dueben, S. Scher, J. A. Weyn, S. Mouatadid, and N. Thuerey, "Weatherbench: a benchmark data set for data-driven weather forecasting," *Journal of Advances in Modeling Earth Systems*, vol. 12, no. 11, p. e2020MS002203, 2020.

[11] J. A. Weyn, D. R. Durran, and R. Caruana, "Can machines learn to predict weather? using deep learning to predict gridded 500-hpa geopotential height from historical weather data," *Journal of Advances in Modeling Earth Systems*, vol. 11, no. 8, pp. 2680–2693, 2019.

[12] J. A. Weyn, D. R. Durran, R. Caruana, and N. Cresswell-Clay, "Sub-seasonal forecasting with a large ensemble of deep-learning weather prediction models," *Journal of Advances in Modeling Earth Systems*, vol. 13, no. 7, p. e2021MS002502, 2021.

[13] R. Keisler, "Forecasting global weather with graph neural networks," *arXiv preprint arXiv:2202.07575*, 2022.

[14] J. Pathak, S. Subramanian, P. Harrington, S. Raja, A. Chattopadhyay, M. Mardani, T. Kurth, D. Hall, Z. Li, K. Azizzadenesheli *et al.*, "Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators," *arXiv preprint arXiv:2202.11214*, 2022.

[15] Y. Hu, L. Chen, Z. Wang, and H. Li, "Swinvrnn: A data-driven ensemble forecasting model via learned distribution perturbation," *arXiv preprint arXiv:2205.13158*, 2022.

[16] P. Bougeault, Z. Toth, C. Bishop, B. Brown, D. Burridge, D. H. Chen, B. Ebert, M. Fuentes, T. M. Hamill, K. Mylne *et al.*, "The thorpex interactive grand global ensemble," *Bulletin of the American Meteorological Society*, vol. 91, no. 8, pp. 1059–1072, 2010.

[17] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[18] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[20] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[22] A. K. Betts, D. Z. Chan, and R. L. Desjardins, "Near-surface biases in era5 over the canadian prairies," *Frontiers in Environmental Science*, vol. 7, p. 129, 2019.

[23] Q. Jiang, W. Li, Z. Fan, X. He, W. Sun, S. Chen, J. Wen, J. Gao, and J. Wang, "Evaluation of the era5 reanalysis precipitation dataset over chinese mainland," *Journal of hydrology*, vol. 595, p. 125660, 2021.

[24] H. Hersbach, B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, J. Nicolas, C. Peubey, R. Radu, D. Schepers *et al.*, "The era5 global reanalysis," *Quarterly Journal of the Royal Meteorological Society*, vol. 146, no. 730, pp. 1999–2049, 2020.

[25] P. Lynch, "The origins of computer weather prediction and climate modeling," *Journal of computational physics*, vol. 227, no. 7, pp. 3431–3444, 2008.

[26] E. Kalnay, *Atmospheric modeling, data assimilation and predictability*. Cambridge university press, 2003.

[27] D. J. Stensrud, *Parameterization schemes: keys to understanding numerical weather prediction models*. Cambridge University Press, 2009.

[28] D. Randall, M. Khairoutdinov, A. Arakawa, and W. Grabowski, "Breaking the cloud parameterization deadlock," *Bulletin of the American Meteorological Society*, vol. 84, no. 11, pp. 1547–1564, 2003.

[29] R. Pincus, H. W. Barker, and J.-J. Morcrette, "A fast, flexible, approximate technique for computing radiative transfer in inhomogeneous cloud fields," *Journal of Geophysical Research: Atmospheres*, vol. 108, no. D13, 2003.

[30] A. Arakawa, "The cumulus parameterization problem: Past, present, and future," *Journal of climate*, vol. 17, no. 13, pp. 2493–2525, 2004.

[31] A. Betts, "Non-precipitating cumulus convection and its parameterization," *Quarterly Journal of the Royal Meteorological Society*, vol. 99, no. 419, pp. 178–196, 1973.

[32] H.-L. Kuo, "Further studies of the parameterization of the influence of cumulus convection on large-scale flow," *Journal of Atmospheric Sciences*, vol. 31, no. 5, pp. 1232–1240, 1974.

[33] T. Nakaegawa, "High-performance computing in meteorology under a context of an era of graphical processing units," *Computers*, vol. 11, no. 7, p. 114, 2022.

[34] H. Olafsson and J.-W. Bao, *Uncertainties in Numerical Weather Prediction*. Elsevier, 2020.

[35] I. M. Navon, "Data assimilation for numerical weather prediction: a review," *Data assimilation for atmospheric, oceanic and hydrologic applications*, pp. 21–65, 2009.

[36] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Computer Vision and Pattern Recognition*, 2016.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[40] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[41] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 012–10 022.

[42] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[43] X. Shi, Z. Gao, L. Lausen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo, "Deep learning for precipitation nowcasting: A benchmark and a new model," *Advances in neural information processing systems*, vol. 30, 2017.

[44] S. Agrawal, L. Barrington, C. Bromberg, J. Burge, C. Gazen, and J. Hickey, "Machine learning for precipitation nowcasting from radar images," *arXiv preprint arXiv:1912.12132*, 2019.

[45] S. Ravuri, K. Lenc, M. Willson, D. Kangin, R. Lam, P. Mirowski, M. Fitzsimons, M. Athanassiadou, S. Kashem, S. Madge *et al.*, "Skilful precipitation nowcasting using deep generative models of radar," *Nature*, vol. 597, no. 7878, pp. 672–677, 2021.

[46] V. Lebedev, V. Ivashkin, I. Rudenko, A. Ganshin, A. Molchanov, S. Ovcharenko, R. Grokhovetskiy, I. Bushmarinov, and D. Solomentsev, "Precipitation nowcasting with satellite imagery," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2680–2688.

[47] C. K. Sønderby, L. Espeholt, J. Heek, M. Dehghani, A. Oliver, T. Salimans, S. Agrawal, J. Hickey, and N. Kalchbrenner, "Metnet: A neural weather model for precipitation forecasting," *arXiv preprint arXiv:2003.12140*, 2020.

[48] H. Hersbach, B. Bell, P. Berrisford, G. Biavati, A. Horányi, J. Muñoz Sabater, J. Nicolas, C. Peubey, R. Radu, I. Rozum *et al.*, "Era5 hourly data on pressure levels from 1979 to present," *Copernicus climate change service (c3s) climate data store (cds)*, vol. 10, 2018.

[49] ——, "Era5 hourly data on single levels from 1979 to present," *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)*, vol. 10, 2018.

[50] D. Richardson, J. Bidlot, L. Ferranti, A. Ghelli, C. Gibert, T. Hewson, M. Janousek, F. Prates, and F. Vitart, *Verification statistics and evaluations of ECMWF forecasts in 2008-2009*. ECMWF Reading, UK, 2009.

[51] B. Fildier, W. D. Collins, and C. Muller, "Distortions of the rain distribution with warming, with and without self-aggregation," *Journal of Advances in Modeling Earth Systems*, vol. 13, no. 2, p. e2020MS002256, 2021.

[52] S. Bourdin, S. Fromang, W. Dulac, J. Cattiaux, and F. Chauvin, "Intercomparison of four tropical cyclones detection algorithms on era5," *EGUsphere*, pp. 1–43, 2022.

[53] P. Malakar, A. Kesarkar, J. Bhate, V. Singh, and A. Deshamukhya, "Comparison of reanalysis data sets to comprehend the evolution of tropical cyclones over north indian ocean," *Earth and Space Science*, vol. 7, no. 2, p. e2019EA000978, 2020.

[54] L. Magnusson, S. Majumdar, R. Emerton, D. Richardson, M. Alonso-Balmaseda, C. Baugh, P. Bechtold, J.-R. Bidlot, A. Bonanni, M. Bonavita, N. Bormann, A. Brown, P. Browne, H. Carr, M. Dahoui, G. D. Chiara, M. Diamantakis, D. Duncan, S. English, R. Forbes, A. J. Geer, T. Haiden, S. Healy, T. Hewson, B. Ingleby, M. Janousek, C. Kuehnlein, S. Lang, S.-J. Lock, T. McNally, K. Mogensen, F. Pappenberger, I. Polichtchouk, F. Prates, C. Prudhomme, F. Rabier, P. de Rosnay, T. Quintino, and M. Rennie, "Tropical cyclone activities at ecmwf," no. 888, 10 2021. [Online]. Available: https://www.ecmwf.int/node/20228

[55] P. White, "Newsletter no. 102 - winter 2004/05," 01 2005. [Online]. Available: https://www.ecmwf.int/node/14623

[56] K. R. Knapp, M. C. Kruk, D. H. Levinson, H. J. Diamond, and C. J. Neumann, "The international best track archive for climate stewardship (ibtracs) unifying tropical cyclone data," *Bulletin of the American Meteorological Society*, vol. 91, no. 3, pp. 363–376, 2010.

[57] K. R. Knapp, H. J. Diamond, J. P. Kossin, M. C. Kruk, C. Schreck *et al.*, "International best track archive for climate stewardship (ibtracs) project, version 4," *NOAA National Centers for Environmental Information*, 2018.

[58] E. P. Diaconescu and R. Laprise, "Singular vectors in atmospheric sciences: A review," *Earth-science reviews*, vol. 113, no. 3-4, pp. 161–175, 2012.