Supplementary Information

Residual corrective diffusion modeling for km-scale atmospheric downscaling

Morteza Mardani[1,*,], Noah Brenowitz[1,*], Yair Cohen[1,*], Jaideep Pathak[1], Chieh-Yu Chen[1], Cheng-Chin Liu[2], Arash Vahdat[1], Mohammad Amin Nabian[1], Tao Ge[1], Akshay Subramaniam[1], Karthik Kashinath[1], Jan Kautz[1], Mike Pritchard[1]

[1]NVIDIA, Santa Clara, CA. 95050, USA
[2]Central Weather Administration, 64, Gongyuan Road, Taipei 100006, Taiwan
[*]These authors contributed equally
[†]Corresponding author: mmardani@nvidia.com

Citation	Architecture	Resolutions	Pixels	Variables
Addison et al., (2022) [1]	Diffusion	input: 60km target: 8.8km	64^2	precipitation
Harris et al., (2022) [26]	GANs + ensemble forecast	input: 10km target: 1km	940^2	precipitation
Hatanaka et al., (2023) [27]	Cascaded diffusion	input: 30km target: 1km	128^2	day-ahead solar-irradiance
Leinonen et al., (2020) [39]	GANs	input 8km target: 1km	128^2	precipitation
Leinonen et al., (2020) [39]	GANs	input 16km target: 2km	128^2	cloud optical-thickness
Price and Rasp, (2022) [51]	Corrector GAN	input 32km target 4km	128^2	precipitation
Vosper et al., (2023) [72]	WGAN	input 100km target 10km	100^2	precipitation from tropical-cyclones
Current work	CorrDiff	input 25km target 2km	448^2	10-meter winds 2-meter temperature radar-reflectivity

Table S1: Summary table of some of the relevant works. We list here downscaling models presented in the most relevant works we could find with respect to the current study. We highlight the resolution ratios, the pixel size of the high resolution prediction, predicted variables and architecture.

Supplementary Information

1 Supplementary Note 1

To highlight the novel component of our work, we provide an expanded review of relevant literature. Table S1 presents a shortlist of the most relevant works that perform weather downscaling. Previous ML downscaling efforts have achieved notable successes in various areas. These include state vector inflation as [1], application to large spatial domains [26], achieving a large resolution ratio, e.g. [27, 1] and downscaling of precipitation in tropical cyclones [72]. It is worth noting, however, that the majority of these studies concentrate on downscaling a single variable per model (note that [39] provided two models, each for a single variable, and is thus listed twice in the table). The variables of interest in all these works primarily relate to cloud and precipitation properties. [72] showed a successful super resolution (recovering 10km from data coarsened to 100km) of tropical cyclone precipitation. Despite these advancements, to the best of our knowledge ML downscaling that accounts for different physics, across many channels and channel synthesis in a single model was not shown before.

The combined prediction of selected dynamical, thermodynamical and microphysical (cloud related) variables in concert marks a new capability of such models. Its utility is demonstrated here by examining coherent structures and how all variables jointly downscaled in a physically consistent manner.

A hybrid approach to the dynamical downscaling problem was recently presented by [41]. Their two-step approach downscaled from 100 km to 9 km resolution. First, a numerical model downscaled from 100 km to 45 km, followed by a diffusion model, similar to the one used here, that downscaled to 9 km. Notably, this work did not include channel synthesis. While the hybrid nature of this method increases computational cost, it may offer improved generalization capabilities.

2 Supplementary Methods

This section elaborates on the proposed CorrDiff methodology for probabilistic downscaling. It begins with a background on diffusion models to provide the machinery. It then delves into CorrDiff and its associated components. We further detail our experimental setup including the CWA dataset, network architecture, and training protocols. At the end, we briefly discuss evaluation criteria.

2.1 Background on diffusion models

Consider the data distribution represented by $p_{\text{data}}(\mathbf{x})$. This distribution has an associated standard deviation, denoted by σ_{data}. The forward diffusion process seeks to adjust this distribution, yielding modified distributions denoted by $p_{\text{data}}(\mathbf{x};\sigma)$. This transformation is achieved by incorporating i.i.d. Gaussian noise with a standard deviation of σ into the data. When σ surpasses σ_{data} by a considerable margin, the resulting distribution approximates pure Gaussian noise.

Conversely, the backward diffusion process operates by initially sampling noise, represented as $\mathbf{x}_0$, from the distribution $\mathcal{N}(0,\sigma_{\text{max}}^2\mathbf{I})$. The process then focuses on the denoising of this image into a series, $\mathbf{x}_i$, that is characterized by a descending order of noise levels: $\sigma_0 = \sigma_{\text{max}} > \sigma_1 > \ldots > \sigma_N = 0$. Each noise level corresponds to a specific distribution of the form $\mathbf{x}_i \sim p_{\text{data}}(\mathbf{x}_i;\sigma_i)$. The terminal image of the backward process, $\mathbf{x}_N$, is expected to approach the original data distribution.

formulation of the underlying stochastic differential equations. To present the forward and backward processes rigorously, they can be captured via stochastic differential equations (SDEs). Such SDEs ensure that the sample, $\mathbf{x}$, aligns with the designated data distribution, p, over its progression through time [66, 33]. A numerical SDE solver can be used here, where a critical component is the noise schedule, $\sigma(t)$, which prescribes the noise level at a specific diffusion-time, t. Here 'diffusion time' is a virtual variable used for denoting denoising steps and has roots in differential equations that are derived from Langevin dynamics. To avoid confusion with the time of day, we use diffusion time to denote this variable. A typical noise scheduler is $\sigma(t) \propto \sqrt{t}$. Based on [33], the forward SDE is given as

$$d\mathbf{x} = \sqrt{2\dot{\sigma}(t)\sigma(t)}d\boldsymbol{\omega}(t), \tag{3}$$

while the backward SDE is

$$d\mathbf{x} = -2\dot{\sigma}(t)\sigma(t)\nabla_{\mathbf{x}}\log p(\mathbf{x};\sigma(t))dt + \sqrt{2\dot{\sigma}(t)\sigma(t)}d\bar{\boldsymbol{\omega}}(t). \tag{4}$$

The term $\dot{\sigma}(t)$ refers to the derivative of $\sigma(t)$ with respect to the diffusion-time. Here ω in the forward SDE is a Wiener process, while the backward SDE comprises two terms: a deterministic component representing the probability flow ODE with noise degradation, and noise injection via the Wiener process.

Denoising score matching. An examination of the SDE in Eq. (4) indicates the necessity of the score function, $\nabla_{\mathbf{x}}\log p(\mathbf{x};\sigma)$, for sampling from diffusion models. Intriguingly, this score function remains unaffected by the normalization constant of the base distribution, regardless of its computational complexity. Given its independence, it can be deduced using a denoising method. If $\nabla_{\mathbf{x}}\log p(\mathbf{x};\sigma) = (D_\theta(\mathbf{x};\sigma)-\mathbf{x})/\sigma^2$, a denoising neural network, namely $D_\theta(\mathbf{x};\sigma)$, can be trained for the denoising task using

$$\min_\theta \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}}\mathbb{E}_{\sigma\sim p_\sigma}\mathbb{E}_{\mathbf{n}\sim\mathcal{N}(0,\sigma^2\mathbf{I})}\left[\|D_\theta(\mathbf{x}+\mathbf{n};\sigma)-\mathbf{x}\|^2\right]. \tag{5}$$

Note, the noise variance is also modeled as a random variable that simulates different noise levels in the forward process e.g., based on log-normal distribution; see [33].

2.2 Proposed approach

As discussed in Section 4, the Fig. state $\mathbf{x}$ in (1) can be written as the sum of the mean $\boldsymbol{\mu}$ and the difference $\mathbf{r}$, where the latter will be nearly zero mean and exhibits a small distribution shift, which facilities training diffusion models for correcting the mean prediction. It is worth noting that this two-step method has further implications for learning physics. The UNet-regression step can anticipate many of the physics of downscaling, some of which are deterministic. These include high-resolution topography (which to first order controls

the 2-meter temperature variation due to the lapse-rate effect), and the large-scale horizontal wind which combine leading balances in the free atmosphere with the effect of surface friction and topography. Stochastic phenomena such as convective storms that also change temperatures and winds are easier to model as deviations from the mean. Also, cloud resolving models are explicitly formulated using deviations from a larger scale balanced state [50]. In the next section, we discuss the regression and generation step in details.

2.2.1 Regression on the mean

In order to predict the conditional mean $\boldsymbol{\mu} = \mathbb{E}[\mathbf{x}|\mathbf{y}]$, we employ to a UNet-regression model trained on a dataset $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{N}$. We utilize the specific architecture described in [64] that incorporates attention and residual layers, allowing it to effectively capture both short and long-range dependencies in the data (see Section 3 and S1). The model is optimized using a Mean-Squared-Error (MSE) loss during training.

2.2.2 Corrective Denoising diffusion

Once equipped with the UNet-regression network, we can begin by predicting the conditional mean $\hat{\boldsymbol{\mu}}$, which serves as an approximation of $\mathbb{E}[\mathbf{x}|\mathbf{y}]$. Subsequently, we proceed to train the diffusion model directly on the difference: $\mathbf{r} = \mathbf{x} - \hat{\boldsymbol{\mu}}$. Notably, the difference exhibits a small departure from the target data, allowing for the utilization of smaller noise levels during the training of the diffusion process.

In our approach, we adopt the Elucidated diffusion model (EDM), a continuous-time diffusion model that adheres to the principles of SDEs (in Eq. (3)-(4)) [33] to design the diffusion process and architecture. As a result it has an intuitive and physics driven hyperparameter tuning, which makes it work across different domains. In our case, we want to generate $\mathbf{r}$ by sampling from the conditional distribution $p(\mathbf{r}|\mathbf{y})$ following the SDEs in Eq. (3)-(4). To condition the diffusion model, we concatenate the input coarse-resolution data $\mathbf{y}$ with the noise over different channels. We also learn the score function $\nabla_{\mathbf{r}} \log p(\mathbf{r}|\mathbf{y})$ using the score matching loss in Eq. (5) where the denoiser is now $D_\theta(\mathbf{r} + \mathbf{n}; \sigma; \mathbf{y})$ with the conditioning input $\mathbf{y}$. For the denoiser we again follow the design principles in EDM to use a UNet architecture with skip connections weighted by the noise variance. Architecture details are discussed in Section 2.3.2.

To generate samples from the distribution $p(\mathbf{r}|\mathbf{y})$, we employ the second-order EDM stochastic sampler [33] [Algorithm 2] to solve for the reverse SDE in Eq. (4). Upon sampling $\mathbf{r}$, we augment it with the predicted conditional mean $\hat{\boldsymbol{\mu}}$ from regression, to generate the sample $\hat{\boldsymbol{\mu}} + \mathbf{r}$. This entire workflow is illustrated in Fig. 1, providing a visual representation of the steps involved in our proposed method.

2.3 Experimental setup

2.3.1 Dataset

Our training and test data cover the region of Taiwan and surrounding ocean. The choice of region is driven by our partnership with a local government agency (CWA) which operates a regional dynamical downscaling. However, the region of Taiwan also presents a unique diversity of meteorological conditions and phenomena such as tropical cyclones (i.e., typhoons), mid-latitude cyclones which generate weather fronts, and steep topography with snow-capped mountains and land-sea contrast. Such diversity is hard to find elsewhere in such a relatively small domain.

The input (conditioning) dataset is taken from ERA5 reanalysis [28], a global dataset at spatial resolution of about 25km and a temporal resolution of 1h, the latter matches the target data listed below. To facilitate training, we interpolate the input data onto the curvilinear grid of CWA with bilinear interpolation (with a rate of 4x), which results in 36×36 pixels over the region of Taiwan. Each sample in the input dataset consists of 12 channels of information; see Table S2. This includes two pressure levels (500 hPa and 850 hPa) with four corresponding variables: temperature, eastward and northward components of the horizontal wind vector, and geopotential height. Additionally, the dataset includes single-level variables such as 2-meter Temperature, 10-meter wind vector and total column water vapor. This input channel set is admittedly somewhat arbitrary, but serves the purposes of (i) allowing a reasonable sparse representation of the thermodynamic state of the macro-scale atmosphere, while (ii) intentionally including only limited information about atmospheric water via the total vertical integral of an invisible trace gas - water vapor. This ensures the task of synthesizing

radar reflectivity - a much more complex observable that relates to the sixth moment of the raindrop size distribution - is appropriately ambitious, as a strong test of CorrDiff's generative component.

The target dataset used in this study is a subset of the proprietary RWRF model data (Radar Data Assimilation with WRFDA [2]). The RWRF model is one of the operational regional Numerical Weather Prediction (NWP) models developed by CWA [15], which focuses on radar Data Assimilation (DA) in the vicinity of Taiwan. Assimilating radar data is a common strategy used in regional weather prediction, which helps constrain especially stochastic convective processes such as mesoscale convective systems and short-lived thunderstorms. In addition, CWA assimilates several surface measurements to complement the radar data that often miss the surface values. The WRF-CWA system uses a nested 2km domain in a larger 10km domain that is driven by a global model (GFS) as boundary conditions [15]. By incorporating radar data [14], RWRF improves the short-term prediction of high-impact weather events. The radar observations possess high spatial resolution of approximately 1km and temporal resolutions of 5-10 minutes at a convective scale. These observations provide useful wind information (radial velocity) as well as hydrometeors (radar reflectivity), with a particular emphasis on the lower atmosphere. The radar data assimilation relies on the availability of reliable and precise observations, which contributes significantly to enhance the accuracy and performance of the applied deep learning algorithms in the context of NWP applications.

The target dataset covers the years 2018-2021. It has a temporal frequency of one hour and a spatial resolution of 2km. We use only the inner (nested) 2km domain, which has 448×448 pixels, projected using the Lambert conformal conical projection around Taiwan. The geographical extent of the dataset spans from approximately 116.371°E to 125.568°E in longitude and 19.5483°N to 27.8446°N in latitude. We sub-selected 4 channels (variables) as the target variables that are most relevant for practical forecasting: temperature at 2 meter above the surface, the horizontal winds at 10 meter above the surface and the maximum value of the derived radar reflectivity at each column (radar reflectivity hereafter) - a surrogate of the expected precipitation. Notably, the radar reflectivity channel is not present in the input data, and needs to be predicted based on the other channels (channel synthesis). The radar reflectivity data also exhibits a distinct distribution compared to the other output channels, with positively skewed values and a prominent zero-mode consistent with typical non-raining conditions.

Initially, the target data is provided in the NetCDF format, which is the output of the WRFDA assimilation process. A vertical interpolation from hybrid coordinates (i.e., sigma levels) to pressure coordinates (i.e., isobaric levels) is applied. As part of the preprocessing steps, the data is converted to the Hadoop Distributed File System (HDFS) format. Additionally, any missing or corrupted data points represented by "inf" or "nan" values are eliminated from the dataset. This leads to a reduction in the number of images from 37,944 to 33,813. See Section 3 for more details.

To avoid over-fitting, we divide the data into training and testing sets. Three years of data 2018-2020 are used for training (24,154 images total) and 2021 is used for testing. Some selected dates from 2022 and 2023 are used for case studies.

	Input	**Output**
Pixel side	36 x 36	448 x 448
Single level channels	Total column water vapor Temperature at 2 meter Eastward wind at 10 meter Northward wind at 10 meter	Maximum radar reflectivity Temperature at 2 meter Eastward wind at 10 meter Northward wind at 10 meter
Pressure level channels	Temperature Geopotential Eastward wind Northward wind	

Table S2: A list of the input and output resolutions and channel for the CorrDiff downscaling model. Input channels include the both single level variables and pressure level variables, the latter are used at 850 and 500 (hPa) levels.

[2] https://www.mmm.ucar.edu/models/wrfda

2.3.2 Network architecture and training

The CorrDiff method has two step learning approach. To ensure compatibility and consistency, we employ the same UNet architecture for both the regression and diffusion networks. We enhance the UNet of [64] by increasing its size to include 6 encoder layers and 6 decoder layers. The base embedding size is set to 128, and it is multiplied over channels according to the multipliers [1,2,2,2,2]. The attention resolution is defined as 28. To represent *time* (i.e., timesteps in the diffusion process, not to be confused with the time of day), we utilize the Fourier-based position embedding. However, in the regression network, time embedding is disabled as no probability flow ODE is involved. No data augmentation techniques are employed during training. Overall, the UNet architecture comprises 80 million parameters. Additionally, we introduce 4 channels for sinusoidal positional embedding to improve spatial consistency, following established practices in the field [71, 19, 13].

During the training phase, we use the Adam optimizer with a learning rate of 2×10^{-4}, $\beta_1 = 0.9$, and $\beta_2 = 0.99$. Exponential moving averages (EMA) with a rate of $\eta = 0.5$ are applied, and dropout with a rate of 0.13 is utilized. We adopt the hyperparameters based on the guidelines proposed in EDM, Karras et al., (2022), including the same optimizer hyperparameters and learning rate schedule. EDM offers a physics-inspired design space based on ODEs that can be auto-tuned to our scenario (see Table 1 in [33]). The only relevant hyperparameter that is tenable in this framework is P_{mean} of the noise schedule. The value of this parameter was selected based on the resolution and dynamic range of the data as done in [32].

The regression network receives only the 12 input channels from the ERA5 conditioning data. In contrast, the diffusion training concatenates these same 12 input conditioning channels from the coarse-resolution ERA5 data with 4 noise channels to generate the output for each denoiser. To further enhance the diffusion conditioning, we also add the mean obtained by the regression model in the first stage. This addition provides more context and global information to the diffusion process, potentially improving its convergence.. For diffusion training, we adopt the Elucidated Diffusion Model (EDM), a continuous-time diffusion model. During training, EDM randomly selects the noise variance such that $\ln(\sigma(t)) \sim \mathcal{N}(0, 1.2^2)$ and aims to denoise the samples per mini-batch. EDM is trained for 28 million steps, while the regression UNet is trained for only 2 million steps. The training process is distributed across 16 DGX nodes, each equipped with 8 H100 GPUs, utilizing data parallelism and a total batch size of 512. The total training time for regression and diffusion models was 7 days that amounts to approximately $21,504$ GPU-hours.

For the residual diffusion process, during training we adopt log-normal distribution for the noise standard-deviation σ; see (5) [33]. We choose $\sigma \sim \text{lognormal}(\mu = 0.0, \sigma = 1.2)$ to ensure the forward diffusion completely destructs the large data intensity especially for the radar reflectivity variable. For sampling purposes, we employ the second-order stochastic sampler provided by EDM. This sampler performs 18 steps, starting from a maximum noise variance of $\sigma_{\max} = 800$ and gradually decreasing it to a minimum noise variance of $\sigma_{\min} = 0.002$. We adopt the rest of hyperparamaters from EDM as listed in [33].

2.4 Evaluation criterion

Probabilistic predictions aim to maximize sharpness subject to calibration [53]. Qualitatively, calibration means that the likelihood of observing the true value is the same as observing a member drawn from the ensemble. A necessary condition for calibration is that the spread-error relationship be 1-to-1 when averaged over sufficient samples [21]. Calibration also manifests as a flat rank-histogram, both are reported in 2.4. A simple metric used below is the root-mean-squared error of the sample mean. In the large sample limit, the sample mean becomes deterministic. So we expect this error to be comparable for generative and deterministic models.

Instead of considering both calibration and spread separately, it can be easier to use proper scoring rules like the continuous-ranked-probability score (CRPS) [23]. Let x be a scalar observation and F be the cumulative distribution of the probabilistic forecast (e.g., the empirical CDF of generated samples). Then, CRPS is defined as

$$CRPS(F, x) = \int_{-\infty}^{\infty} (F(y) - \mathbb{1}_{\{y \geq x\}})^2 \, dy,$$

here $\mathbb{1}_{\{y \geq x\}}$ is the Heaviside step function, and F which minimizes CRPS is the true cumulative distribution of x. For a deterministic forecast, $F(y) = \mathbb{1}_{\{y \geq x_0\}}$ where x_0 is the forecast value, CRPS is equivalent to the mean absolute .

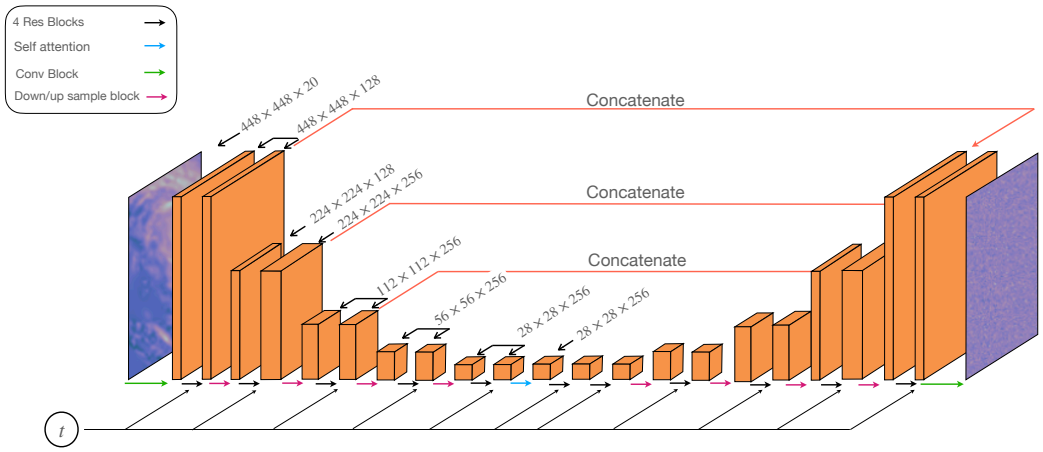


Figure S1: A sketch of the hierarchical UNet architecture adopted in both the regression model and the denoising diffusion model. Note, in the regression stage, time embedding is not used.

3 Supplementary Note 2

Figure S1 illustrates the architecture of the UNet, which serves a dual purpose in CorrDiff: it functions as both the regression model and the denoiser in the diffusion model. Table S2 provides a comprehensive list of the input and output channels utilized by CorrDiff. It is important to note that these input and output datasets differ not only in their pixel size but also in the specific channels they encompass. For single-level variables, the input includes total column water vapor but lacks the maximum radar reflectivity, which is present in the output, and vice versa. The input also includes pressure level variables at the 850 and 500 (hPa) levels, which combine to 12 input channels.

Figure S2 below shows a time series of the normalized target data, where the mean and the 2 and 4 standard deviations from the mean are plotted. A single corrupted data point is evident by the negative spike in 2m temperature. Some missing data periods are also evident by gaps, such as around May 2021; these were removed due to 'NaNs' or 'Inf' in either input or target data. The validation data constitutes the last 12 months from 2021-01-01 00:00:00 UTC. Moreover, the training data (2018-2020) has 22 hourly snapshots from 5 named typhoons of category 1 or higher in the domain. As a rough estimate, frontal systems seem to pass through the domain between 4-8 time per year (every 2-3 weeks in the winter) and typically last about 8h within the domain. Therefore, both typhoons and frontal systems are rather rare in the training data.

4 Supplementary Note 3

Figure S3 (left) demonstrates the role of the two steps associated with CorrDiff as a function of spatial scales. From the target data (blue), it is seen that the regression step learns larger spatial scales, leaving some of the small scales for the diffusion step. In addition, from Fig. S3 (right), it is observed that the residual is more localized and varies less overall, especially for the temperature field, which is strongly driven by changes in

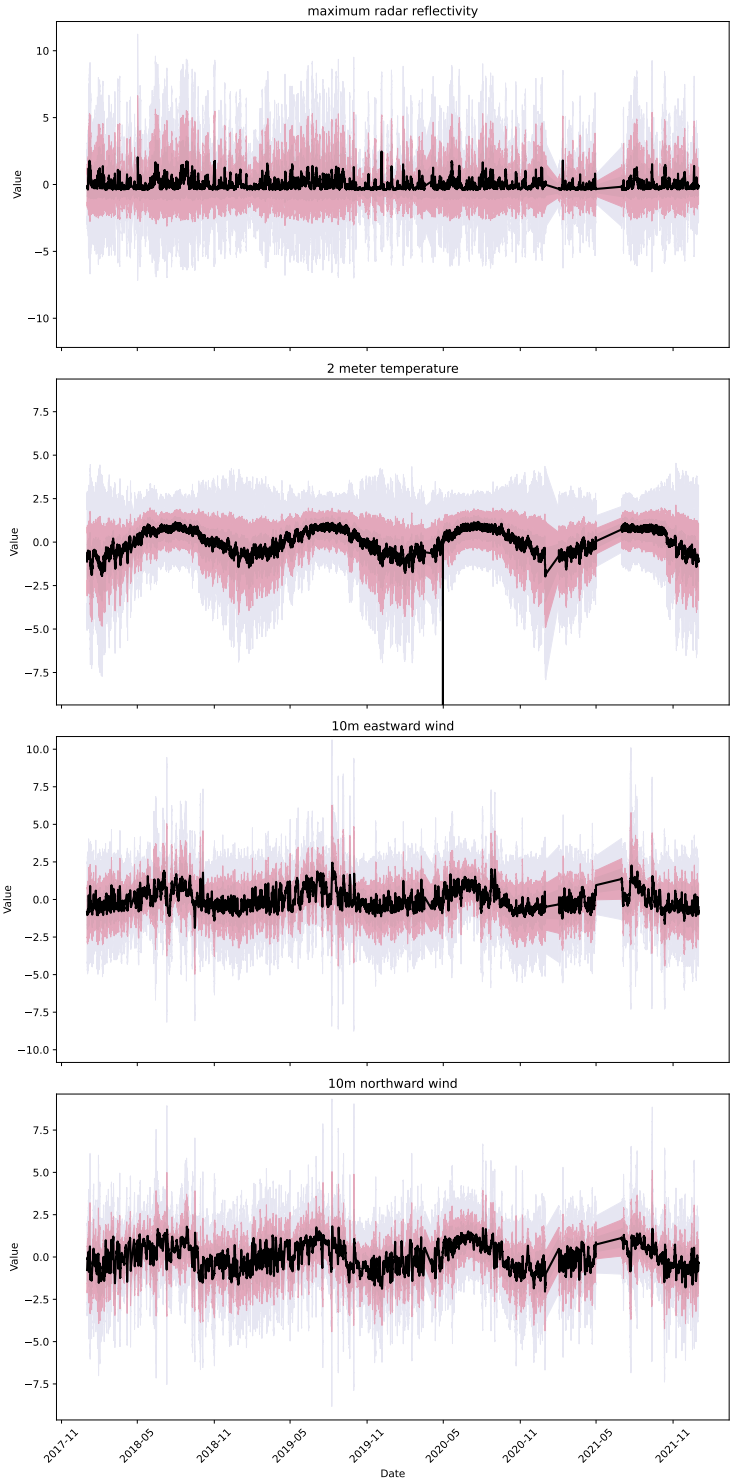


Figure S2: Statistics of the scaled training data. Time series of the mean (black line), mean ±2 standard deviation (red shading) and mean ±4 standard deviation (grey shading) for the four target channels as used by the model. This analysis is after re-scaling the channels by subtracting their global mean and dividing by global standard deviation. Statistics is computed over the domain at each date and time in the dataset.

	Radar	t2m	u10m	v10m
CorrDiff (CPRS)	1.64	0.5	0.73	0.82
CorrDiff (MAE)	2.0	0.59	0.88	0.98
UNet (MAE)	1.98	0.53	0.89	0.99
RF (MAE)	2.91	0.62	0.91	1.01
ERA5 (MAE)	-	0.83	0.97	1.05

Table S3: Pooled CRPS counterpart table for 1, where the prediction and target from each model are pooled on 28 grid boxes.

elevation. This has important implications for training and sampling efficiency of diffusion models as one can deploy diffusion models, with smaller UNet denoising architectures to aggregate the local information. We leave further study of this for future research.

5 Supplementary Note 4

In order to showcase the realism of individual samples from CorrDiff and their quality compared with the target data, Fig. S4 shows an animation of the target data (left), UNet (middle), and 20 generated samples of the CorrDiff prediction (right) of maximum radar reflectivity.

6 Supplementary Note 5

Due to the stochastic nature of km atmospheric fields, and specifically radar reflectivity, we complement the main text with a pooled counterpart of the metrics reported in Section 1. In S3 the MAE and CRPS are computed from data data pooled over 14 points, which is roughly the resolution of the ERA5 data used for conditioning.

Here as well, CorrDiff CRPS is superior to the MAE for all baselines, followed by the UNet, RF and ERA5 interpolation. This implies that the results presented in the main text are a valid reflection of the model's performance compared with the baselines.

7 Supplementary Note 6

To complement the example we have in the main text, we provide here an additional analysis of the case studies. For weather fronts, we analyze the collocation of the front with the reflectivity (clouds and rain), and for typhoons, we examine two different typhoons as well as historical typhoons for which there is no target data.

7.1 Additional weather front analysis

Building on the analysis of the front in the main text (see Fig. 5), we further examine the coherence of synthesized reflectivity in two frontal events: one from 2022 (mentioned above) and another from 2023. Figures S5 and S6 compare the reflectivity maps for the target and CorrDiff along with the cross-section of the along-front winds.

This analysis shows that although CorrDiff does not always sharpen the fronts to meet the target data, it syntheses the radar reflectivity consistently with the other variables and respects the frontal boundary by keeping the warm sector (of negative along front winds) cloud-free. We find that the frontal boundary is most visible in the along front wind. In all frontal events we examined, except the 2022 event in the main paper, fronts are moving fast and have large scale components of across front temperature gradient, and winds, which complicates tracking them in these channels.

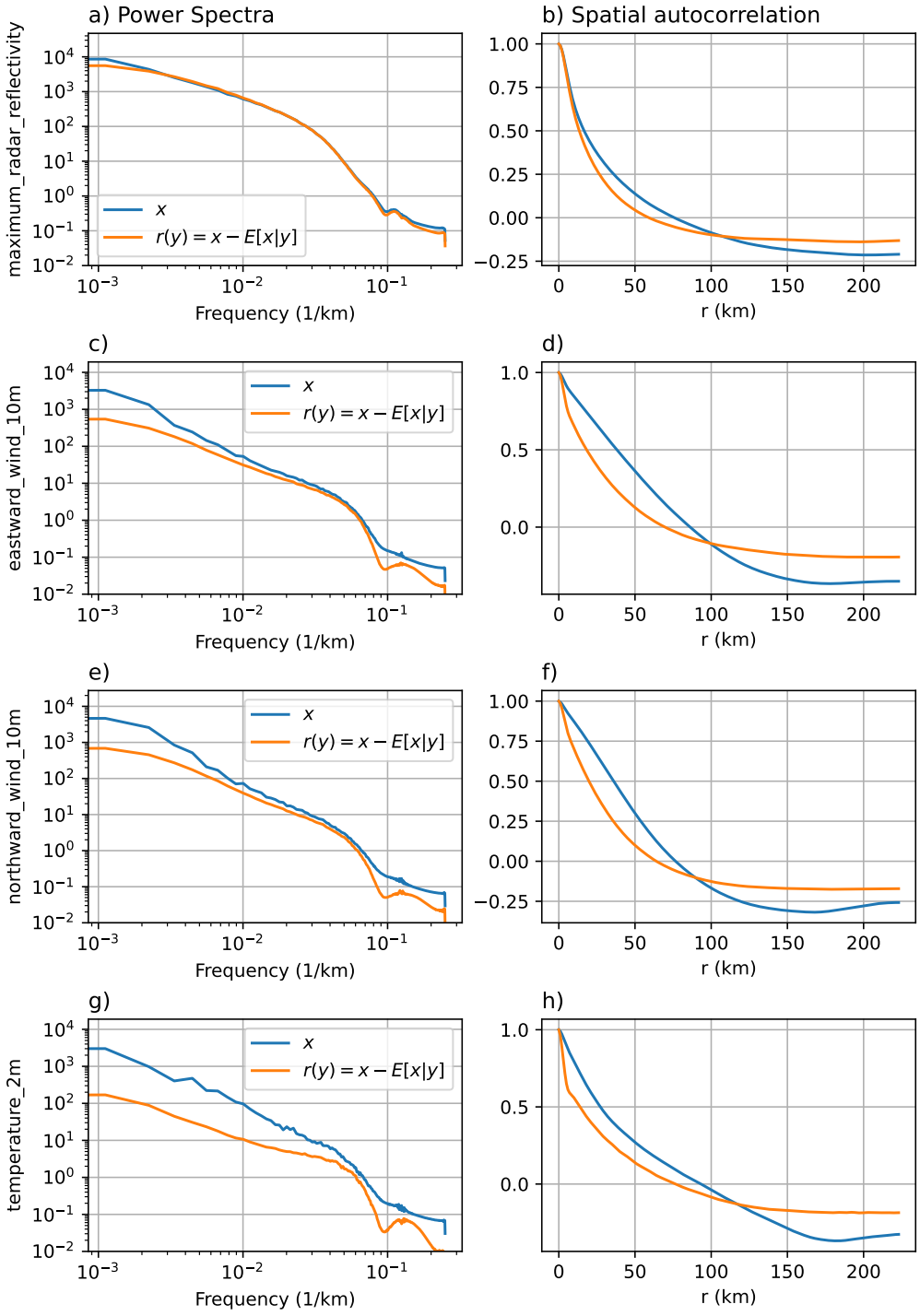


Figure S3: Examining the power spectra and spatial auto-correlation for the two steps in CorrDiff. Left column: power spectra, right column: spatial auto-correlation. Top to bottom: maximum radar reflectivity, 10m eastward wind, 10m northward wind and 2m temperature. This figure compares the original target x and the difference $r = x - \mathbb{E}[x|y]$. The difference has reduced variance at large-scales and equivalently removes the long-range auto-correlations.

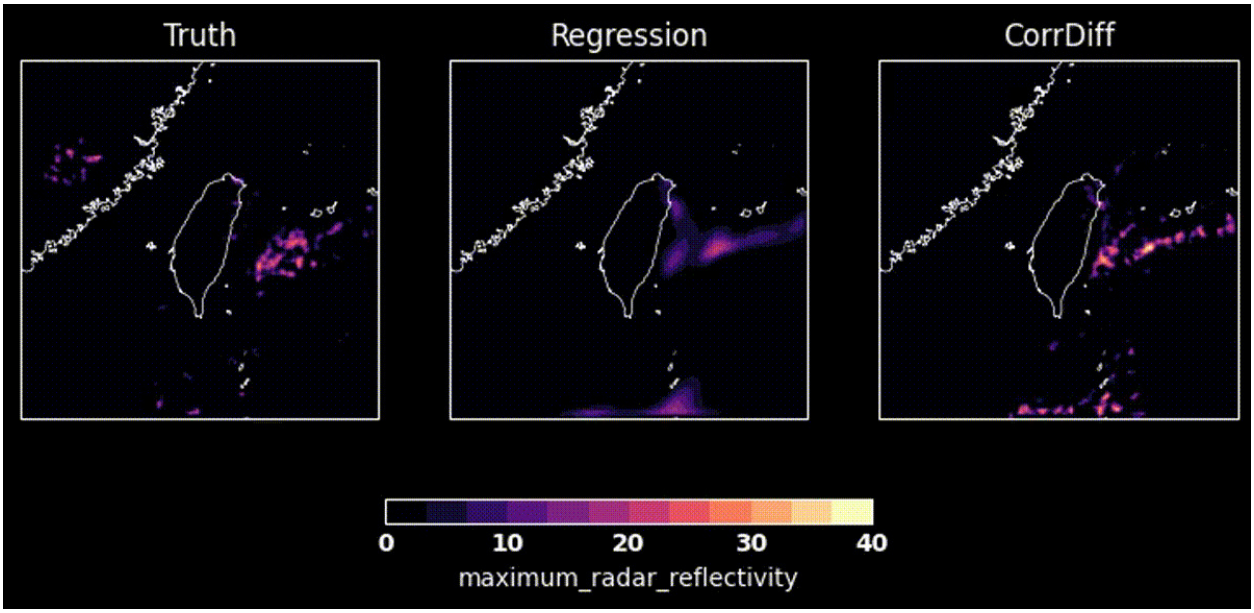


Figure S4: Comparative analysis of maximum radar reflectivity across multiple samples. Diffusion model predictions versus UNet regression and WRF target simulations for diverse cloud regimes. This figure is presented as an animated visualization, viewable as a video in Adobe-compatible formats. The full animation can be accessed and downloaded from `https://figshare.com/articles/media/CorrDiff_reflectivity/26798581?file=50490570`.

7.2 Additional typhoon analysis

As additional analysis we first analyze two out-of-sample typhoons and then analyze a large number of typhoons without CWA target data by comparing their diagnosed properties to observed records.

Figures S7 and S8 below show analyses of typhoons Chanthu (2021) and Haikui (2023) respectively. The former is in our out-of-sample year and the latter was received in an additional data from CWA specifically for this analysis. The performance of ERA5, and consequently CorrDiff, differs substantially between these two typhoons. Chanthu (2021) presents an exceptional challenge for low resolution models like the one underlying ERA5, due to its meridional trajectory along Taiwan's coast at a distance of about 100-200 km. In such a trajectory the impact of Taiwan's steep topography (up to 3km elevation) depends on the size of the simulated typhoon. The small Chanthu (2021) simulated by the WRF model, with a radius of maximum winds of about 25km, is effectively over ocean. Conversely, the larger Chanthu (2021) in ERA5, with radius of maximum winds of about 100km, is significantly affected by the topography.

As a result, the ERA5 Chanthu (2021) can have less than a third of the intensity of its WRF counterpart, see the first and third columns in S7. Furthermore, in ERA5 it often does not have a closed windspeed contour above $10ms^{-1}$, while in WRF Chanthu (2021) displays a coherent structure at $50ms^{-1}$ contours. Such a gap between the target (WRF) and the condition (ERA5) is hard for CorrDiff to overcome, and indeed the model fails to recover the intensity of the typhoon for most days. Only on 2023-09-03 12:00:00 UTC (top row of S7), when the typhoon has passed the island, does CorrDiff provide an improvement over ERA5.

Unlike Chanthu (2021), typhoon Haikui (2023) has a zonal trajectory leading to landfall in southern Taiwan. CorrDiff improves upon ERA5 by correcting about 50% of the intensity error and contracts the typhoon's radius of maximum winds.

Further analysis compares CorrDiff-simulated typhoons in the CWA region with historical records to evaluate typhoons for which no target data exists. The Japan Meteorological Agency best track data (JMA tracks) [4] includes the maximum windspeed (intensity) and radius of maximum windspeed (size) of typhoons in the West Pacific for several decades. We identified 648 instances of typhoons with intensities of $30ms^{-1}$ or greater within the CWA domain from 1980 to 2020. Panels (a) and (b) of Fig. S9 display the storm size and intensity, respectively, revealing the expected correction for ERA5 typhoons achieved through the

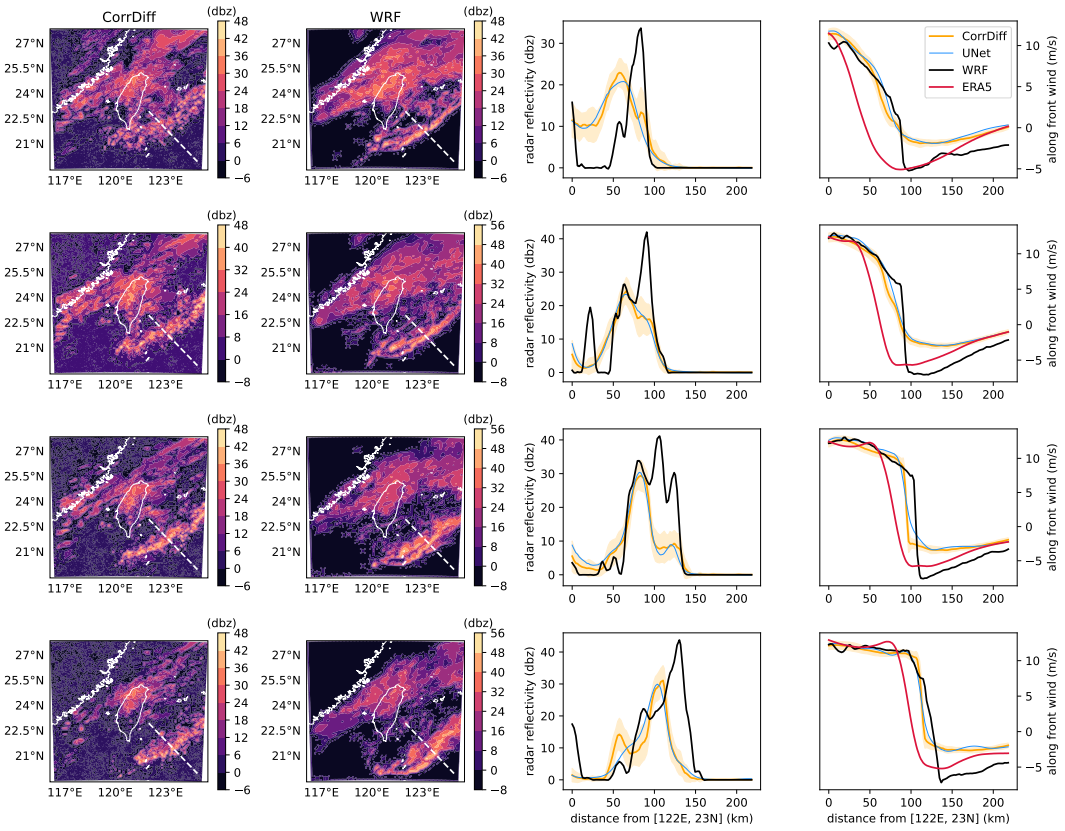


Figure S5: Analyzing the coherence of synthesized reflectivity with windspeed during a cold front. Left to right: radar reflectivity maps of CorrDiff and the target data, followed by the transects of reflectivity, and of the along front wind, along the dashed line in the adjacent maps. Top to bottom: 2022-02-13 17:000:00, 2022-02-13 19:00:00, 2022-02-13 21:00:00 and 2022-02-13 23:00:00 UTC.

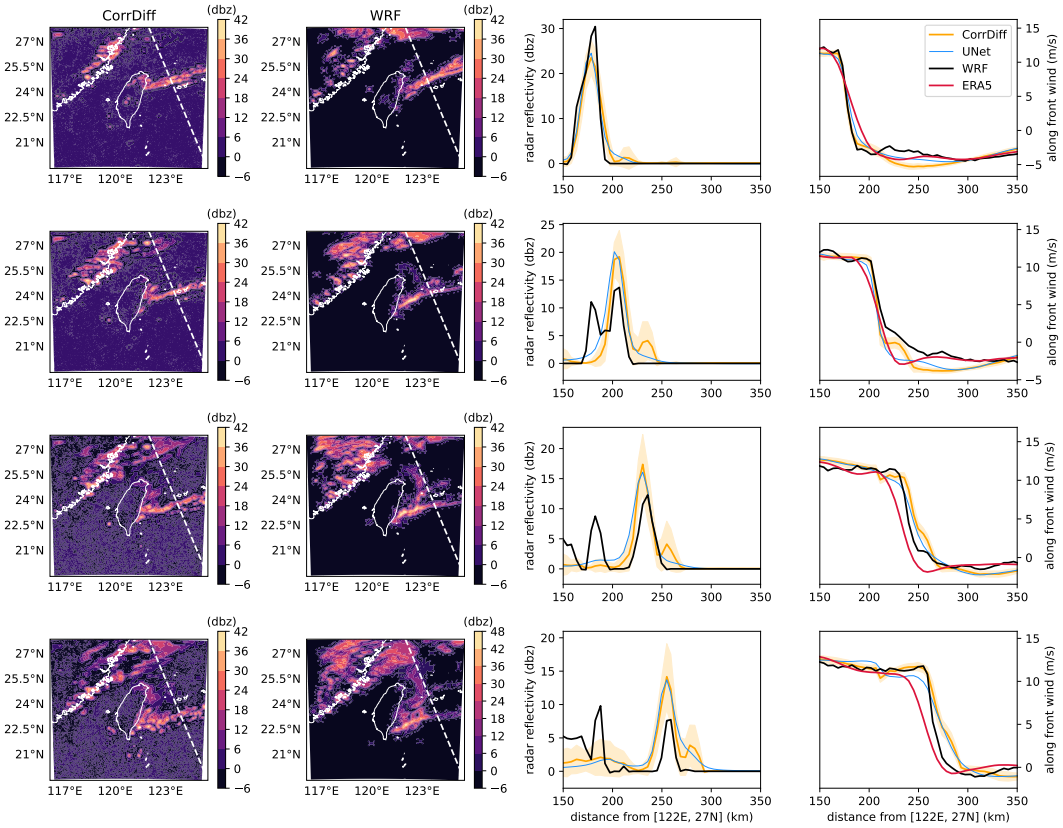


Figure S6: Same as S5 but for time: 2023-02-13 08:00:00, 2023-02-13 10:00:00, 2023-02-13 12:00:00 and 2023-02-13 14:00:00 UTC.

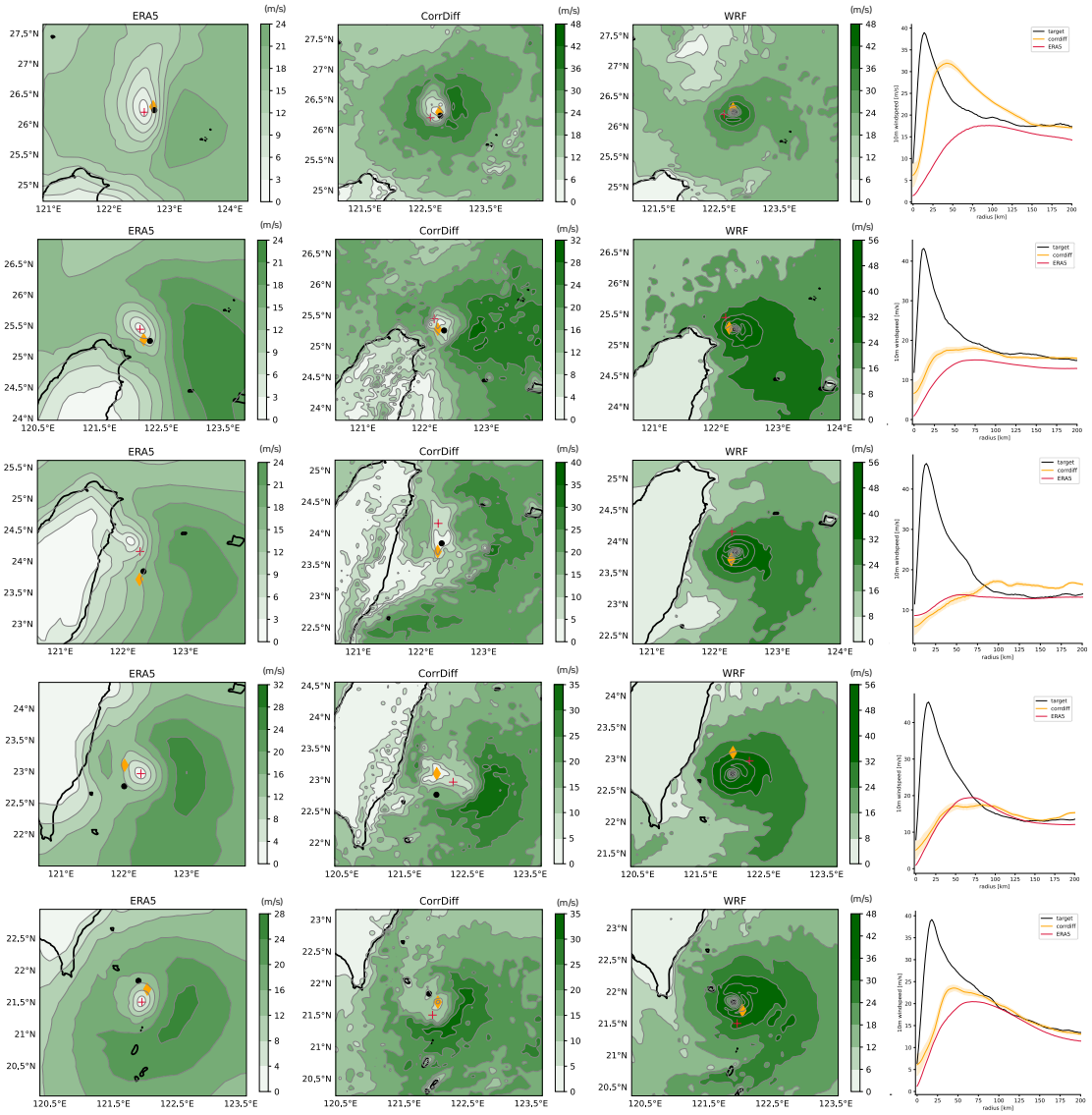


Figure S7: Windspeed maps and axisymmetric windspeed of typhoon Chanthu (2021). Left to right: windspeed in ERA5, CorrDiff, target (WRF model) and their axisymmetric profiles. The red '+', orange diamond and black dot show the storm center for the ERA5, CorrDiff, WRF respectively. Time is increasing from bottom to top: 2021-09-02 12:00:00 UTC, 2021-09-02 18:00:00 UTC, 2021-09-03 00:00:00 UTC, 2021-09-03 06:00:00 UTC, 2021-09-03 12:00:00.

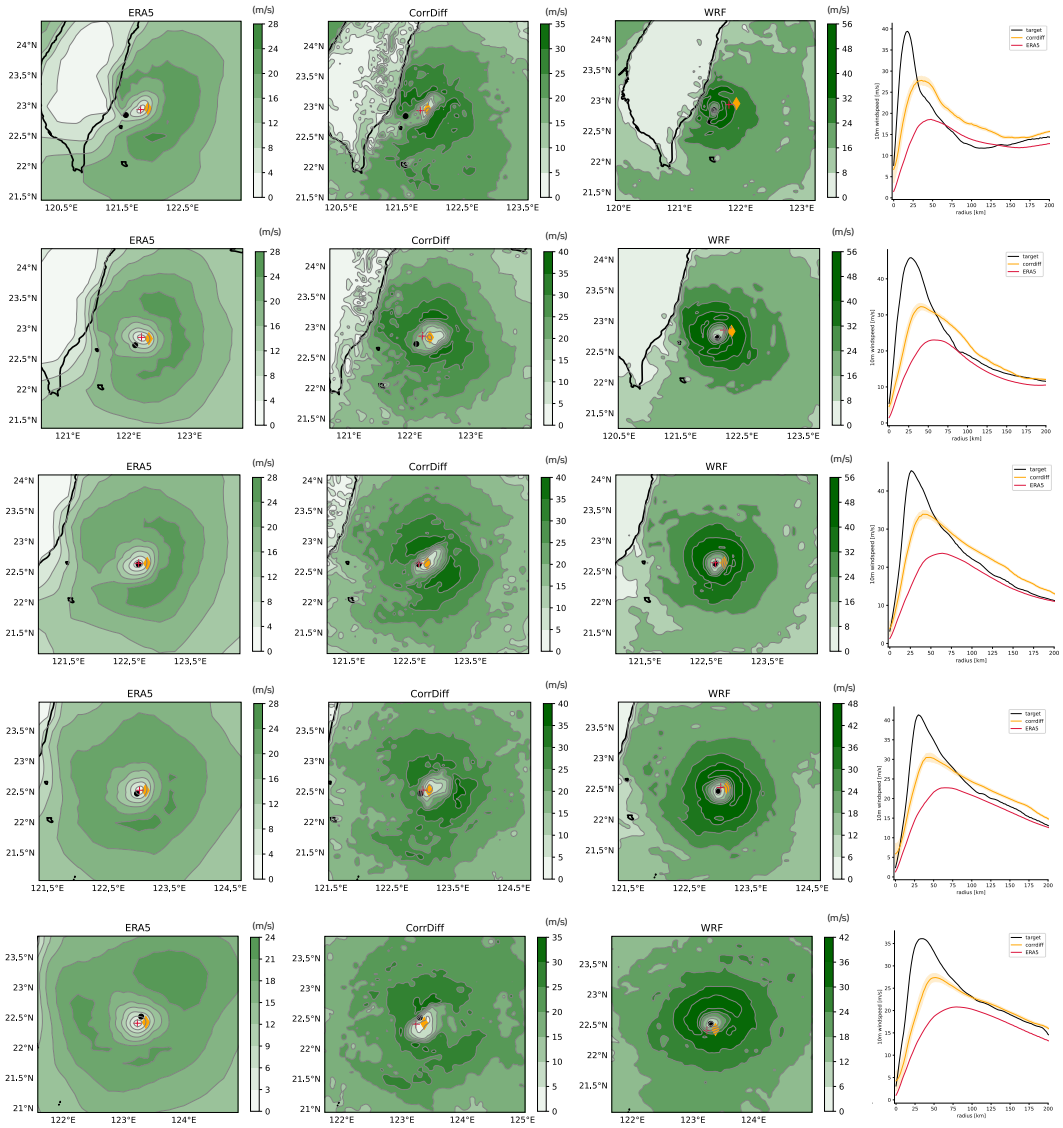


Figure S8: Same as S7 but for typhoon Haikui (2023). Time is increasing from bottom to top: 2023-09-02 18:00:00 UTC, 2023-09-02 21:00:00 UTC, 2023-09-03 00:00:00 UTC, 2023-09-03 03:00:00 UTC, 2023-09-03 06:00:00 UTC.

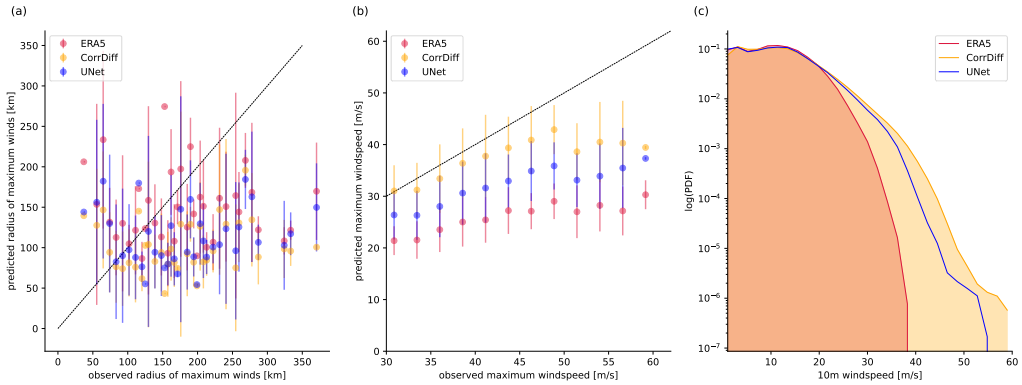


Figure S9: Comparison of predicted vs. observed (a) radius (km) and (b) value (ms^{-1}) of maximum axisymmetric windspeed (ms^{-1}) for all typhoons found in the domain with windspeed of $30ms^{-1}$ or more spanning the years 1980-2020. Predictions are compared via the mean (dot) and one standard deviation (bars) of all predictions falling within each observational bin of the reference JMA observations. Panel (c) shows the log(PDF) of the predicted windspeed flattened across all of these typhoon instances.

application of CorrDiff downscaling. One limitation of CorrDiff is that it reduces the size of all storms, including those with the correct size or those already too small in the ERA5 input data (panel a). The main benefit is improved windspeeds, removing most of the error between the ERA5 and the observed records for windspeeds up to $50ms^{-1}$ (panel b); though stronger storms have room for improvement. By correcting ERA5 toward JMA tracks, CorrDiff generates a five-fold increase in the probability of windspeed values exceeding hurricane-force winds (i.e., $33ms^{-1}$), see panel c. To the extent that JMA tracks can serve as ground truth, such distribution shift has significant societal implications, as these low-probability, high-impact events represent a substantial portion of the overall risk.

The UNet alone presents a less attractive alternative for downscaling typhoons, as its maximum axisymmetric windspeed is consistently positioned between the ERA5 and the CorrDiff values. CorrDiff offers a meaningful improvement over the UNet in both the maximum of the axisymmetric wind speed (panel b) and the wind speed maxima (panel c).

7.3 Supplementary Note 9

Comparing the performance of statistical downscaling like CorrDiff with dynamical downscaling like the WRF-CWA is challenging due to their different approaches and outcomes. Statistical downscaling produces a high-resolution state for a subset of channels at time t from a different (potentially larger) set of channels at lower resolution at the same time. Dynamical downscaling produces a full, high-resolution, state vector at time $t + \Delta t$ from both high and low resolution state vector at t using a numerical time-stepper. The time step (Δt) here is constrained by numerical stability and can be of the order of seconds. Thus, to produce the prediction at +1h, the dynamical downscaling model might run hundreds of autoregressive steps while the statistical downscaling will make a single inference.

Nonetheless, from a utility perspective, both approaches are used for producing a high resolution state at a given time. Thus, we can compare the latency and energy required to obtain a single high resolution prediction in Taiwan from the two approaches. We compare the speed of CorrDiff against the operational WRF run by CWA on their respective hardware.

The CWA-WRF is run on Fujitsu FX-100 system, with each node equipped with 32 SPARC64 Xifx CPU cores. A 13-hour deterministic CWA-WRF forecast (excluding data assimilation) is run on 928 CPU cores (across 29 nodes with a maximum system memory of 6.9GB per node) and takes about 20 minutes. CorrDiff inference is run on a single NVIDIA H100 Generation GPU, which takes 0.18 sec per downscaling sample. Given a global model 1-hour lead time forecast, the CorrDiff statistical downscaling on a single GPU is about 500 times faster than the dynamic downscaling that runs CWA-WRF on 928 CPUs, see S4.

	Hardware	Latency (sec/FH)	Power (J/sec)	Energy (kJ/FH)
WRF-CWA	928 CPUs	91.38	15.15	1285.46
CorrDiff	1 GPU	0.18	700	0.126

Table S4: A comparison of running the WRF model on the CWA system with CorrDiff inference on a single NVIDIA H100 GPU. Latency is given per Forecast Hour (FH) and Power is given in Joule/sec (W) per a single hardware unit (a CPU or a GPU), while Energy is for the entire forecast system (928 CPU for WRF-CWA) per FH.

Moreover, CorrDiff is about $10,000$ times more energy efficient. Since the individual samples are computed independently, conditioned on given global model data (which both systems need), CorrDiff can be produce a 13 hour forecast on 13 GPUs, thus obtaining about a 13x speedup over the above results (but with the same energy efficiency). These results ignore the additional compute needed for the regional DA in CWA-WRF, which is absent in CorrDiff (and likely impacts its performance). The regional DA in CWA, depending on the method used, can increase the compute for the CWA-WRF by a factor of 1.5 or 2 [15].

7.4 Supplementary Note 10

We further diagnose the statistical significance of the metrics presented in Table 1. We show this significance both graphically and using hypothesis testing.

By evaluating our models on the same set of times, we can take advantage of paired statistical tests, which provide much more power. Given the scores for two models x_i and y_i evaluated over times i, we find that $Var(x_i - y_i) \ll Var(x_i)$. So even though the scores may vary in time, the improvements are robust. Figure S10 shows the improvement in CRPS relative to the RF baseline. In all cases, the improvements of our models (CorrDiff and UNet) are larger than the error bars.

We further elaborate on the significance with formal hypothesis testing for the radar field. The standard non-parametric test for assessing whether $x_i > y_i$ is the Wilcoxon signed-rank test for $d_i = (x_i - y_i)$. An alternative test is the binomial test of $x_i > y_i$. In both cases, the p-values comparing the RF baseline with CorrDiff are smaller than 10^{-30}. This remarkably low p-value may seem surprising, but CorrDiff has lower CRPS in 205 out of 205 times. The likelihood this occurring by random chance is vanishingly small.

References

[1] Henry Addison, Elizabeth Kendon, Suman Ravuri, Laurence Aitchison, and Peter AG Watson. Machine learning emulation of a local-scale UK climate model. *arXiv preprint arXiv:2211.16116*, 2022.

[2] Rilwan A Adewoyin, Peter Dueben, Peter Watson, Yulan He, and Ritabrata Dutta. TRU-NET: a deep learning approach to high resolution prediction of rainfall. *Machine Learning*, 110:2035–2062, 2021.

[3] Jorge Baño-Medina, Rodrigo Manzanas, and José Manuel Gutiérrez. Configuration and intercomparison of deep learning neural models for statistical downscaling. *Geoscientific Model Development*, 13(4):2109–2124, 2020.

[4] Monika Barcikowska, Frauke Feser, and Hans Von Storch. Usability of best track data in climate statistics in the western north pacific. *Monthly Weather Review*, 140(9):2818–2830, 2012.

[5] G. Batzolis, J. Stanczuk, C.-B. Schönlieb, and C. Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021.

[6] Zied Ben-Bouallegue, Mariana C A Clare, Linus Magnusson, Estibaliz Gascon, Michael Maier-Gerber, Martin Janousek, Mark Rodwell, Florian Pinault, Jesper S Dramsch, Simon T K Lang, Baudouin Raoult, Florence Rabier, Matthieu Chevallier, Irina Sandu, Peter Dueben, Matthew Chantry, and Florian Pappenberger. The rise of data-driven weather forecasting, 2023.

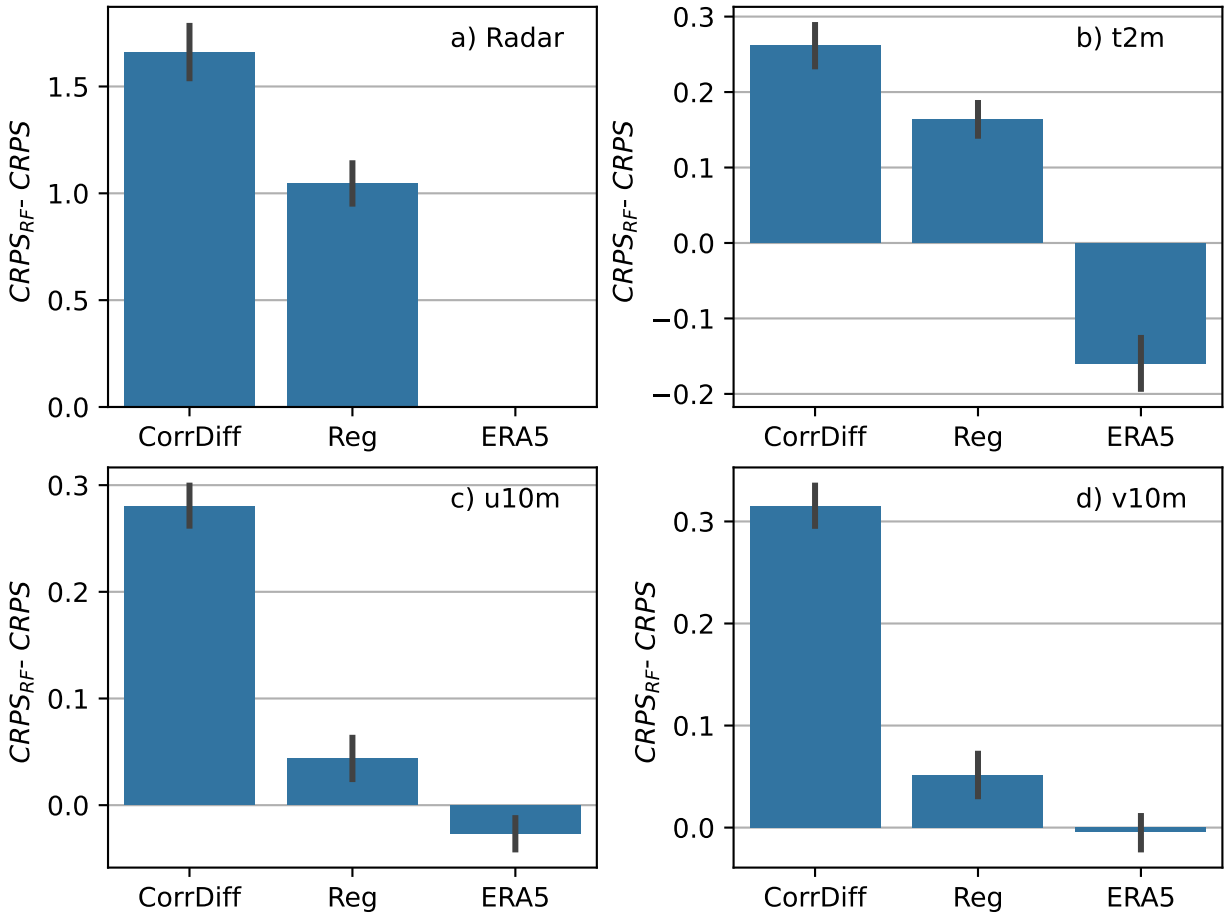


Figure S10: Boostrapping analysis of skill improvements presented in Table 1. The error bar shows a 95% confidence interval for the mean obtained by bootstrapping.

[7] Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, pages 1–6, 2023.

[8] Gu-Feng Bian, Gao-Zhen Nie, and Xin Qiu. How well is outer tropical cyclone size represented in the ERA5 reanalysis dataset? *Atmospheric Research*, 249:105339, 2021.

[9] Tobias Bischoff and Katherine Deck. Unpaired downscaling of fluid flows with diffusion bridges. *arXiv preprint arXiv:2305.01822*, 2023.

[10] Joseph K Blitzstein and Jessica Hwang. *Introduction to probability*. Crc Press, 2019.

[11] Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan Weyn, Haiyu Dong, Anna Vaughan, et al. Aurora: A foundation model of the atmosphere. *arXiv preprint arXiv:2405.13063*, 2024.

[12] Boris Bonev, Thorsten Kurth, Christian Hundt, Jaideep Pathak, Maximilian Baust, Karthik Kashinath, and Anima Anandkumar. Spherical fourier neural operators: Learning stable dynamics on the sphere. *arXiv preprint arXiv:2306.03838*, 2023.

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[14] Pao-Liang Chang, Jian Zhang, Yu-Shuang Tang, Lin Tang, Pin-Fang Lin, Carrie Langston, Brian Kaney, Chia-Rong Chen, and Kenneth Howard. An operational multi-radar multi-sensor qpe system in taiwan. *Bulletin of the American Meteorological Society*, 102(3):E555–E577, 2021.

[15] I-Han Chen, Jing-Shan Hong, Ya-Ting Tsai, and Chin-Tzu Fong. Improving afternoon thunderstorm prediction over taiwan through 3dvar-based radar and surface data assimilation. *Weather and Forecasting*, 35(6):2603–2620, 2020.

[16] Lei Chen, Xiaohui Zhong, Feng Zhang, Yuan Cheng, Yinghui Xu, Yuan Qi, and Hao Li. Fuxi: A cascade machine learning forecasting system for 15-day global weather forecast. *npj Climate and Atmospheric Science*, 6:190, 2023.

[17] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[18] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. In *Proceedings of NeurIPS*, volume 34, pages 8780–8794, 2021.

[19] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[20] Devajyoti Dutta, Ashish Routray, D Preveen Kumar, and John P George. Regional data assimilation with the ncmrwf unified model (ncum): impact of doppler weather radar radial wind. *Pure and Applied Geophysics*, 176:4575–4597, 2019.

[21] V Fortin, M Abaza, F Anctil, and R Turcotte. Why should ensemble spread match the RMSE of the ensemble mean? *J. Hydrometeorol.*, 15(4):1708–1713, August 2014.

[22] Andrew Geiss and Joseph C Hardin. Radar super resolution using a deep convolutional neural network. *Journal of Atmospheric and Oceanic Technology*, 37(12):2197–2207, 2020.

[23] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *J. Am. Stat. Assoc.*, 102(477):359–378, March 2007.

[24] Aofan Gong, Ruidong Li, Baoxiang Pan, Haonan Chen, Guangheng Ni, and Mingxuan Chen. Enhancing spatial variability representation of radar nowcasting with generative adversarial networks. *Remote Sensing*, 15(13):3306, 2023.

[25] William J Gutowski, Paul Aaron Ullrich, Alex Hall, L Ruby Leung, Travis Allen O'Brien, Christina M Patricola, RW Arritt, MS Bukovsky, Katherine V Calvin, Zhe Feng, et al. The ongoing need for high-resolution regional climate models: Process understanding and stakeholder information. *Bulletin of the American Meteorological Society*, 101(5):E664–E683, 2020.

[26] Lucy Harris, Andrew TT McRae, Matthew Chantry, Peter D Dueben, and Tim N Palmer. A generative deep learning approach to stochastic downscaling of precipitation forecasts. *Journal of Advances in Modeling Earth Systems*, 14(10):e2022MS003120, 2022.

[27] Yusuke Hatanaka, Yannik Glaser, Geoff Galgon, Giuseppe Torri, and Peter Sadowski. Diffusion models for high-resolution solar forecasts. *arXiv preprint arXiv:2302.00170*, 2023.

[28] Hans Hersbach, Bill Bell, Paul Berrisford, Shoji Hirahara, András Horányi, Joaquín Muñoz-Sabater, Julien Nicolas, Carole Peubey, Raluca Radu, Dinand Schepers, et al. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730):1999–2049, 2020.

[29] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Proceedings of NeurIPS*, volume 33, pages 6840–6851, 2020.

[30] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[31] Cathy Hohenegger, Peter Korn, Leonidas Linardakis, René Redler, Reiner Schnur, Panagiotis Adamidis, Jiawei Bao, Swantje Bastin, Milad Behravesh, Martin Bergemann, et al. Icon-sapphire: simulating the components of the earth system and their interactions at kilometer and subkilometer scales. *Geoscientific Model Development*, 16(2):779–811, 2023.

[32] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. simple diffusion: End-to-end diffusion for high resolution images. In *International Conference on Machine Learning*, pages 13213–13232. PMLR, 2023.

[33] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

[34] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. *arXiv [cs.CV]*, June 2024.

[35] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of GANs. *arXiv preprint arXiv:1705.07215*, 2017.

[36] Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global weather forecasting. *Science*, 0(0):eadi2336, 2023.

[37] Simon Lang, Mihai Alexe, Matthew Chantry, Jesper Dramsch, Florian Pinault, Baudouin Raoult, Mariana CA Clare, Christian Lessig, Michael Maier-Gerber, Linus Magnusson, et al. Aifs-ecmwf's data-driven forecasting system. *arXiv preprint arXiv:2406.01465*, 2024.

[38] Jussi Leinonen, Ulrich Hamann, Daniele Nerini, Urs Germann, and Gabriele Franch. Latent diffusion models for generative precipitation nowcasting with accurate uncertainty quantification. *arXiv preprint arXiv:2304.12891*, 2023.

[39] Jussi Leinonen, Daniele Nerini, and Alexis Berne. Stochastic super-resolution for downscaling time-evolving atmospheric fields with a generative adversarial network. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9):7211–7223, 2020.

[40] Lizao Li, Rob Carver, Ignacio Lopez-Gomez, Fei Sha, and John Anderson. Seeds: Emulation of weather forecast ensembles with diffusion models. *arXiv preprint arXiv:2306.14066*, 2023.

[41] Ignacio Lopez-Gomez, Zhong Yi Wan, Leonardo Zepeda-Núñez, Tapio Schneider, John Anderson, and Fei Sha. Dynamical-generative downscaling of climate model ensembles. *arXiv preprint arXiv:2410.01776*, 2024.

[42] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[43] Bin Mu, Bo Qin, Shijin Yuan, and Xiaoyun Qin. A climate downscaling deep learning model considering the multiscale spatial correlations and chaos of meteorological events. *Mathematical Problems in Engineering*, 2020:1–17, 2020.

[44] Pritthijit Nath, Pancham Shukla, and César Quilodrán-Casas. Forecasting tropical cyclones with cascaded diffusion models. *arXiv preprint arXiv:2310.01690*, 2023.

[45] National Centers for Environmental Prediction, National Weather Service, NOAA, U.S. Department of Commerce. Ncep gfs 0.25 degree global forecast grids historical archive, 2015.

[46] Nidhi Nishant, Sanaa Hobeichi, Steven C Sherwood, Gab Abramowitz, Yawen Shao, Craig Bishop, and Andy J Pitman. Comparison of a novel machine learning approach with dynamical downscaling for australian precipitation. *Environmental Research Letters*, 2023.

[47] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, et al. FourCastNet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

[48] Stephen B Pope. *Turbulent flows*. Cambridge university press, 2000.

[49] Jordan G Powers, Joseph B Klemp, William C Skamarock, Christopher A Davis, Jimy Dudhia, David O Gill, Janice L Coen, David J Gochis, Ravan Ahmadov, Steven E Peckham, et al. The weather research and forecasting model: Overview, system efforts, and future directions. *Bulletin of the American Meteorological Society*, 98(8):1717–1737, 2017.

[50] Kyle G Pressel, Colleen M Kaul, Tapio Schneider, Zhihong Tan, and Siddhartha Mishra. Large-eddy simulation in an anelastic framework with closed water and entropy balances. *Journal of Advances in Modeling Earth Systems*, 7(3):1425–1456, 2015.

[51] Ilan Price and Stephan Rasp. Increasing the accuracy and resolution of precipitation forecasts using deep generative models. In *International conference on artificial intelligence and statistics*, pages 10555–10571. PMLR, 2022.

[52] Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Timo Ewalds, Andrew El-Kadi, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, Remi Lam, and Matthew Willson. GenCast: Diffusion-based ensemble forecasting for medium-range weather. *arXiv preprint arXiv:2312.15796*, 2023.

[53] Adrian E Raftery, Tilmann Gneiting, Fadoua Balabdaoui, and Michael Polakowski. Using bayesian model averaging to calibrate forecast ensembles. *Mon. Weather Rev.*, 133(5):1155–1174, May 2005.

[54] Neelesh Rampal, Peter B Gibson, Abha Sood, Stephen Stuart, Nicolas C Fauchereau, Chris Brandolino, Ben Noll, and Tristan Meyers. High-resolution downscaling with interpretable deep learning: Rainfall extremes over new zealand. *Weather and Climate Extremes*, 38:100525, 2022.

[55] Neelesh Rampal, Sanaa Hobeichi, Peter B Gibson, Jorge Baño-Medina, Gab Abramowitz, Tom Beucler, Jose González-Abad, William Chapman, Paula Harder, and José Manuel Gutiérrez. Enhancing regional climate downscaling through advances in machine learning. *Artificial Intelligence for the Earth Systems*, 3(2):230066, 2024.

[56] Suman Ravuri, Karel Lenc, Matthew Willson, Dmitry Kangin, Remi Lam, Piotr Mirowski, Megan Fitzsimons, Maria Athanassiadou, Sheleem Kashem, Sam Madge, et al. Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878):672–677, 2021.

[57] Eduardo R. Rodrigues, Igor Oliveira, Renato L. F. Cunha, and Marco A. S. Netto. Deepdownscale: a deep learning strategy for high-resolution weather forecast. *arXiv preprint arXiv:1808.05264*, 2018.

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[59] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[60] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2021.

[61] Badr-eddine Sebbar, Saïd Khabba, Olivier Merlin, Vincent Simonneaux, Chouaib El Hachimi, Mohamed Hakim Kharrou, and Abdelghani Chehbouni. Machine-learning-based downscaling of hourly ERA5-land air temperature over mountainous regions. *Atmosphere*, 14(4):610, 2023.

[62] Tobias Selz and George C Craig. Upscale error growth in a high-resolution simulation of a summertime weather event over europe. *Monthly Weather Review*, 143(3):813–827, 2015.

[63] Jiefeng Song, Chaoyue Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[64] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[65] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[66] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Ashish Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[67] Bjorn Stevens, Masaki Satoh, Ludovic Auger, Joachim Biercamp, Christopher S Bretherton, Xi Chen, Peter Düben, Falko Judt, Marat Khairoutdinov, Daniel Klocke, et al. Dyamond: the dynamics of the atmospheric general circulation modeled on non-hydrostatic domains. *Progress in Earth and Planetary Science*, 6(1):1–17, 2019.

[68] Jason Stock, Jaideep Pathak, Yair Cohen, Mike Pritchard, Piyush Garg, Dale Durran, Morteza Mardani, and Noah Brenowitz. Diffobs: Generative diffusion for global forecasting of satellite observations. *arXiv preprint arXiv:2404.06517*, 2024.

[69] B Teufel, F Carmo, L Sushama, L Sun, MN Khaliq, S Bélair, A Shamseldin, D Nagesh Kumar, and J Vaze. Physics-informed deep learning framework to model intense precipitation events at super resolution. *Geoscience Letters*, 10(1):19, 2023.

[70] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

[71] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[72] Emily Vosper, Peter Watson, Lucy Harris, Andrew McRae, Raul Santos-Rodriguez, Laurence Aitchison, and Dann Mitchell. Deep learning for downscaling tropical cyclone rainfall to hazard-relevant spatial scales. *Journal of Geophysical Research: Atmospheres*, page e2022JD038163, 2023.

[73] Robert L Wilby, TML Wigley, D Conway, PD Jones, BC Hewitson, J Main, and DS Wilks. Statistical downscaling of general circulation model output: A comparison of methods. *Water resources research*, 34(11):2995–3008, 1998.

[74] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. In *Internatiodnal Conference on Learning Representations (ICLR)*, 2022.

[75] Hongkai Zheng, Weili Nie, Arash Vahdat, Kamyar Azizzadenesheli, and Anima Anandkumar. Fast sampling of diffusion models via operator learning. In *International Conference on Machine Learning*, pages 42390–42402. PMLR, 2023.