MoCast: Learning Turbulent Motions Under Physical Guidance for Precipitation Nowcasting

Binqing Wu [1,2], Weiqi Chen [3], Shiyu Liu [4], Zongjiang Shang [1,2], Haiou Wang [4], Liang Sun [3,*], Ling Chen [1,2,*]

[1] State Key Laboratory of Blockchain and Data Security, Zhejiang University
[2] College of Computer Science and Technology, Zhejiang University
[3] Damo Academy, Alibaba Group
[4] State Key Laboratory of Clean Energy Utilization, Zhejiang University
{binqingwu, zongjiangshang, lingchen}@cs.zju.edu.cn
{jarvus.cwq, liang.sun}@alibaba-inc.com, {shiyuliu, wanghaiou}@zju.edu.cn

Abstract

Precipitation nowcasting, a critical task for weather-sensitive applications, is highly challenging owing to the chaotic nature of atmospheric dynamics. Despite recent progress in deep learning, existing methods are limited in their capacity to model turbulent motions, one of the key drivers of precipitation evolution. Thus, we propose MoCast, the first work that incorporates turbulence knowledge to decompose turbulent motions into solvable components for precipitation nowcasting. Specifically, inspired by the continuity equation, MoCast introduces two core innovations: (1) a physics-guided motion module that learns turbulent motions from physically interpretable mean and fluctuating components based on Reynolds, Helmholtz, and Wavelet decomposition techniques, and (2) a motion-guided source-sink module that learns source-sink features considering the multi-scale impact from motions based on a mixture-of-experts architecture. Extensive experiments on three real-world datasets demonstrate that MoCast achieves the state-of-the-art performance. MoCast and its diffusion-based variant MoCast+ reduce CSI error by an average of 4.9% and 4.5% compared to the best deterministic and probabilistic baselines, respectively.

Introduction

Precipitation nowcasting refers to the short-term prediction of precipitation, typically within the next six hours, for a specific region (Zhong et al. 2024). It is crucial for wide-ranging weather-sensitive applications (e.g., aviation planning, urban flood control, and disaster management) (Espeholt et al. 2022). Due to the chaotic nature of precipitation evolution, this task poses significant challenges.

Recent advances in deep learning have shown strong potential in this field, offering higher accuracy and efficiency than traditional numerical methods (Shi et al. 2017; Ravuri et al. 2021; Wu et al. 2024a; Zhong et al. 2024). Most deep learning methods treat nowcasting as a spatial-temporal prediction task, using precipitation frame sequences derived from radar, satellite, or other products (Yu et al. 2024). Among purely deep learning methods, many methods aim to capture spatiotemporal patterns and generate deterministic forecasts (Gao et al. 2022a; Bai et al. 2022; Wu et al. 2024c). Nevertheless, they often overlook inherent stochasticity, leading to blurry results. To address this, some methods employ probabilistic models (Ravuri et al. 2021; Leinonen et al. 2023; Gao et al. 2023) to model the precipitation evolution stochastically. While improving visual realism, these methods may compromise spatiotemporal consistency (e.g., structure integrity and movement tendency) due to excessive randomness. Building on this foundation, hybrid methods first use deterministic models to ensure spatiotemporal consistency, then refine forecasts with probabilistic approaches to capture stochasticity (Yu et al. 2024; Gong et al. 2024), leading to strong overall performance. Despite this, these methods tend to neglect the physical laws governing precipitation evolution, which may limit their physical plausibility.

To address this limitation, some methods embed physical laws (e.g., conservation of moisture and momentum) into model architectures (Zhang et al. 2023b; Wang, Zhang, and Bai 2025) and loss functions (Li et al. 2023; Yin et al. 2024). A key purpose of these physics-guided methods is to model motions for advection, as motions govern moisture transport and help maintain spatiotemporal consistency (An et al. 2025). They often utilize optical flows as motions (Ayzel, Heistermann, and Winterrath 2019; Ha and Lee 2024) or parameterize motions (Zhang et al. 2023b; Lin et al. 2025) as a unified entity. However, these methods oversimplify motions, potentially overlooking distinct structural dynamics arising from complex physical processes. For example, in a supercell thunderstorm, small-scale vortices coexist with strong wind shear, embedded within the storm's larger-scale background flow. These combined motions (e.g., rotational, translational, and irregular components) stem from different physical processes and jointly influence precipitation evolution (Markowski and Richardson 2011). These insights motivate modeling motion from a decomposable perspective to address structural complexity.

Despite its significance, decomposing motions in precipitation systems remains challenging. Due to complex atmospheric dynamics, motions are **highly turbulent**, involving eddies, swirls, and rapid variations across time and space (Sura 2011). Many existing decomposition meth-

*Corresponding authors.

ods assume smoothness or continuity (Bhatia et al. 2012; Xing et al. 2024), making them ineffective at capturing the turbulence (e.g., high-frequency or small-scale structures). Moreover, motions are **tightly coupled with moisture processes** through kinetic and thermal energy exchange (Feingold et al. 2010). For example, divergence facilitates rapid horizontal spreading, resulting in sinks (i.e., moisture removal from the precipitation system). Eddies enhance collision–coalescence by increasing velocity fluctuations among cloud droplets, thereby accelerating raindrop formation and contributing to sources (i.e., moisture addition within the system) (Markowski and Richardson 2011).

To this end, we propose MoCast, a novel method that learns and utilizes physics-guided turbulent motions for precipitation nowcasting. Specifically, MoCast decomposes turbulent motions into structural components guided by turbulence knowledge and adapts their diverse effects to model multi-scale moisture evolution. The main contributions are as follows:

- We introduce a Physics-guided Motion Modeling (PMM) module. Based on Reynolds decomposition, PMM learns turbulent motions from the mean and fluctuating components. The mean component is derived from curl-free and divergence-free velocity fields guided by the Helmholtz decomposition, while the fluctuating component is derived from the high-frequency features of movement tendencies using Wavelet decomposition. This design enables PMM to learn motions under structural priors with strong physical interpretability.
- We introduce a Motion-guided Source-sink Modeling (MSM) module. MSM adopts a mixture-of-experts architecture, where each expert modulates scale-specific source-sink representations via adaptive normalization guided by different motion components. This design enables MSM to learn source-sink features considering diverse atmospheric dynamics.
- Extensive experimental results on three real-world datasets demonstrate that MoCast and its diffusion-based variant MoCast+ reduce CSI error by an average of 4.9% and 4.5% compared to the best deterministic and probabilistic baselines, respectively.

Related Work

Most deep learning methods for precipitation nowcasting are purely data-driven and can be grouped into deterministic, probabilistic, and hybrid methods. Deterministic methods learn spatiotemporal patterns directly, evolving from CNNs (Qiu et al. 2017; Ayzel et al. 2019) and RNNs (Shi et al. 2015, 2017) to attentions (Chang et al. 2021; Gao et al. 2022a; Bai et al. 2022) and FNOs (Wu et al. 2024b,c; Yan et al. 2024). Earthformer (Gao et al. 2022a), as a notable example, uses cuboid attention and global vectors to capture local and global spatiotemporal patterns. Nevertheless, due to ignoring inherent stochasticity, these models average multiple possible futures, producing blurry and over-smoothed predictions. Probabilistic methods address this by modeling uncertainty (Yu et al. 2025). They use GANs (Ravuri et al. 2021; Zhang et al. 2023b) and diffusions (Leinonen et al. 2023; Gao et al. 2023; Wang, Zhang, and Dodgson 2024; Wen et al. 2024) to model precipitation evolution as stochastic. For instance, DGMR (Ravuri et al. 2021) employs conditional GANs, while PreDiff (Gao et al. 2023) uses latent diffusion. Though visually sharper, such methods risk excessive randomness and may generate artifacts. Combining the previous two paradigms, hybrid methods have been developed (Yu et al. 2024; Gong et al. 2024; She et al. 2024). They typically start with deterministic forecasts and then refine them with probabilistic methods. DiffCast (Yu et al. 2024) and CasCast (Gong et al. 2024) are representative examples, which couple base models (Guen and Thome 2020; Gao et al. 2022b) with diffusions to enhance visual details. Nevertheless, these methods neglect the underlying physical laws governing precipitation evolution, which may result in physically implausible results.

Some methods incorporate physical laws either by enforcing equation-based constraints (e.g., moisture conservation) (Li et al. 2023; Yin et al. 2024) or by designing computation graphs guided by physical principles (e.g., continuity) (Zhang et al. 2023b; Wang, Zhang, and Bai 2025). A key component in these approaches is explicit motion modeling, as motions control moisture transport and ensure spatiotemporal consistency. Some methods use optical flows (Ayzel, Heistermann, and Winterrath 2019; Ha and Lee 2024), while others employ parameterized motions (Zhang et al. 2023b; Wang, Zhang, and Bai 2025) to capture precipitation evolution. However, these models typically treat motion as a single unified field, lacking explicit structural priors to represent diverse motion dynamics.

To this end, we propose MoCast, the first work to explicitly incorporate turbulence knowledge to learn and utilize turbulent motions. In brief, we capture motions by decomposing them into solvable components and utilize motion components to adjust source-sink learning.

Preliminary

Task Formulation. Given historical precipitation frames $\boldsymbol{X}^{1:T} \in \mathbb{R}^{T\times H\times W\times C}$, precipitation nowcasting aims to predict future frames $\boldsymbol{X}^{T+1:T+P} \in \mathbb{R}^{P\times H\times W\times C}$, where T and P are the historical and future lengths, respectively. H, W, and C are the height, width, and channel number. A model $\mathcal{F}$, parameterized by $\boldsymbol{\Theta}$, is trained to minimize the loss $\mathcal{L}$ between predictions and ground truth:

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} \mathcal{L}(\mathcal{F}(\boldsymbol{X}^{1:T};\boldsymbol{\Theta}), \boldsymbol{X}^{T+1:T+P}). \quad (1)$$

Continuity Equation for Precipitation Evolution. Precipitation evolution can be conceptualized as a spatial movement of quantities over time governed by the continuity equation (Zhang et al. 2023b), formulated as:

$$\underbrace{\frac{\partial \boldsymbol{x}}{\partial t}}_{\text{Time evolution term}} + \underbrace{\nabla\cdot(\boldsymbol{v}\boldsymbol{x})}_{\text{Advection term}} = \underbrace{\boldsymbol{s}}_{\text{Source-sink term}}, \quad (2)$$

where $\boldsymbol{x}$, $\boldsymbol{v}$, and $\boldsymbol{s}$ represent the precipitation quantities (e.g., integrated liquid), motions (e.g., wind velocity), and sources/sinks (e.g., condensation and evaporation), respectively. The time-evolution term describes temporal changes

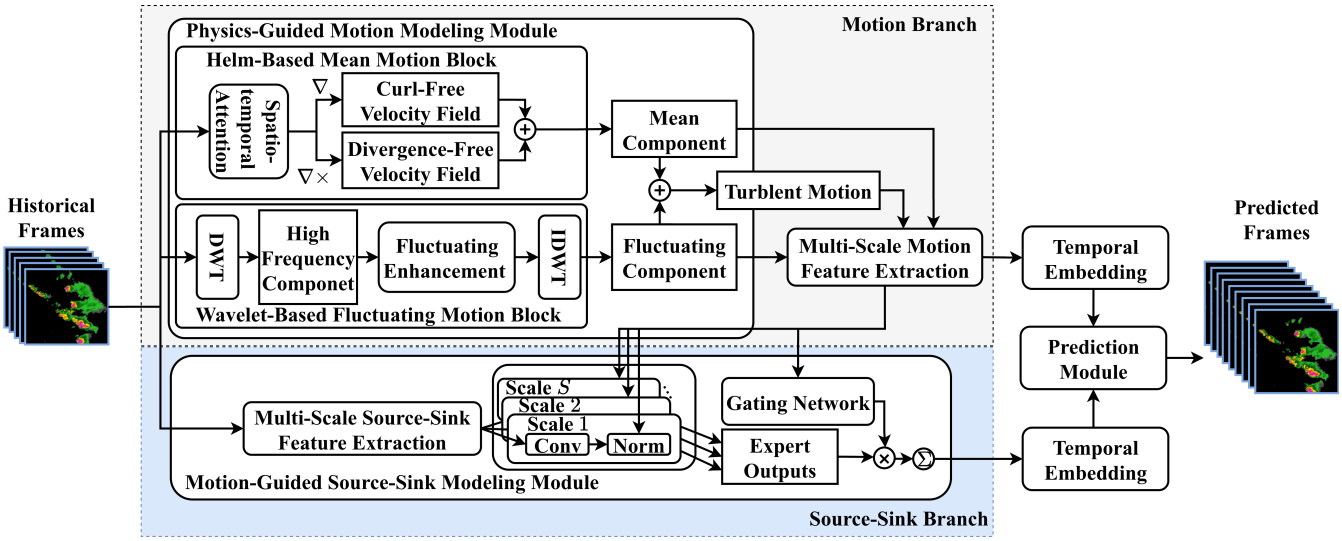

Figure 1: Framework of MoCast.

within an area, the advection term captures motion-driven transport, and the source–sink term represents gains or losses of precipitation quantities (Qiu, Bao, and Xu 1993).

Reynolds Decomposition. Reynolds decomposition is fundamental in turbulence theory, where any instantaneous flow can be split into average and fluctuating components (Alfonsi 2009). For the velocity field, the decomposition is formulated as:

$$\boldsymbol{v} = \overline{\boldsymbol{v}} + \boldsymbol{v}', \tag{3}$$

where $\boldsymbol{v}$, $\overline{\boldsymbol{v}}$, and $\boldsymbol{v}'$ represent the instantaneous, time-averaged, and fluctuating velocity, respectively.

Helmholtz Decomposition. The Helmholtz decomposition theorem states that a sufficiently smooth and rapidly decaying vector field can be resolved into the sum of a curl-free vector field and a divergence-free vector field (Bhatia et al. 2012; Xing et al. 2024), formulated as:

$$\boldsymbol{F} = \underbrace{\nabla\Phi}_{\text{Curl-free}} + \underbrace{\nabla \times \boldsymbol{A}}_{\text{Divergence-free}}, \tag{4}$$

where $\boldsymbol{F}$ is the target vector field (e.g., the velocity field). Φ and $\boldsymbol{A}$ are the scaler and vector potential, respectively.

Methodology

Overview. Fig. 1 illustrates the framework of MoCast. Guided by the continuity equation (Eq. 2), precipitation evolution is driven by two relatively independent yet interconnected processes: advection and source-sink (Zhang et al. 2023b). Accordingly, we adopt a two-branch paradigm to model the evolution. In the **motion branch**, the PMM module learns turbulent motions for advection. PMM models complex motion from physically interpretable components, each capturing a specific dynamic pattern. These components are summed to reconstruct the full motion field. Specifically, we apply Reynolds decomposition as a structural prior, separating motions into mean and fluctuating components to capture global transport and local variability. The mean component is further decomposed via Helmholtz decomposition into divergent and rotational motions for finer-grained modeling. The fluctuating part is modeled via wavelet decomposition to capture directionally sensitive variations. The mean and fluctuating components are added to get turbulent motions for advection. In the **source-sink branch**, the MSM module learns source-sink features under the multi-scale guidance of different motion components, enabling the modeling of motion-induced moisture processes. Subsequently, motion and source-sink features are individually processed by temporal embedding to capture temporal dependencies. The embedded features are then separately forecasted by the prediction module and fused to produce the final output.

Physics-Guided Motion Modeling Module

Helm-Based Mean Motion Block

Spatiotemporal Attention. Due to diffusion and dynamics, precipitation at one location can influence nearby regions over time. To model this, we use attention mechanisms to capture spatiotemporal correlations across frames, enhanced with relative coordinates for spatial awareness. The resulting attention scores implicitly encode motion patterns and thus serve as basic motion cues (Zhong et al. 2023; Zhang et al. 2023a). We first encode the inputs $\boldsymbol{X}^{1:T}$ into precipitation features $\boldsymbol{H}_{\mathrm{m}}$ by 2D convolutionals. We then partition $\boldsymbol{H}_{\mathrm{m}}$ into non-overlapping patches of size $p \times p$. For each patch, its spatiotemporal neighbors are defined as the surrounding patches within a $K \times K$ window at the next time step. For a patch $\boldsymbol{p}_i^t \in \mathbb{R}^{p\times p\times d}$ at position i and time t, attention scores with its spatiotemporal neighbors $\boldsymbol{p}_{\mathcal{N}(i)}^{t+1} \in \mathbb{R}^{K\times K\times p\times p\times d}$ at time $t+1$ are formulated as:

$$\begin{aligned}
\boldsymbol{Q}_i^t &= \mathrm{Linear}(\mathrm{Flatten}(\boldsymbol{p}_i^t)),\\
\boldsymbol{K}_{\mathcal{N}(i)}^{t+1} &= \mathrm{Linear}(\mathrm{Flatten}(\boldsymbol{p}_{\mathcal{N}(i)}^{t+1})),\\
\boldsymbol{\alpha}_i^{t\to t+1} &= \mathrm{SoftMax}(\boldsymbol{Q}_i^t(\boldsymbol{K}_{\mathcal{N}(i)}^{t+1})^T/d_{\mathrm{e}}),
\end{aligned} \tag{5}$$

where the attention map $\boldsymbol{\alpha}_i^{t\to t+1} \in \mathbb{R}^{K\times K}$ captures the similarity between patch $\boldsymbol{p}_i^t$ and its neighbors. To enrich spatial awareness, we introduce a position matrix $\boldsymbol{D}^{2\times h_d\times w_d}$, representing normalized relative coordinates of patches. For position i, the motion cue is computed by weighting neighbor coordinates with the attention map, formulated as:

$$\widetilde{\boldsymbol{D}}_i^{t\to t+1} = \boldsymbol{\alpha}_i^{t\to t+1} \odot \boldsymbol{D}_{\mathcal{N}(i)} - \boldsymbol{D}_i, \tag{6}$$

where $\widetilde{\boldsymbol{D}}_i^{t\to t+1} \in \mathbb{R}^{2\times K\times K}$ denotes the motion cue at position i. For all positions, the motion cues from time t to $t+1$ is represented as $\widetilde{\boldsymbol{D}}^{t\to t+1} \in \mathbb{R}^{2\times K\times K\times h_d\times w_d}$.

Curl-Free and Divergence-Free Term. In precipitation systems, convergence/divergence and vorticity coexist and interact nonlinearly. Although strict separation into curl-free and divergence-free components is idealized, existing works (Xing et al. 2024) provide a useful approximation based on Helmholtz decomposition, modeling the two components via learnable scalar and vector potentials. Motion cues $\widetilde{\boldsymbol{D}}$ estimated from spatiotemporal patch-level similarity suppress pixel noise and produce smooth fields, making them suitable for potential fields. Thus, the mean motion component can be derived from the motion cue, formulated as:

$$\begin{aligned}&\boldsymbol{\Phi}^{t\to t+1} = f_\Phi(\widetilde{\boldsymbol{D}}^{t\to t+1}), \boldsymbol{A}^{t\to t+1} = f_{\boldsymbol{A}}(\widetilde{\boldsymbol{D}}^{t\to t+1}),\\ &\boldsymbol{M}_{\mathrm{m}}^{t\to t+1} = \nabla\boldsymbol{\Phi}^{t\to t+1} + \nabla\times\boldsymbol{A}^{t\to t+1},\end{aligned} \tag{7}$$

where f_Φ and f_A are two convolutional networks used to obtain the scaler and vector potential (i.e., $\boldsymbol{\Phi}^{t\to t+1} \in \mathbb{R}^{h_d\times w_d}$ and $\boldsymbol{A}^{t\to t+1} \in \mathbb{R}^{h_d\times w_d}$), respectively. The curl-free and divergence-free velocity fields are derived by applying the gradient and the curl operators on these two potentials. The mean component $\boldsymbol{M}_{\mathrm{m}}^{t\to t+1} \in \mathbb{R}^{2\times h_d\times w_d}$ is obtained.

In addition, since the mean component is time-averaged in Reynolds decomposition, a motion trend-consistency loss is introduced to enforce temporal coherence of motions and avoid overly rigid constraints, balancing trend awareness and local flexibility (detailed in Training Loss).

Wavelet-Based Fluctuating Motion Block

Due to the inherent turbulence, capturing small-scale directional fluctuations is crucial for motion modeling. The discrete wavelet transform (DWT), with its directional sensitivity, enables the extraction of such features by decomposing precipitation fields into orientation-aware frequency components (Othman and Zeebaree 2020). Given the precipitation features $\boldsymbol{H}_{\mathrm{m}}^t$ at time t, we apply DWT to extract high-frequency components in multiple directions, formulated as: $\boldsymbol{H}_{\mathrm{h}}^t, \boldsymbol{H}_{\mathrm{v}}^t, \boldsymbol{H}_{\mathrm{d}}^t = \mathrm{DWT}(\boldsymbol{H}_{\mathrm{m}}^t)$, where $\boldsymbol{H}_{\mathrm{h}}^t, \boldsymbol{H}_{\mathrm{v}}^t, \boldsymbol{H}_{\mathrm{d}}^t$ represent the high-frequency features in the horizontal, vertical, and diagonal directions, respectively.

Although the high-frequency features capture rich fluctuations, they also contain considerable noise. To enhance representation, we incorporate spatial gradients from consecutive frames. Gradients are first computed from the original frame and then used to enhance direction-specific fluctuations, which can selectively emphasize informative regions. Taking the horizontal direction as an example, for two consecutive frames at time t and $t+1$, the fluctuating features enhanced by reweighting are formulated as:

$$\begin{aligned}\boldsymbol{H}_{\mathrm{h}}'^t &= \mathrm{Sigmoid}(f_{\mathrm{h}}(\nabla\boldsymbol{X}^t)) \odot \boldsymbol{H}_{\mathrm{h}}^t,\\ \boldsymbol{H}_{\mathrm{h}}'^{t+1} &= \mathrm{Sigmoid}(f_{\mathrm{h}}(\nabla\boldsymbol{X}^{t+1})) \odot \boldsymbol{H}_{\mathrm{h}}^{t+1},\\ \boldsymbol{H}_{\mathrm{h}}^{t\to t+1} &= g_{\mathrm{h}}(\boldsymbol{H}_{\mathrm{h}}'^t || \boldsymbol{H}_{\mathrm{h}}'^{t+1}),\end{aligned} \tag{8}$$

where f_{h} is a direction-specific network to extract features about movement tendencies. The sigmoid activation function scales the values between 0 and 1. g_{h} is a network to extract features about changes across time steps. $||$ denotes concatenation.

After applying the reweighting mechanisms for each direction, we perform an inverse wavelet transform to capture the fluctuating component, formulated as

$$\begin{aligned}\boldsymbol{H}_{\mathrm{f}}^{t\to t+1} &= \mathrm{IDWT}(\boldsymbol{H}_{\mathrm{h}}^{t\to t+1}, \boldsymbol{H}_{\mathrm{v}}^{t\to t+1}, \boldsymbol{H}_{\mathrm{d}}^{t\to t+1}),\\ \boldsymbol{M}_{\mathrm{f}}^{t\to t+1} &= f_{\mathrm{f}}^{t+1}(\boldsymbol{H}_{\mathrm{f}}^{t\to t+1}),\end{aligned} \tag{9}$$

where f_{f} is a convolutional network to learn fluctuating components. $\boldsymbol{M}_{\mathrm{f}}^{t\to t+1} \in \mathbb{R}^{2\times h_d\times w_d}$ is the fluctuating components from time step t to $t+1$.

According to the Reynolds decomposition, we add the mean component (Eq.7) and and fluctuating component (Eq.9) to obtain the turbulent motions in advection process (i.e., $\boldsymbol{M}_{\mathrm{a}} = \boldsymbol{M}_{\mathrm{m}} + \boldsymbol{M}_{\mathrm{f}} \in \mathbb{R}^{(T-1)\times 2\times h_d\times w_d}$).

Motion-Guided Source-Sink Modeling Module

Different components of motion exert distinct influences on source-sink processes. Moreover, due to the multi-scale cascading interactions, source-sink processes are modulated by motions on different spatial scales (Gong et al. 2024). To capture multi-scale modulation, the MSM module employs a mixture-of-experts architecture, where each expert is guided by scale-specific motion components to effectively learn the corresponding source-sink features.

Multi-Scale Motion and Source-Sink Feature Extraction. Given precipitation frames $\boldsymbol{X}$, we apply 2D convolutions with varying kernel sizes to extract multi-scale source–sink features, denoted as $\boldsymbol{E}^s$ for spatial scale s. For the mean $\boldsymbol{M}_{\mathrm{m}}$, fluctuating $\boldsymbol{M}_{\mathrm{f}}$, and total motions $\boldsymbol{M}_{\mathrm{a}}$, we similarly extract multi-scale motion features using multi-kernel convolutions. The resulting representation $\boldsymbol{M}^s$ concatenates $\boldsymbol{M}_{\mathrm{m}}^s$, $\boldsymbol{M}_{\mathrm{f}}^s$, and $\boldsymbol{M}_{\mathrm{a}}^s$ at scale s, capturing scale-specific motion characteristics.

Motion-Guided Experts. Inspired by adaptive normalization in cross-domain fusion (Peebles and Xie 2023; Lin et al. 2024), at spatial scale s, the expert uses the motion features $\boldsymbol{M}^s$ as guidance to modulate the source-sink features $\boldsymbol{E}^s$. Since $\boldsymbol{M}^s$ comprises three types of motion (i.e., mean, fluctuating, and total), each influencing moisture differently, we separately generate three groups of scale and shift parameters. Taking the total motion as an example, The adjustment is formulated as:

$$\boldsymbol{\gamma}_{\mathrm{a}}^s, \boldsymbol{\beta}_{\mathrm{a}}^s = f_{\mathrm{a}}^s(\boldsymbol{M}^s), \quad \boldsymbol{E}_{\mathrm{a}}^{s\prime} = \boldsymbol{\gamma}_{\mathrm{a}}^s \cdot \boldsymbol{E}_{\mathrm{a}}^s + \boldsymbol{\beta}_{\mathrm{a}}^s, \tag{10}$$

where f_{a}^s is a convolutional network that generates scale ($\boldsymbol{\gamma}_{\mathrm{a}}^s \in \mathbb{R}^{h^s\times w^s}$) and shift ($\boldsymbol{\beta}_{\mathrm{a}}^s \in \mathbb{R}^{h^s\times w^s}$) parameters from the total motion. Similarly, we derive adjusted source–sink

features from the mean and fluctuating motions. Given the source–sink features modulated by all three motion components, the final adjusted representation is obtained through a linear projection, i.e., $\boldsymbol{E}^{s\prime} = \mathrm{Linear}(\boldsymbol{E}^{s\prime}_{\mathrm{a}}, \boldsymbol{E}^{s\prime}_{\mathrm{m}}, \boldsymbol{E}^{s\prime}_{\mathrm{f}})$, where $\boldsymbol{E}^{s\prime}$ is the output of each scale-specific expert.

Motion-Adaptive Gating Network. At each spatial location, we compute a soft expert selection vector guided by motion features. We concatenate the multi-scale motion features and feed them into a lightweight network to compute a spatially adaptive softmax over all experts. The resulting selection weights $\boldsymbol{W}_{\text{gate}} \in \mathbb{R}^{h_d \times w_d \times S}$ indicate the contribution of each expert (or scale) at each spatial location, where S is the total number of experts. Given the expert outputs and gating weights, the outputs is formulated as:

$$\boldsymbol{E} = \sum_{s=1}^{S} \boldsymbol{W}^{s}_{\text{gate}} \odot \mathrm{Resize}(\boldsymbol{E}^{s\prime}), \tag{11}$$

where $\boldsymbol{E} \in \mathbb{R}^{T \times h_d \times w_d \times d}$ are the learned source-sink features. Resize ensures a consistent shape by padding between scales.

Temporal Embedding and Prediction Module

Given the motion features and source-sink features, we apply separate temporal embedding networks to capture their respective temporal dependencies. Each embedding network comprises a bottleneck 1×1 Conv2D layer followed by group convolutional operators (Gao et al. 2022b).

We then predict the motion and source-sink terms independently for all time steps. The prediction network consists of two sequential CNN-based blocks: the first maps the embedded temporal dimension to the target prediction horizon; the second transforms the embedded spatial dimension to the target resolution.

Given the predicted motions $\widehat{\boldsymbol{M}} \in \mathbb{R}^{P \times H \times W \times 2}$ and source-sink terms $\widehat{\boldsymbol{S}} \in \mathbb{R}^{P \times H \times W}$, following the prior work (Zhang et al. 2023b), we fuse them via advection and additive operations to obtain the final prediction, formulated as:

$$\widehat{\boldsymbol{X}}^{t+1} = \mathrm{Adv}(\widehat{\boldsymbol{X}}^{t}, \widehat{\boldsymbol{M}}^{t \to t+1}) + \widehat{\boldsymbol{S}}^{t+1}, \tag{12}$$

where $t \in [T, T+P-1]$. Adv is an advection operator implemented via a warping mechanism (Zhang et al. 2023b). At the initial time step $t = T$, the last input frame is used to initialize the prediction (i.e., $\widehat{\boldsymbol{X}}^{T} = \boldsymbol{X}^{T}$).

Training Loss

As emphasized in the PMM module, to balance trend awareness and local flexibility, we introduce a motion trend-consistency loss that enforces temporal coherence of motion and avoids overly rigid constraints. To mitigate the risk of introducing artifacts from background or static regions, we impose motion trend-consistency primarily on precipitation-effective areas, where temporal smoothness is more essential. The motion loss is formulated as:

$$\mathcal{L}_{\text{motion}} = \frac{1}{\sum_{t,i,j} \boldsymbol{W}^{t}_{i,j}} \sum_{t=1}^{T-2} \sum_{i=1}^{h_d} \sum_{j=1}^{w_d} \boldsymbol{W}^{t}_{i,j} || \boldsymbol{M}^{t+1}_{\text{m}i,j} - \boldsymbol{M}^{t}_{\text{m}i,j} ||_2^2 \tag{13}$$

where $\boldsymbol{M}_{\text{m}}$ is the learned mean component. $\boldsymbol{W}$ is a binary mask that highlights regions with meaningful precipitation. To ensure temporal alignment and spatial consistency, the mask is derived by evaluating precipitation intensity in the corresponding precipitation frames and subsequently downsampled via average pooling. Specifically, for each $t \in [1, T-2]$, the mask is defined as $\boldsymbol{W}^{t} = \mathrm{Binary}[\max(\mathrm{avg}(\boldsymbol{X}^{t}), \mathrm{avg}(\boldsymbol{X}^{t+1})) > \theta]$, where θ is a dataset-provided threshold indicating significant precipitation, determined from dataset statistics.

In addition, we employ the pixel-wise Mean Squared Error (MSE) $\mathcal{L}_{\text{precip}}$ to supervise the pixel-level precipitation values. The total training loss is formulated as:

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{precip}} + \lambda \mathcal{L}_{\text{motion}}, \tag{14}$$

where λ is a weighting coefficient controlling the influence of the motion trend-consistency loss.

Experiment

Experimental Settings

Datasets. We conduct experiments on Storm EVent ImagRy (SEVIR) (Veillette, Samsi, and Mattioli 2020), MeteoNet (Larvor et al. 2020), and Shanghai (Chen et al. 2020), which span nation-wise, region-wise, and city-wise meteorological systems, respectively. SEVIR (USA, 2017–2019) provides Vertically Integrated Liquid (VIL) data at 5-minute intervals and 1 km spatial resolution. MeteoNet (France, 2016–2018) offers radar reflectivity from two regions at 5-minute intervals and 0.01°. Shanghai (China, 2015–2018) contains radar echoes over Pudong at 6-minute intervals and 0.01° resolution. All datasets are preprocessed following (Yu et al. 2024), including cropping, normalization, and resizing to 128×128. The input and output sequence lengths are set to 5 and 20, respectively.

MoCast and MoCast+. Following the hybrid paradigm (Yu et al. 2024), we extend MoCast with a probabilistic diffusion-based post-processing module to capture stochasticity, referred to as MoCast+. Given the basic predictions $\boldsymbol{\mu} = \mathrm{MoCast}(\boldsymbol{X})$, MoCast+ constructs residual frames $\boldsymbol{r} = \boldsymbol{Y} - \boldsymbol{\mu}$, where $\boldsymbol{X}$ is the input and $\boldsymbol{Y}$ the ground truth. These residuals are then modeled through a Gaussian diffusion process using a UNet backbone enhanced with temporal attention and ConvGRU to capture contextual information.

Baselines. We compare MoCast and its variant MoCast+ against 12 competitive baselines. To ensure a fair comparison, we adopt a two-tier evaluation: MoCast, being fully deterministic, is evaluated against deterministic baselines, including PhyDNet (Guen and Thome 2020), MAU (Chang et al. 2021), SimVP (Gao et al. 2022b), Earthformer (Gao et al. 2022a), MIMO (Ning et al. 2023), and PastNet (Wu et al. 2024c). MoCast+, which integrates a diffusion-based post-processing module, is evaluated against non-deterministic baselines that incorporate stochastic generation or refinement, including STRPM (Chang et al. 2022), MCVD (Voleti, Jolicoeur-Martineau, and Pal 2022), NowCast (Zhang et al. 2023b), PreDiff (Gao et al. 2023), DiffCast (Yu et al. 2024), and CasCast (Gong et al. 2024).

Dataset	Metric	Deterministic							Non-deterministic						
		PhyDNet 2020	MAU 2021	SimVP 2022	Earthformer 2022	MIMO 2023*	PastNet 2024*	MoCast (Ours)	STRPM 2022	MCVD 2022	Nowcast 2023*	PreDiff 2023	DiffCast 2024	CasCast 2024*	MoCast+ (Ours)
SEVIR	↑CSI	0.2560	0.2463	0.2662	0.2513	0.2701	0.2894	**0.3114**	0.2512	0.2148	0.2101	0.2304	0.3077	0.3128	**0.3310**
	↑CSI-P4	0.2685	0.2566	0.2844	0.2617	0.2878	0.3073	**0.3279**	0.3243	0.3020	0.2819	0.3041	0.4122	0.4092	**0.4354**
	↑CSI-P16	0.3005	0.2861	0.3452	0.2910	0.3158	0.3235	**0.3474**	0.4959	0.4706	0.3791	0.4028	0.5683	0.5375	**0.5974**
	↑HSS	0.3124	0.3004	0.3369	0.3073	0.3441	0.3529	**0.3896**	0.3277	0.2743	0.2776	0.2986	0.4033	0.3919	**0.4190**
	↓LPIPS	0.3785	0.3933	0.3914	0.4140	0.4000	0.3609	**0.3430**	0.2577	0.2170	0.2029	0.2851	0.1812	0.1802	**0.1756**
	↑SSIM	0.6764	0.6361	0.6304	0.6773	0.6178	0.6246	**0.6776**	0.6513	0.5265	0.5658	0.5185	0.6354	0.6563	**0.6783**
MeteoNet	↑CSI	0.3384	0.3232	0.3346	0.3296	0.3237	0.3329	**0.3489**	0.2606	0.2336	0.2231	0.2657	0.3511	0.3246	**0.3552**
	↑CSI-P4	0.3824	0.3304	0.3383	0.3428	0.3298	0.3492	**0.3875**	0.4138	0.3841	0.3697	0.3854	**0.5081**	0.4486	0.4863
	↑CSI-P16	**0.4986**	0.4165	0.4143	0.4333	0.4043	0.4002	0.4857	0.6882	0.6128	0.5365	0.5692	0.7155	0.6797	**0.7184**
	↑HSS	0.4673	0.4451	0.4568	0.4604	0.4466	0.4339	**0.4738**	0.3688	0.3393	0.3248	0.3782	0.4846	0.4165	**0.4918**
	↓LPIPS	0.2941	0.3089	0.3523	0.3718	0.3189	0.3674	**0.2917**	0.2004	0.1652	0.1848	0.1543	**0.1198**	0.1467	0.1266
	↑SSIM	0.8022	0.7897	0.7557	0.7899	0.7703	0.7779	**0.8146**	0.5996	0.5414	0.5047	0.7059	0.7887	0.7283	**0.8012**
Shanghai	↑CSI	0.3653	0.3996	0.3841	0.3575	0.3622	0.3769	**0.4156**	0.3606	0.2872	0.3338	0.3583	0.3955	0.4131	**0.4410**
	↑CSI-P4	0.4552	0.4695	0.4467	0.4008	0.4293	0.3951	**0.4810**	0.4944	0.3984	0.3134	0.4389	0.5116	0.5036	**0.5265**
	↑CSI-P16	0.5980	0.5787	0.5603	0.4863	0.4668	0.4753	**0.5997**	0.6783	0.5675	0.3137	0.5448	0.6576	0.6381	**0.6678**
	↑HSS	0.4957	0.5356	0.5183	0.4843	0.4891	0.5140	**0.5528**	0.4931	0.4036	0.4674	0.4849	0.5296	0.5517	**0.5791**
	↓LPIPS	0.1894	0.2735	0.2984	0.2564	0.2734	0.2483	**0.1835**	0.1681	0.2081	0.2452	0.1696	0.1571	0.1579	**0.1561**
	↑SSIM	0.7751	0.7303	0.7764	0.7750	0.7344	0.7449	**0.7812**	0.7724	0.5119	0.6510	0.7557	0.7902	0.7755	**0.8092**

Table 1: Results of MoCast and MoCast+ compared with deterministic and non-deterministic baselines, respectively. The best results are bolded, and the second best results are underlined. The results with $*$ are rerun by us to meet our settings using their official codes, while those without $*$ are cited from DiffCast (Yu et al. 2024).

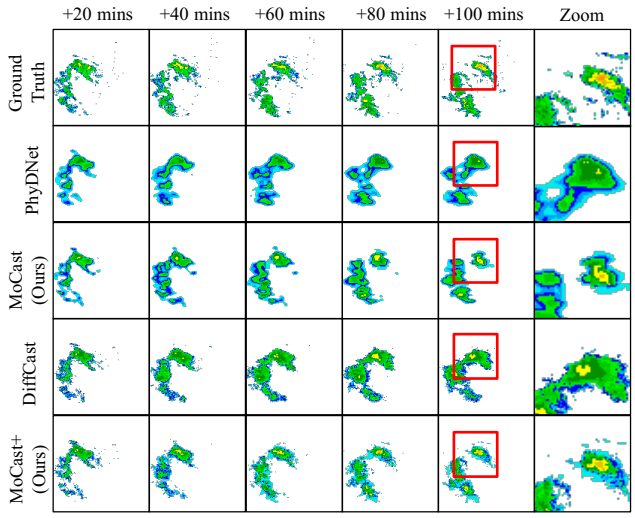

Figure 2: Example of prediction results from MeteoNet. We compare MoCast and MoCast+ with the best deterministic and non-deterministic baselines (PhyDNet and DiffCast).

Metrics. (1) Critical success index(CSI): measuring the accuracy considering hits and errors. CSI-P4 and CSI-P16 are processed by 4×4 and 16×16 max-pooling, respectively. (2) Heidke skill score (HSS): measuring the accuracy compared to random chance. (3) Learned perceptual image patch similarity (LPIPS): measuring perceptual quality. (4) Structural similarity index measure (SSIM): measuring structural similarity. Notably, the frames are binarized to calculate CSI and HSS at thresholds [16, 74, 133, 160, 181, 219], [12, 18, 24, 32], and [20, 30, 35, 40], on the SEVIR, MeteoNet, and Shanghai datasets, respectively. The final CSI and HSS scores reported are averaged over these thresholds.

Training details. MoCast is trained with AdamW (lr=$1e$-4) and cosine scheduling for up to 200 epochs with early stopping (step=20). The batch size is set to 6. We leverage AutoML toolkit NNI (Microsoft 2021) to efficiently search for optimal hyperparameters with reduced computational overhead. All experiments are run on a single A100 GPU.

Main Results

Table 1 summarizes the results of all methods. (1) **MoCast vs. Deterministic Baselines.** MoCast outperforms the strongest baselines with an average 4.9% reduction on CSI. MoCast also attains the best LPIPS and SSIM scores on all datasets, reflecting superior perceptual fidelity and structural consistency. Representative visualizations are shown in Fig. 2. (2) **MoCast+ vs. Non-deterministic Baselines.** MoCast+ outperforms the strongest baselines with an average 4.5% reduction on CSI. Notably, MoCast+ surpasses DiffCast by 2.4% on CSI-P16, highlighting its improved ability to capture high-intensity precipitation events. Since MoCast+ and DiffCast adopt the same diffusion-based post-processing, such performance gains of MoCast+ stem from its physically grounded backbone (i.e., MoCast), which more faithfully reflects the underlying dynamics of precipitation evolution. (3) **MoCast vs. MoCast+.** MoCast+ demonstrates a clear quantitative advantage over MoCast. On LPIPS, which measures perceptual quality, MoCast+ achieves an average improvement of over 40%. A qualitative comparison further highlights the distinction. As shown in Fig. 2, MoCast tends to produce blurry forecasts, especially at longer lead times, missing small-scale variations. In contrast, MoCast+ captures small-scale variations across all horizons.

Detailed Analysis

Long-Term Prediction. As shown in Table 2, MoCast's advantage grows with prediction length, reducing CSI error by 7.6% to 18.21% for longer-range forecasts. We attribute this to MoCast's ability to learn physically meaningful motions that constrain precipitation evolution, mitigating long-lead

Prediction Length	SimVP (2022)	MIMO (2023)	PastNet (2024)	MoCast (Ours)	Impro.
20	0.2662	0.2701	0.2894	**0.3114**	7.60 %
30	0.2271	0.1922	0.2397	**0.2728**	13.81 %
40	0.1774	0.1499	0.1673	**0.2097**	18.21 %

Table 2: CSI of long-term predictions on SEVIR.

Variant	PMM module			MSM module		MoCast
	w/o M	w/o F	w/o PMM	w/o MS	w/o MSM	
Shanghai	0.3682	0.3654	0.3595	0.3743	0.3801	**0.4156**
MeteoNet	0.3164	0.3185	0.3039	0.3063	0.3121	**0.3489**

Table 3: CSI of variants about PMM and MSM modules.

drift and improving stability.

Ablation. Table 3 and 4 summarize the results of variations. (1) **PMM module.** Removing the mean motion block (-w/o M), fluctuating motion block (-w/o F), and entire PMM module (-w/o PMM) consistently degrades performance, validating the effectiveness of our structured motion learning strategy. (2) **MSM module.** Removing the multi-scale design (-w/o MS) and entire MSM module (-w/o MSM) leads to performance drops, highlighting the effectiveness of our multi-scale motion-guided strategy for source-sink learning. (3) **Post-processing design.** We evaluate non-deterministic post-processing strategies using GAN (Zhang et al. 2023b) and diffusion (Yu et al. 2024). Both variants improve performance, highlighting the importance of uncertainty modeling in precipitation nowcasting.

Hyperparameter. (1) **Number of Spatial Scales/Experts** S**.** We vary S in $[1, 2, 3, 4]$. As shown in Fig. 3(a), the performance is best at 3. Fewer scales may limit resolution diversity, hindering effective motion-source interactions. More scales may introduce redundancy and dilute supervision, impairing expert specialization. (2) **Weight of Trend-Consistency Loss** λ**.** We vary λ in $\{0, 0.005, 0.01, 0.02, 0.04\}$, where 0 indicates the removal of the consistency loss. As shown in Fig. 3(b), the performance is best at 0.01. Smaller values may underweight temporal regularization, leading to unstable or inconsistent predictions. Larger values may over-constrain the model, suppressing flexibility.

Efficiency. We compare training time (h/epoch), GPU memory (GB), and CSI with competitive baselines. As shown in Table 5, MoCast emphasizes predictive skill, achieving 7.6% improvement in CSI over the best baseline (PastNet). This performance comes with a moderate increase in training cost: 1.20 × training time and 1.26 × GPU memory, primarily due to our multi-scale guidance design. Although our model incurs higher computational cost, in high-stakes precipitation nowcasting scenarios where improved accuracy significantly impacts decision-making (e.g., flood warnings), such trade-offs are justifiable.

Dataset	MoCast	+GAN	Impro.	+Diffusion	Impro.
Shanghai	0.4156	0.4252	2.31 %	0.4410	6.11 %
MeteoNet	0.3489	0.3503	0.40 %	0.3552	1.81 %

Table 4: CSI of variants about post-processing.

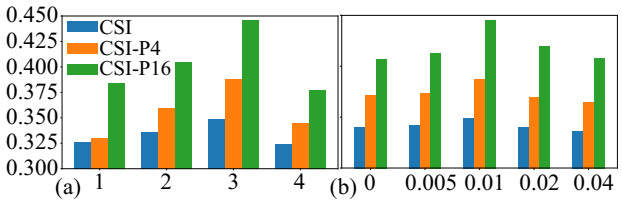

Figure 3: Results of hyperparameters on Shanghai. (a) Results of different scales. (b) Results of different loss weights.

Metric	SimVP	MIMO	PastNet	MoCast(S=2)	MoCast(S=3)
Training Time	4.31	8.76	5.84	6.23	7.02
GPU Memory	16.07	18.92	13.14	14.92	16.26
CSI	0.2662	0.2701	0.2894	0.3032	0.3114

Table 5: Results of efficiency study on SEVIR.

Case. We visualize a storm in 2019-08-13 in the SEVIR test set. From Fig. 4, we can observe that: (1) MoCast captures velocity patterns aligned with precipitation, with higher speeds in heavier rainfall areas (red color). (2) The mean component captures the moving direction of the high-intensity precipitation area, i.e., towards the upper-right as indicated in the squares. (3) The fluctuating component captures vortex-like structures. This can indicate source regions linked to precipitation cluster growth as marked in circles.

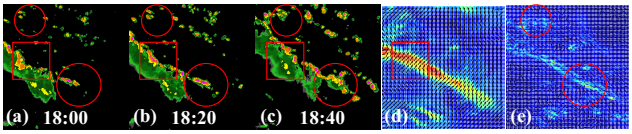

Figure 4: Motion visualization. (a)-(c) are ground-truth VIL frames. (d) and (e) are the mean component $\boldsymbol{M}_{\mathrm{m}}$ and fluctuating $\boldsymbol{M}_{\mathrm{f}}$ component at 18:20, learned from MoCast. In (d) and (e), the background color represents the magnitude of speed (individually normalized), while arrows indicate direction.

Conclusions and Future Work

In this work, we introduce MoCast to learn physics-guided turbulent motions via mean-fluctuation decomposition and source-sink terms via multi-scale motion-guided modeling. Extensive experimental results demonstrate that MoCast outperforms SOTA baselines in nowcasting accuracy, perceptual fidelity, and structural consistency. These results highlight a promising shift from purely data-driven to physics-guided in turbulent motion learning, which can offer better predictive power and interpretability. Moreover, we identify key opportunities for future work: (1) addressing subtle physical inconsistencies through multi-modal data fusion and (2) incorporating convective dynamics via 3D atmospheric modeling to capture extreme rainfall events.

References

Alfonsi, G. 2009. Reynolds-averaged Navier-Stokes equations for turbulence modeling. *Applied Mechanics Reviews*, 62(4): 20.

An, S.; Oh, T.-J.; Sohn, E.; and Kim, D. 2025. Deep learning for precipitation nowcasting: A survey from the perspective of time series forecasting. *Expert Systems with Applications*, 268: 126301.

Ayzel, G.; Heistermann, M.; Sorokin, A.; Nikitin, O.; and Lukyanova, O. 2019. All convolutional neural networks for radar-based precipitation nowcasting. *Procedia Computer Science*, 150: 186–192.

Ayzel, G.; Heistermann, M.; and Winterrath, T. 2019. Optical flow models as an open benchmark for radar-based precipitation nowcasting. *Geoscientific Model Development*, 12(4): 1387–1402.

Bai, C.; Sun, F.; Zhang, J.; Song, Y.; and Chen, S. 2022. Rainformer: Features extraction balanced network for radar-based precipitation nowcasting. *IEEE Geoscience and Remote Sensing Letters*, 19: 1–5.

Bhatia, H.; Norgard, G.; Pascucci, V.; and Bremer, P.-T. 2012. The Helmholtz-Hodge decomposition—a survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(8): 1386–1404.

Chang, Z.; Zhang, X.; Wang, S.; Ma, S.; and Gao, W. 2022. Strpm: A spatiotemporal residual predictive model for high-resolution video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13946–13955.

Chang, Z.; Zhang, X.; Wang, S.; Ma, S.; Ye, Y.; Xinguang, X.; and Gao, W. 2021. Mau: A motion-aware unit for video prediction and beyond. *Advances in Neural Information Processing Systems*, 34: 26950–26962.

Chen, L.; Cao, Y.; Ma, L.; and Zhang, J. 2020. A deep learning-based methodology for precipitation nowcasting with radar. *Earth and Space Science*, 7(2): e2019EA000812.

Espeholt, L.; Agrawal, S.; Sønderby, C.; Kumar, M.; Heek, J.; Bromberg, C.; Gazen, C.; Carver, R.; Andrychowicz, M.; Hickey, J.; et al. 2022. Deep learning for twelve hour precipitation forecasts. *Nature Communications*, 13(1): 1–10.

Feingold, G.; Koren, I.; Wang, H.; Xue, H.; and Brewer, W. A. 2010. Precipitation-generated oscillations in open cellular cloud fields. *Nature*, 466(7308): 849–852.

Gao, Z.; Shi, X.; Han, B.; Wang, H.; Jin, X.; Maddix, D.; Zhu, Y.; Li, M.; and Wang, Y. B. 2023. Prediff: Precipitation nowcasting with latent diffusion models. *Advances in Neural Information Processing Systems*, 36: 78621–78656.

Gao, Z.; Shi, X.; Wang, H.; Zhu, Y.; Wang, Y. B.; Li, M.; and Yeung, D.-Y. 2022a. Earthformer: Exploring space-time transformers for earth system forecasting. *Advances in Neural Information Processing Systems*, 35: 25390–25403.

Gao, Z.; Tan, C.; Wu, L.; and Li, S. Z. 2022b. Simvp: Simpler yet better video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3170–3180.

Gong, J.; BAI, L.; Ye, P.; Xu, W.; Liu, N.; Dai, J.; Yang, X.; and Ouyang, W. 2024. CasCast: Skillful high-resolution precipitation nowcasting via cascaded modelling. In *Proceedings of the International Conference on Machine Learning*, 15809–15822.

Guen, V. L.; and Thome, N. 2020. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11474–11484.

Ha, J.-H.; and Lee, H. 2024. A deep learning model for precipitation nowcasting using multiple optical flow algorithms. *Weather and Forecasting*, 39(1): 41–53.

Larvor, G.; Berthomier, L.; Chabot, V.; Le Pape, B.; Pradel, B.; and Perez, L. 2020. MeteoNet, an open reference weather dataset by METEO FRANCE.

Leinonen, J.; Hamann, U.; Nerini, D.; Germann, U.; and Franch, G. 2023. Latent diffusion models for generative precipitation nowcasting with accurate uncertainty quantification. *arXiv preprint arXiv:2304.12891*.

Li, D.; Deng, K.; Zhang, D.; Liu, Y.; Leng, H.; Yin, F.; Ren, K.; and Song, J. 2023. LPT-QPN: A lightweight physics-informed transformer for quantitative precipitation nowcasting. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–19.

Lin, B.; Jin, Y.; Yan, W.; Ye, W.; Yuan, Y.; Zhang, S.; and Tan, R. T. 2024. NightRain: Nighttime video deraining via adaptive-rain-removal and adaptive-correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3378–3385.

Lin, K.; Zhang, B.; Yu, D.; Feng, W.; Chen, S.; Gao, F.; Li, X.; and Ye, Y. 2025. AlphaPre: Amplitude-phase disentanglement model for precipitation nowcasting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 17841–17850.

Markowski, P.; and Richardson, Y. 2011. *Mesoscale meteorology in midlatitudes*. John Wiley & Sons.

Microsoft. 2021. Neural Network Intelligence.

Ning, S.; Lan, M.; Li, Y.; Chen, C.; Chen, Q.; Chen, X.; Han, X.; and Cui, S. 2023. MIMO is all you need: A strong multi-in-multi-out baseline for video prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 1975–1983.

Othman, G.; and Zeebaree, D. Q. 2020. The applications of discrete wavelet transform in image processing: A review. *Journal of Soft Computing and Data Mining*, 1(2): 31–43.

Peebles, W.; and Xie, S. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4195–4205.

Qiu, C.-J.; Bao, J.-W.; and Xu, Q. 1993. Is the mass sink due to precipitation negligible? *Monthly weather review*, 121(3): 853–857.

Qiu, M.; Zhao, P.; Zhang, K.; Huang, J.; Shi, X.; Wang, X.; and Chu, W. 2017. A short-term rainfall prediction model using multi-task convolutional neural networks. In *IEEE International Conference on Data Mining*, 395–404. IEEE.

Ravuri, S.; Lenc, K.; Willson, M.; Kangin, D.; Lam, R.; Mirowski, P.; Fitzsimons, M.; Athanassiadou, M.; Kashem,

S.; Madge, S.; et al. 2021. Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878): 672–677.

She, L.; Zhang, C.; Man, X.; and Shao, J. 2024. LLMDiff: Diffusion model using frozen LLM Transformers for precipitation nowcasting. *Sensors*, 24(18): 6049.

Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Advances in Neural Information Processing Systems*, 28: 802–810.

Shi, X.; Gao, Z.; Lausen, L.; Wang, H.; Yeung, D.-Y.; Wong, W.-k.; and Woo, W.-c. 2017. Deep learning for precipitation nowcasting: A benchmark and a new model. *Advances in Neural Information Processing Systems*, 30.

Sura, P. 2011. A general perspective of extreme events in weather and climate. *Atmospheric Research*, 101(1-2): 1–21.

Veillette, M.; Samsi, S.; and Mattioli, C. 2020. Sevir: A storm event imagery dataset for deep learning applications in radar and satellite meteorology. *Advances in Neural Information Processing Systems*, 33: 22009–22019.

Voleti, V.; Jolicoeur-Martineau, A.; and Pal, C. 2022. Mcvd-masked conditional video diffusion for prediction, generation, and interpolation. *Advances in neural information processing systems*, 35: 23371–23385.

Wang, Y.; Zhang, F.-L.; and Dodgson, N. A. 2024. Scantd: 360° scanpath prediction based on time-series diffusion. In *Proceedings of the ACM International Conference on Multimedia*, 7764–7773.

Wang, Z.; Zhang, H.; and Bai, C. 2025. PiCNet: Physics-infused Convolution Network for Radar-Based Precipitation Nowcasting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1–5.

Wen, P.; Bai, L.; He, M.; Filippi, P.; Zhang, F.; Bishop, T. F.; Wang, Z.; and Hu, K. 2024. DuoCast: Duo-Probabilistic Meteorology-Aware Model for Extended Precipitation Nowcasting. *arXiv preprint arXiv:2412.01091*.

Wu, B.; Chen, W.; Wang, W.; Peng, B.; Sun, L.; and Chen, L. 2024a. WeatherGNN: Exploiting meteo-and spatial-dependencies for local numerical weather prediction bias-correction. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2433–2441.

Wu, H.; Liang, Y.; Xiong, W.; Zhou, Z.; Huang, W.; Wang, S.; and Wang, K. 2024b. Earthfarsser: Versatile spatio-temporal dynamical systems modeling in one model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 15906–15914.

Wu, H.; Xu, F.; Chen, C.; Hua, X.-S.; Luo, X.; and Wang, H. 2024c. Pastnet: Introducing physical inductive biases for spatio-temporal video prediction. In *Proceedings of the ACM International Conference on Multimedia*, 2917–2926.

Xing, L.; Wu, H.; Ma, Y.; Wang, J.; and Long, M. 2024. HelmFluid: Learning Helmholtz Dynamics for Interpretable Fluid Prediction. In *International Conference on Machine Learning*, 54673–54697.

Yan, C.-W.; Foo, S. Q.; Trinh, V. H.; Yeung, D.-Y.; Wong, K.-H.; and Wong, W.-K. 2024. Fourier amplitude and correlation loss: Beyond using l2 loss for skillful precipitation nowcasting. *Advances in Neural Information Processing Systems*, 37: 100007–100041.

Yin, J.; Meo, C.; Roy, A.; Cher, Z. B.; Lică, M.; Wang, Y.; Imhoff, R.; Uijlenhoet, R.; and Dauwels, J. 2024. Precipitation nowcasting using physics informed discriminator generative models. In *European Signal Processing Conference*, 967–971. IEEE.

Yu, D.; Feng, W.; Lin, K.; Li, X.; Ye, Y.; Luo, C.; and Du, W. 2025. Integrating multi-source data for long sequence precipitation forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 28539–28547.

Yu, D.; Li, X.; Ye, Y.; Zhang, B.; Luo, C.; Dai, K.; Wang, R.; and Chen, X. 2024. Diffcast: A unified framework via residual diffusion for precipitation nowcasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 27758–27767.

Zhang, G.; Zhu, Y.; Wang, H.; Chen, Y.; Wu, G.; and Wang, L. 2023a. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5682–5692.

Zhang, Y.; Long, M.; Chen, K.; Xing, L.; Jin, R.; Jordan, M. I.; and Wang, J. 2023b. Skilful nowcasting of extreme precipitation with NowcastNet. *Nature*, 619(7970): 526–532.

Zhong, X.; Chen, L.; Liu, J.; Lin, C.; Qi, Y.; and Li, H. 2024. FuXi-Extreme: Improving extreme rainfall and wind forecasts with diffusion model. *Science China Earth Sciences*, 1–13.

Zhong, Y.; Liang, L.; Zharkov, I.; and Neumann, U. 2023. Mmvp: Motion-matrix-based video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4273–4283.