Motion-Guided Global-Local Aggregation Transformer Network for Precipitation Nowcasting

Dong, Xichao; Zhao, Zewei; Wang, Yupei; Wang, Jianping; Hu, Cheng

DOI
10.1109/TGRS.2022.3217639

Publication date
2022

Document Version
Final published version

Published in
IEEE Transactions on Geoscience and Remote Sensing

Motion-Guided Global–Local Aggregation Transformer Network for Precipitation Nowcasting

Xichao Dong, *Member, IEEE*, Zewei Zhao, Yupei Wang, Jianping Wang, and Cheng Hu, *Senior Member, IEEE*

***Abstract*— Nowadays deep learning-based weather radar echo extrapolation methods have competently improved nowcasting quality. Current pure convolutional or convolutional recurrent neural network-based extrapolation pipelines inherently struggle in capturing both global and local spatiotemporal interactions simultaneously, thereby limiting nowcasting performances, e.g., they not only tend to underestimate heavy rainfalls' spatial coverage and intensity but also fail to precisely predict nonlinear motion patterns. Furthermore, the usually adopted pixel-wise objective functions lead to blurry predictions. To this end, we propose a novel motion-guided global–local aggregation Transformer network for effectively combining spatiotemporal cues at different time scales, thereby strengthening global–local spatiotemporal aggregation urgently required by the extrapolation task. First, we divide existing observations into both short- and long-term sequences to represent echo dynamics at different time scales. Then, to introduce reasonable motion guidance to Transformer, we customize an end-to-end module for jointly extracting motion representation of short- and long-term echo sequences (MRS, MRL), while estimating optical flow. Subsequently, based on Transformer architecture, MRS is used as queries to retrospect the most useful information from MRL for an effective aggregation of global long-term and local short-term cues. Finally, the fused feature is employed for future echo prediction. Additionally, for the blurry prediction problem, predictions from our model trained with an adversarial regularization achieve superior performances not only in nowcasting skill scores but also in precipitation details and image clarity over existing methods. Extensive experiments on two challenging radar echo datasets demonstrate the effectiveness of our proposed method.**

Manuscript received 19 January 2022; revised 22 August 2022; accepted 18 October 2022. Date of publication 26 October 2022; date of current version 11 November 2022. This work was supported in part by the Special Fund for Research on National Major Research Instruments (NSFC) under Grant 61827901 and Grant 31727901, in part by the National Natural Science Foundation of China under Grant 61960206009, in part by the Natural Science Foundation of Chongqing under Grant cstc2020jcyj-msxmX0621, and in part by the Distinguished Young Scholars of Chongqing under Grant cstc2020jcyj-jqX0008. *(Corresponding author: Yupei Wang.)*

Xichao Dong is with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China, also with the Key Laboratory of Electronic and Information Technology in Satellite Navigation, Ministry of Education, Beijing Institute of Technology, Beijing 100081, China, and also with the Chongqing Key Laboratory of Novel Civilian Radar, Beijing Institute of Technology Chongqing Innovation Center, Chongqing 401120, China.

Zewei Zhao, Yupei Wang, and Cheng Hu are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China, and also with the Key Laboratory of Electronic and Information Technology in Satellite Navigation, Ministry of Education, Beijing Institute of Technology, Beijing 100081, China (e-mail: wangyupei2019@outlook.com).

Jianping Wang is with the Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS), Delft University of Technology, 2628 CD Delft, The Netherlands.

Digital Object Identifier 10.1109/TGRS.2022.3217639

***Index Terms*— Attention mechanism, optical flow, precipitation nowcasting, transformers, weather radar.**

I. Introduction

PRECIPITATION nowcasting (PN) is of great value for reducing adverse effects of weather disasters on modern society. Weather radar provides high-quality data support for developing nowcasting algorithms. Based on recent past weather radar observations in local regions, weather radar echo extrapolation algorithms aim to predict near future radar echo sequences precisely and promptly.

Traditional methods [1], [2], [3], [4], [5], [6], [7] usually extrapolate linearly relying on the precalculated motion vectors. However, they either simplify intricacy strong radar echo as isolated storm cells [1], [2] or assume the echo intensity remains constant [3], [4], [5], [6], [7], failing in extrapolating scattered or split echo as well as predicting fine-grained precipitation pattern evolutions. Furthermore, traditional methods cannot benefit fully from the large amount of historical weather radar data.

Nowadays, the data-driven deep learning (DL)-based methods have shown remarkable potential [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24] for pn tasks. They are either based on pure convolutional structures or convolutional recurrent neural networks (ConvRNNs).

The convolutional structures [9], [10], [11] are with advantages of simple and easy to understand while inherently limited to the size of the reception field, hence unable to capture long-range spatiotemporal relationships.

Prevailing ConvRNN models [12], [13], [14], [15], [16], [17], [18], [19], [20], [25], [26] are generally dedicated to designing more satisfied ConvRNN units based on convolutional long short-term memory (ConvLSTM) network [12]. However, basic topologies of these recent developed units are too complex to be optimized easily and bring higher computation burden. What is more, common ConvRNNs are first-order Markovian models, i.e., they only use information from the previous time step to update the hidden state, resulting in inherent struggles in perceiving long-range spatiotemporal dependencies simultaneously. Furthermore, previous models only consider short-term input sequence with limited dynamics to encode spatiotemporal interactions.

In addition, for the choice of objective functions, current deep prediction models [12], [13], [14], [15], [16], [17]

usually minimize the mean squared error (MSE) or mean absolute error (MAE) between the ground-truth and the predicted images, which leads to conservative, blurry, and over-smoothing results lacking the key texture information of precipitation fields. However, since the nowcasting products are aimed for forecasters, the human perceptual quality and the clarity of predicted images are also important and the blurry results cannot provide effective references for the issuance of severe weather forecast warning.

In summary, previous DL-based nowcasting models struggle in 1) capturing both global and local spatiotemporal interactions simultaneously, limiting their performance and 2) predicting high fidelity echo image sequence with rich rainfall details. For example, they tend to underestimate heavy rainfalls' spatial coverages and intensities at longer nowcasting lead times, and show constrains in precisely nowcasting complex echo dynamics such as convective initialization, dissipation, and deformation.

More recently, the Transformer [27], [28] architectures' emergence provides an alternative to mitigate the limitations of previous pure convolutional or ConvRNN models applied for nowcasting tasks. The self-attention module in Transformer is dedicated to calculating the long-range spatial correlations among pixels and cross-attention module is proper for perceiving the temporal similarities among echo sequences with different temporal scales. Thus, the Transformer architecture essentially resonates with the goal of effective global–local spatiotemporal aggregation urgently required by pn.

Nevertheless, using existing Transformers directly for nowcasting precipitations has the following issues. On the one hand, the computational burden is sometimes unaffordable when we directly use the whole standard encoder–decoder vision Transformer [27] architecture, since the computational cost becomes quadratic to spatiotemporal dimensions [29]. On the other hand, when performing attention mechanism, the guidance of motion information is lacked in previous Transformer architecture. This leads to the model neglects some key information of similar and fine-grained echo patterns in the spatiotemporal neighborhood when fast echo motions such as convection generation present, thereby fails in capturing correct and accurate echo pattern evolutions.

To tackle aforementioned challenges, we propose a novel motion-guided global–local aggregation Transformer network for pn. Particularly, we divide existing observations into both short- and long-term sequences to represent echo dynamics at different time scales. Instead of directly using the whole encoder–decoder structure of Transformer architecture, we mainly leverage the decoder part for effectively and efficiently combining spatiotemporal cues at different time scales. What is more, to introduce reasonable motion guidance to Transformer, we customize an end-to-end module for jointly extracting motion representation of short- and long-term echo sequences abbreviated to MRS and MRL, while estimating optical flow. Then based on Transformer architecture, MRS is used to perform the self-attention in addition to the cross-attention operations to MRL, so as to retrospect the most useful information from MRL for an effective aggregation of global long-term and local short-term cues. Finally, the fused feature is employed for future echo prediction. For the blurry prediction problem, we further adopt an adversarial training strategy for improving predictions' perceptual quality and clarity.

The contributions of our work are summarized as follows.

1) We propose a motion-guided global–local aggregation Transformer network for pn. We divide existing observations into both short- and long-term sequences, and introduce the Transformer architecture for effective combination of spatiotemporal cues at different time scales, thereby strengthening global–local spatiotemporal aggregation required by the pn task.
2) We propose an end-to-end module to jointly obtain motion representation (MR) of echo sequences while estimating optical flow, thereby introducing reasonable motion guidance to the Transformer architecture.
3) We further train our proposed model with an adversarial strategy to tackle the blurry prediction problem. Experimental results demonstrate that our predictions achieve superior performances not only in nowcasting skill scores but also in precipitation details and image clarity over existing methods.

The rest of this article is organized as follows. Section II describes a review of the basics. Section III introduces the details of our proposed method. In Section IV, we report and analyze quantitative and qualitative experimental results. The discussion and conclusion are drawn in Section V.

II. Review of the Basics

A. Problem Definition and ConvRNN Structures

We formulate the pn task as follows. Given past weather radar echo observations $S_{1:t} = \{X_k | k = 1, 2, \ldots, t\} \in \mathbb{R}^{t \times H \times W \times C}$ as input (where $X_k \in \mathbb{R}^{H \times W \times C}$ represents the kth frame of $S_{1:t}$), our goal is to optimize the extrapolation model $\mathcal{F}$ for obtaining predicted sequence $\widetilde{S}_{(t+1):T}$ similar with the ground-truth future sequence $S_{(t+1):T}$. The echo images are always stored as grayscale images hence the channel C is 1 here.

In ConvLSTM [12] basic unit, full connections of the standard LSTMs [30] are replaced with convolutions to capture both the spatial and temporal information at the same time. Additionally, [13] builds an encoder-forecaster structure by stacking ConvLSTMs, as shown in Fig. 1. The cell states and hidden states are delivered horizontally along the temporal dimension. In addition, the hidden states are transferred vertically to handle spatial appearances. The down sampling and up sampling blocks are inserted in between two ConvLSTM layers.

Follow-up ConvRNN methods [14], [15], [16], [17], [18], [19], [20] are generally dedicated to designing more completed ConvRNN units based on ConvLSTMs, e.g., PredRNN [14] uses pairwise memory cells to extend ConvLSTM, memory in memory (MIM) [16] adopts additional memory cells for capturing both nonstationary and stationary processes better. MotionRNN [17] is improved from MIM and it decomposes motions into transient variations and motion trends. Interaction dual attention long short-term memory (IDA-LSTM) [18]

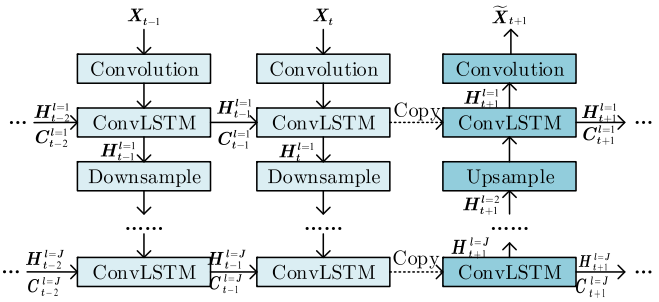


Fig. 1. Encoder-forecaster architecture based on ConvLSTMs. The superscript l denotes the lth ConvLSTM layer in spatial dimensions and J denotes the total ConvLSTM layers. The subscript represents the time step.

leverages attention mechanism to reweight the fused hidden state and cell state features in channel and spatial dimension. They have achieved promising performances.

B. End-to-End Optical Flow Estimation

Optical flow encodes motion information between echo frames and traditional optical flow-based pn methods [4], [5], [6], [7] still provide important references for the issuance of severe weather forecast warnings. However, the bottlenecks of cumbersome calculation of optical flow and using optical flow to extrapolating linearly hinder further applications of optical flows in pn task.

For the first bottleneck, encouraged by the success of DL-based optical flow estimation [31], [32] methods and 3-D convolutions for video optical flow estimation [33], [34], we customize an end-to-end module for optical flow estimation but emphasize on improving pn qualities.

For the second bottleneck, instead of using optical flow for linear echo extrapolation, our model jointly obtains MR of echo sequences while estimating optical flow, thereby introducing reasonable motion guidance. Obtaining high-level MR of the echo sequence in latent space enhances the robustness of our model, and further helps model nowcast fine-grained echo pattern evolutions precisely.

C. Brief Review of Transformers

The Transformer architecture [27] is originally proposed for natural language processing tasks. Recently many variant Transformer structures have been adopted for computer vision tasks [27], [28], [29], [35] and achieved impressive results. The prominent performance of Transformers in these tasks has fascinated researchers to explore their applications in remote sensing fields, including hyperspectral image classification [36], remote sensing image change detection [37], remote sensing image captioning [38], and so on.

The self-attention and cross-attention mechanisms are key components of Transformers. The intuition of the attention mechanisms in Transformer is that each token can interact with others and exploit rich semantic information more efficiently, which makes Transformers suitable for preforming long-range interactions [27].

D. Blurry Prediction Problem in pn

Despite the promising performances in improving nowcasting skill scores, the DL-based pn models tend to produce blurry predictions, which is a common problem of ill repute. This could be explained by the analysis of the adopted objective functions, i.e., current models [12], [13], [14], [15], [16], [17], [18], [19], [20] usually minimize the MSE or MAE between the ground truth and the predictions. However, the widely used MSE estimator tends to return the average of many possible solutions and the MAE estimator tends to return the median of the set of equally like values [39], which leads to conservative, blurry, and over smoothing results lacking the key texture information of precipitation fields.

Some recent studies [40], [41], [42], [43], [44] try adding different generative adversarial (GAN) losses to tackle this problem. However, instead of specially designing new network architectures for the underestimation problem of rainfall regions and intensities, these methods mainly add adversarial training strategies based on the existing extrapolation frameworks and explore the performances. For example, the generator in [42] is based on the spatiotemporal long short-term memory (ST-LSTM) [14], the generator in [43] is based on a simple 3D-convolutional neural network (CNN), in [44] the authors mainly adopt the network architecture from [45] for radar extrapolation, and in [41] the authors explore two classic DL-based radar extrapolation models' (U-Net [8] and ConvLSTM [12]) performance when combined with GAN losses. This adoption of existing architectures limits further improvements.

III. Our Proposed Method

As shown in Fig. 2, our model has two branches, i.e., we adopt the encoder-forecaster structure as our basic echo prediction branch (hereinafter referred to as the prediction branch), and we further propose the global–local aggregation branch (hereinafter referred to as the aggregation branch) which contains of the optical flow guided MR module (hereinafter referred to as the motion module), Transformer decoder (TD), and a channel attention-based fusion module.

We first divide existing observations into both short- and long-term sequences, mainly considering that storms usually have their own life cycles [46]. For example, the single-cell storms are usually small scale and fairly disorganized convective elements which generate or dissipate rapidly, having 2 h or less life cycle [46]. For the future 1 h nowcasting with the past 1 h observations as input, the past 1-hour short-term sequence hardly capture the whole evolution cycle of these convective elements. However, with a longer term sequence as reference, we have the possibility to accurately describe its motion. For the multicell storms which are maintained in an organized linear pattern, they tend to persist longer and evolve less rapidly. In this situation, the short- and long-term sequence split strategy is also better for tracking this slower movement trend because the motion trend of the short-term sequence is reflected in the long-term sequence.

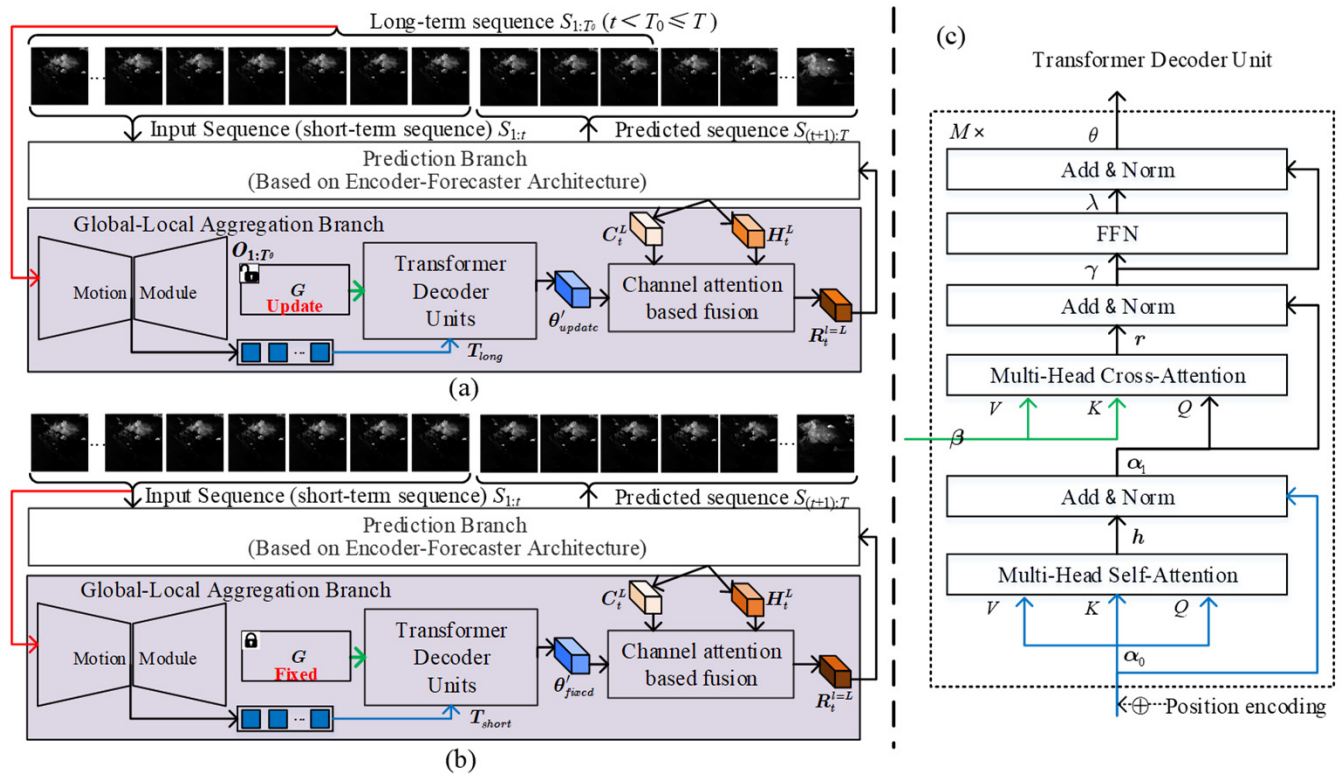


Fig. 2. Overall framework of our proposed network for pn. (a) Data flow of the "update" training stage. (b) Data flow of the "fixed" training stage. (c) Structure of the TD unit.

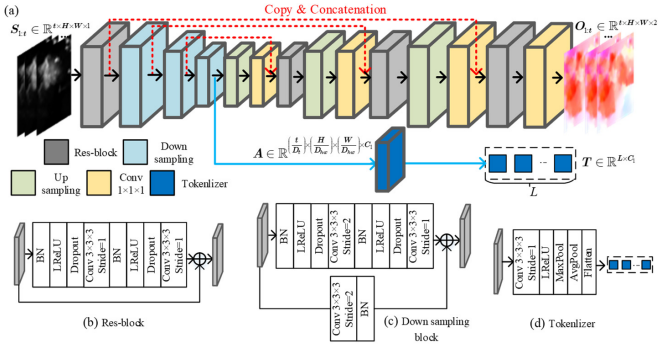


Fig. 3. (a) Overall structure of our proposed optical flow guided MR module. (b)–(d) Demonstrates the detail architectures of our used Res-block, down sampling block, and the tokenizer, respectively. BN represents Batch Normalization layer. LReLU denotes Leaky-ReLU activation function.

A. Our Motion Module

Taken $S_{1:t}$ as input, our motion module outputs the corresponding optical flows $O_t = \{o_k | k = 1, 2, \ldots, t\} \in \mathbb{R}^{t\times H\times W\times 2}$, where $o_k \in \mathbb{R}^{H\times W\times 2}$ is the 2-D optical flow vector for the kth and $(k+1)$th input frames.

As shown in Fig. 3, our designed motion module is an encoder–decoder architecture based on 3-D residual blocks [47] (Res-blocks). Particularly, the encoder contains of one Res-block and three down-sampling blocks, the decoder has three Res-blocks, three up-sampling blocks, and four $1 \times 1 \times 1$ convolutional layers. To mitigate the information lost by down sampling in the encoding stage, we also adopt the skip connections [8], i.e., the red dashed lines in Fig. 3. This is helpful for deep network training [48], [49], and enables the decoder to obtain more high-resolution information so as to restore better detailed information.

Note that o_t is the optical flow for the tth and $(t+1)$th input frames while we do not give the $(t+1)$th frame to the model. We predict this last echo image's optical flow mainly considering the following two benefits: 1) the model requires semantic inference to predict the future optical flow, and this may force the model to exploit better motion cues for pn task [34] and 2) because the output dimensions of the deconvolution layers are usually a multiple of the input, it is actually easier to implement a model with same input and output dimensions. With optical flow as supervision, our module learns MR without relying on explicit optical flow computation.

For the Res-blocks, as illustrated in Fig. 3(b), we adopt the preactivation mechanism [47] to construct the identify mapping, i.e., the input feature map passes through the normalization layer, the activation layer, and the regularization layer before passing through the $3 \times 3 \times 3$ convolutional layer. Instead of using max pooling layers for down sampling, we adopt stride convolutions to preserve spatiotemporal details. Details of our used down sampling blocks are shown in Fig. 3(c). Note that our used down sampling blocks can be seen as special forms of the Res-blocks where we add one $3 \times 3 \times 3$ convolutional layer with a normalization layer in the identity skip connection stream, and for convolutions in the

down sampling block, the strides are set to 2. The up-sampling blocks are implemented by 3-D deconvolution layers.

After $S_{1:t}$ is passed through the encoder of our motion module and with the guidance of optical flow motion information, our intuition is that here the high-level feature maps $A \in \mathbb{R}^{(t/D_t)\times(H/D_{\text{hw}})\times(W/D_{\text{hw}})\times C_1}$ could describe the motion information of the input sequence in an abstract way, where D_t and D_{hw} represent the down sampling factor in temporal and spatial dimensions, respectively, and C_1 is the channel dimension of A.

Additionally, to generate vector sequence as token embedding which meets the input form requirements of the Transformer module and further aggregate spatiotemporal information, we forward A to a tokenizer to obtain motion tokens T. In specific, we map A into an embedding through one $1 \times 1 \times 1$ convolutional layer with an activation layer. After that, we use a max-polling layer to further aggregate spatial information. Additionally, an adaptive pooling layer is employed for aggregating temporal information. Here we get the intermediate feature map $A_1 \in \mathbb{R}^{(H/D'_{\text{hw}})\times(W/D'_{\text{hw}})\times C_1}$, and finally we flat A_1 in raster-scan order to obtain $T \in \mathbb{R}^{L\times C_1}$, where $L = HW/(D'_{\text{hw}})^2$ and $D'_{\text{hw}} = k_s k_t D_{\text{hw}}$ is another down sampling factor similar with D_{hw}, k_s and k_t are the aggregating factors in spatial dimension of the max pooling layer and temporal dimension of the adaptive pooling layer, respectively. The detail structure of our tokenizer is shown in Fig. 3(d).

B. TDs for Global–Local Aggregation

1) TD Structure: As shown in Fig. 2(c), the TD has two groups of inputs, denoted as output tokens $\alpha \in \mathbb{R}^{m\times C}$ and input tokens $\beta \in \mathbb{R}^{n\times C}$.

First, to retain positional information, the positional encoding $\epsilon_{\text{pos}} \in \mathbb{R}^{m\times C}$ is added to α to generate $\alpha_0 = \alpha + \epsilon_{\text{pos}}$. Taken α_0 as input, the multihead self-attention (MSA) block with N_{head} heads is expressed as

$$Q_i = \alpha_0 W_q^i, \quad K_i = \alpha_0 W_k^i, \quad V_i = \alpha_0 W_v^i \tag{1}$$

$$h_i = \text{Attn}[Q_i; K_i; V_i] = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i \tag{2}$$

$$h = \text{Concat}[h_1, \ldots, h_i, \ldots, h_{N_{\text{head}}}] W_d \tag{3}$$

where $Q_i, K_i, V_i \in \mathbb{R}^{m\times(C/N_{\text{head}})}$ are the query, key, and value vectors, respectively. $W_q^i, W_k^i, W_v^i \in \mathbb{R}^{C\times(C/N_{\text{head}})}$ and $W_d \in \mathbb{R}^{C\times C}$ are linear projection layers. h_i denotes the output from the ith head. Concat$[\cdot]$ denotes the concatenation operation. After that, with a layer normalization layer and an identity shortcut, we obtain $\alpha_1 \in \mathbb{R}^{m\times C}$, written as

$$\alpha_1 = \text{LayerNorm}(h + \alpha_0). \tag{4}$$

Subsequently, in the multihead cross-attention (MCA) block, the query vector is projected linearly from α_1, and the key and value vectors are projected linearly from the input embedding β. The output from MCA $r \in \mathbb{R}^{m\times C}$ is formed as follows:

$$r = \text{Concat}[r_1, \ldots, r_j, \ldots, r_{N_{\text{head}}}] W'_d \tag{5}$$

where

$$r_j = \text{Attn}\left[\alpha_1 W_q^j; \beta W_k^j; \beta W_v^j\right] \tag{6}$$

is the output from the jth head, W'_d, W_q^j, W_k^j, and W_v^j represent linear projections.

Then, to provide nonlinearity, the feedforward network (FFN) layer is adopted, which is consist of two linear projection layers with a rectified linear unit (ReLU) activation in between. Formally, the output of FFN $\lambda \in \mathbb{R}^{m\times C}$ is calculated as

$$\lambda = \text{FFN}(\gamma) = \text{ReLU}(\gamma W_1) W_2 \in \mathbb{R}^{m\times C} \tag{7}$$

where $W_1 \in \mathbb{R}^{C\times C_{\text{in}}}$ and $W_2 \in \mathbb{R}^{C_{\text{in}}\times C}$ are linear projections, C_{in} is the dimension of the FFN inter layer.

Finally, after another layer normalization layer and an identity shortcut, we obtain the TD output $\theta \in \mathbb{R}^{m\times C}$ as follows:

$$\theta = \text{LayerNorm}(\lambda + \gamma). \tag{8}$$

In short, the operations in a TD unit are summarized as

$$\theta = \text{TD}(\alpha, \beta) \in \mathbb{R}^{m\times C}. \tag{9}$$

2) "Update" Training Stage Using TDs: To make comprehensive use of echo motion characteristics at different time scales, we divide the existing observations into both long- and short-term histories (denoted as $S_{\text{long}} = S_{1:T_0}(t < T_0 \leq T)$, $S_{\text{short}} = S_{1:t}$), and leverage TDs to perceive the global and local correlations. Note that for a spatiotemporal prediction task like pn, the precondition is that, we have no future observation information as input at model inference stage. Therefore, the divided S_{long} and S_{short} are conceptions only for the training data, and we still only use as the model input during the model inference stage. To this end, we adopt an "update/fixed" strategy to train the proposed model inspired by memory networks [50], [51].

Concretely, in this stage, we use both S_{long} and S_{short} as input of our model. As shown in Fig. 2(a), S_{long} is forwarded to the aggregation branch and S_{short} is forwarded to our prediction branch. Under the guidance of optical flows, the long-term sequence is encoded by our motion module to obtain the corresponding MRL $T_{\text{long}} \in \mathbb{R}^{L\times C_1}$. Then we use the TDs to update the latent long-term motion information into an external memory bank $G \in \mathbb{R}^{M\times C_1}$, where L and M are the number of tokens of T_{long} and G, respectively. In specific, T_{long} and G are adopted as the two sets of inputs of TD unit to obtain corresponding motion prototype θ_{update}, as defined in (9)

$$\theta_{\text{update}} = \text{TD}(T_{\text{long}}, G) \in \mathbb{R}^{L\times C_1} \tag{10}$$

where the subscript represents the feature in "update" stage.

Subsequently, we reshape $\theta_{\text{update}} \in \mathbb{R}^{L\times C_1}$ to obtain the motion context representation $\theta'_{\text{update}} \in \mathbb{R}^{(H/D'_{\text{hw}})\times(W/D'_{\text{hw}})\times C_1}$ for further fusion with inadequate-spatiotemporal-representation features output from the prediction branch.

In the "update" training stage, we first initialize the weights of the memory bank G with normal distribution. Then, during training, the weights in G are updated continuously with the backpropagation algorithm [50], [51], i.e., once there are updates of G's weights, it could be seen as that the MRL is stored to G iteratively.

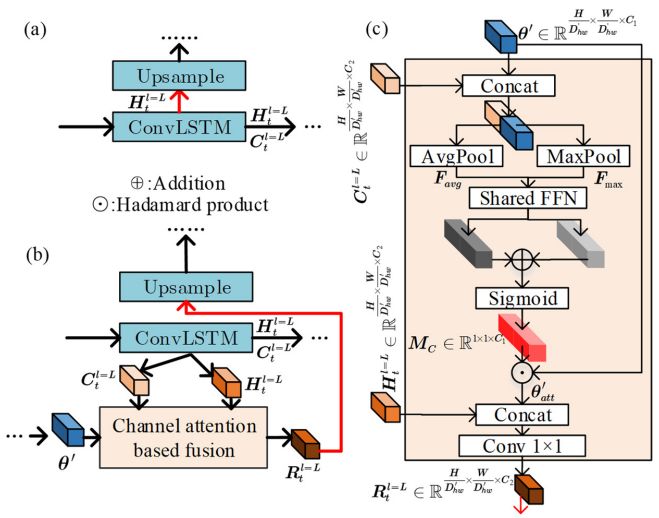


Fig. 4. (a) Data flow in traditional encoder-forecaster structure. (b) Data flow between encoder-forecaster and our aggregation branch. (c) Detail structure of our used channel attention fusion submodule.

3) Channel Attention-Based Feature Fusion: Since ConvLSTMs are the basic components of our prediction branch, and the cell state C_t in ConvLSTM stores the long short-term information iterations from the history to the current, we choose C_t to refine θ'_{update} for inserting the necessary motion context at present time step t. This refinement is based on a channel-attention submodule [52]. Let $C_t^L, H_t^L \in \mathbb{R}^{(H/D'_{\text{hw}})\times(W/D'_{\text{hw}})\times C_2}$ represents the cell state and hidden state of the Lth ConvLSTM layer (i.e., the deepest layer) in our prediction branch, where C_2 is the number of channels. As shown in Fig. 4(c), the spatial information of C_t^L and θ'_{update} are first aggregated using both max-pooling and average-pooling operations, obtaining two sets of spatial context embedding F_{max} and F_{avg}, respectively. Then F_{max} and F_{avg} are forwarded to a shared FFN layer, added together and passed through a Sigmoid activation layer to generating the channel attention map $M_C \in \mathbb{R}^{1\times1\times C_1}$. Subsequently, Hadamard product is performed between θ'_{update} and M_C to obtain θ'_{att}. In addition, θ'_{att} is concatenated with H_t^L and forwarded to a 1×1 convolutional layer to obtain $R_t^{l=L}$. Finally, $R_t^{l=L}$ is fed to the corresponding up sampling layer in the prediction branch for subsequent operations, and predict corresponding next echo image $\widetilde{X}_{t+1}$ as illustrated in Fig. 4(b).

4) "Fixed" Training Stage Using TDs: The "fixed" training stage is designed to allow the most proper MRL in the memory bank G to be retrieved by the dynamic-limited MRS extracted from S_{short}. In this stage, only S_{short} is forwarded to the aggregation and prediction branch. Similar to the "update" training process, S_{short} is encoded by our motion module to obtain MRS (i.e., $T_{\text{short}} \in \mathbb{R}^{L\times C_1}$) under the guidance of optical flows. Note that the motion module used here has the same structure with that used in the "update" training process while does not share weights for unique and diverse expression of MRL and MRS characteristics.

Subsequently, T_{short} and G are adopted as two sets of inputs of TD unit to obtain corresponding motion prototype θ_{fixed}

$$\theta_{\text{fixed}} = \text{TD}(T_{\text{short}}, G) \in \mathbb{R}^{L\times C_1} \tag{11}$$

where the subscript "fixed" represents the feature map in the "fixed" training stage. Additionally, we also reshape $\theta_{\text{fixed}} \in \mathbb{R}^{L\times C_1}$ to obtain the motion context representation $\theta'_{\text{fixed}} \in \mathbb{R}^{(H/D'_{\text{hw}})\times(W/D'_{\text{hw}})\times C_1}$. This procedure is the same as the "update" training stage (10). However, we lock the gradient of G, and the weights of G are no longer updated and optimized but remain fixed for retrieving the motion context in MRL during this "fixed" training stage. That is, the entire model parameters are trained to nowcast the long-term echo sequence, except for the memory bank G. Actually, the TD units can be seen as a unique soft addressing paradigm [53], [54] here, i.e., given the query vectors, the attention scores are calculated from the similarity between key-value pairs and appended to the value vectors in a weighted manner, instead of strictly meeting the condition that the key vectors are equal to query vectors to retrieve the corresponding stored values, which is adopted in the hard addressing process. Based on the MSA and MCA mechanism of TD, it enables the most proper MRS extracted by our motion module to access the most useful MRL, realizing effective aggregation of global long-term and local short-term motion cues. In addition, as the same with the "update" training stage, there is also a channel attention-based fusion between the prediction and aggregation branch to generate the predicted image $\widetilde{X}_{t+1}$.

In short, the "update" and "fixed" training stages are performed alternately to predict future echo sequences. The short-term echo sequence $S_{1:t}$ is forwarded to the prediction branch in both the "update" and "fixed" training stage. The long- and short-term sequences $S_{1:T_0}$ and $S_{1:t}$ are forwarded to our aggregation branch alternatively during the two training stages.

C. Loss Functions

Our adopted loss functions include the pixel-wise loss, the optical flow loss, the GAN loss, and the feature-wise style loss. The pixel-wise loss $\mathcal{L}_{\text{pred}}$ in both the "update" and "fixed" training phases are the MAE and MSE error, denoted as

$$\mathcal{L}_{\text{pred}} = \frac{1}{(T-t)HW}\sum_{i=t+1}^{T}\sum_{j}(|\widetilde{s}_{i,j} - s_{i,j}| + (\widetilde{s}_{i,j} - s_{i,j})^2) \tag{12}$$

where $\widetilde{s}_{i,j}$ and $s_{i,j}$ are the ith grid points of the jth timestamp in the predicted sequence $\widetilde{S}_{(t+1):T}$ and ground-truth $S_{(t+1):T}$.

The optical flow loss $\mathcal{L}_{\text{flow}}$ is based on the endpoint error, i.e., the sum of L2 distance between the estimated optical flows and ground-truth optical flows, denoted as

$$\mathcal{L}_{\text{flow}} = \sum_{k=1}^{K}\sum_{p} \parallel \widetilde{o}_{k,p} - o_{k,p} \parallel_2 \tag{13}$$

where K is the number of predicted 2-D optical flow vectors. $\widetilde{o}_{k,p}$ is the estimated 2-D optical flow vector of the kth and $(k+1)$th image at pixel p. $o_{k,p}$ is the corresponding

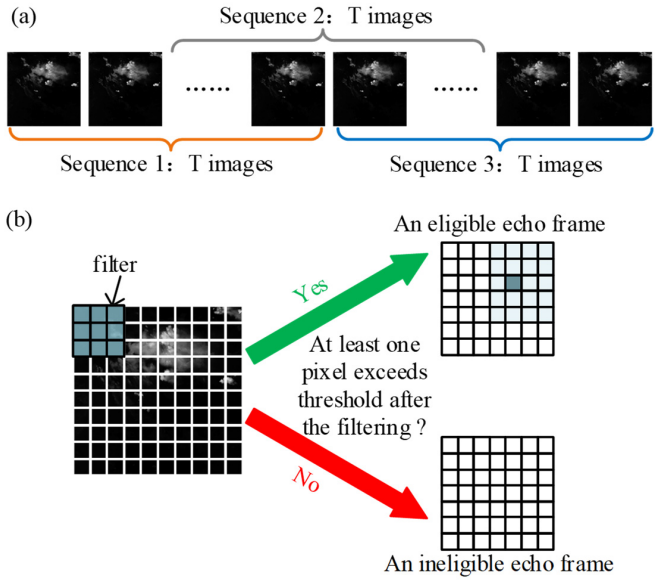


Fig. 5. (a) Split each event into three echo sequences. (b) Determine whether an echo image is eligible or not by average filtering.

ground-truth optical flow vector. The subscript "2" represents the L2-norm. We use the recent EpicFlow [55] method to calculate the pseudo-ground-truth optical flows.

What is more, for the blurry prediction problem, we explore the conditional GAN loss [56] and the style loss [57] to further improve the fidelity and perceptual quality of our predictions, written as

$$\mathcal{L}_{\text{GAN}} = \min_{G}\max_{D}\left\{\begin{array}{l}\mathbb{E}[\log(D(S_{(t+1):T}, \widetilde{S}_{(t+1):T}))] \\ +\mathbb{E}[\log(1 - D(S_{(t+1):T}, \widetilde{S}_{(t+1):T}))]\end{array}\right\} \tag{14}$$

$$\mathcal{L}_{\text{style}} = \sum_{i=1}^{T-t}\sum_{j=1}^{N_\Psi}\left\|\text{Gram}_{i,j}^{\Psi}(S_i) - \text{Gram}_{i,j}^{\Psi}(\widetilde{S}_i)\right\|_F \tag{15}$$

where D is the convolutional discriminator [45]. Ψ is the pretrained Visual Geometry Group (VGG)-19 network. N_Ψ is the number of adopted layers for feature extraction in VGG-19 network. Subscript "F" in (15) represents the Frobenius norm of the matrix. $\text{Gram}_{i,j}^{\Psi}$ is the feature map's Gram matrix [58], which is formed as

$$\text{Gram}_{i,j}^{\Psi}(S_i) = \Psi_j(S_i)\cdot\Psi_j(S_i)^T / C_{i,j}^{\Psi} H_{i,j}^{\Psi} W_{i,j}^{\Psi} \tag{16}$$

where $\Psi_j(S_i) \in \mathbb{R}^{C_{i,j}^{\Psi}\times(H_{i,j}^{\Psi}W_{i,j}^{\Psi})}$ is the reshaped output feature map from the jth VGG-19 layer, with the ith frame S_i as input, and $\Psi_j(S_i)^T$ is the corresponding transpose matrix.

The total loss function $\mathcal{L}_{\text{tot}}$ is finally formulated as

$$\mathcal{L}_{\text{tot}} = \mathcal{L}_{\text{pred}} + \mu_1\cdot\mathcal{L}_{\text{flow}} + \mu_2\cdot\mathcal{L}_{\text{GAN}} + \mu_3\cdot\mathcal{L}_{\text{style}} \tag{17}$$

where μ_1, μ_2, μ_3 are individual loss weights for $\mathcal{L}_{\text{flow}}$, $\mathcal{L}_{\text{GAN}}$, and $\mathcal{L}_{\text{style}}$, respectively.

IV. Experiments and Analyses

A. Dataset and Preprocessing

Two challenging real-world weather radar datasets (Storm EVent ImageRy (SEVIR) [59] and standardized radar dataset (SRAD2018) [60]) are adopted to evaluate our model.

1) SEVIR Dataset: The SEVIR dataset [59] is mainly sampled from storm events such as heavy rainfalls over the US and it contains spatiotemporally aligned image sequences from the geostationary environmental satellite system (GOES) and the next-generation radar (NEXRAD).

Each sequence, or "event" in SEVIR contains a 384 × 384 km region with a spatial resolution of 1 × 1 km and spanning a 4 h of time which is sampled in 5 min steps. Note that we only use NEXRAD derived vertically integrated liquid (VIL) data and do not use other kinds of data such as data from GOES. The VIL images in SEVIR are stored as integers in the range of 0–255. The converting rule between these encoded integers and the true VIL data with units of kg/m^2 is as follows [59]:

$$\text{VIL} = \begin{cases}0, & \text{if } p \le 5 \\ (p-2)/90.66, & \text{if } 5 < p \le 18 \\ \exp[(p-83.9)/38.9], & \text{if } p > 18\end{cases} \tag{18}$$

where p is the integers stored in the images.

Our data preprocessing process mainly contain two steps, i.e., event splitting and echo images filtering. For the event splitting, we concretize the pn task into predicting echo sequence of the future one hour based on previous one-hour observations, i.e., for SEVIR dataset, given the previous 12 echo images, we aim to predict the future 12 echo images. Therefore, as shown in Fig. 5(a), we split each SEVIR event into three input-output subsequences, and each of them has $T = 24$ images spanning 2 h of time.

For the echo images filtering, we use an average filtering strategy [24] to determine whether each image is a rainy image or not, and further determine whether a subsequence is kept or not. Concretely, as shown in Fig. 5(b), a filter was convolved with each image in the subsequences to detect areas of high rainfall intensities and the subsequence is kept if more than half of the images in the subsequence contain at least one pixel exceeding a predefined threshold. The filter size is 1/8th of the image size and the threshold for SEVIR VIL dataset is set to 0.7 kg/m^2 which correspond to a reflectivity value of 30 dBZ.

The sequences are divided into train dataset, validation dataset, and test dataset and finally we obtain 14 154 training sequences, 3654 validation sequences, and 4304 test sequences for SEVIR dataset. The data distribution is shown in Fig. 6.

2) SRAD2018 Dataset: The SRAD2018 dataset is from Tianchi IEEE International Conference on Data Mining (ICDM) 2018 Global Artificial intelligence (AI) Challenge on Meteorology, collected by Shenzhen Meteorological Bureau and Hong Kong Observatory. Each sequence or "event" in SRAD2018 originally contains a 501 × 501 km region with a spatial resolution of 1 × 1 km and spanning a 6 h of time which is sampled in 6 min steps, and is taken from an altitude of 3 km. The reflectivity values are directly stored in images

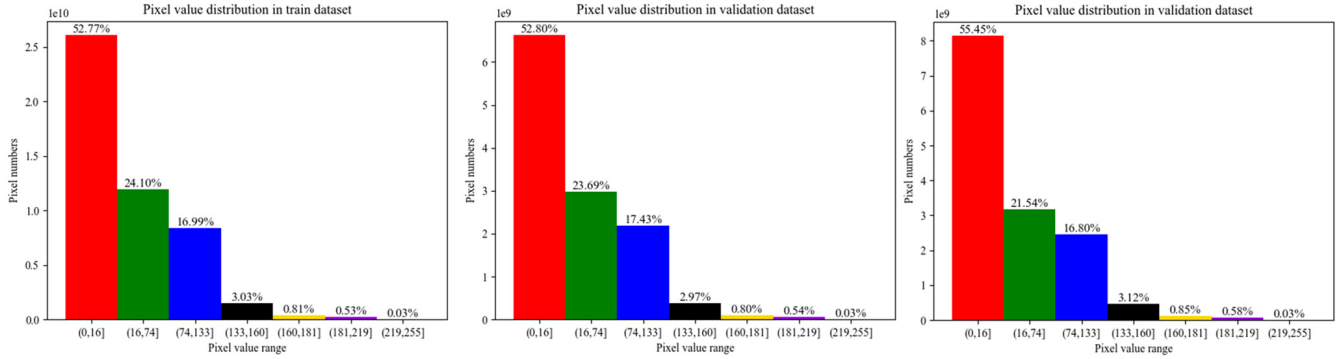


Fig. 6. Histograms of the pixel values in SEVIR dataset.

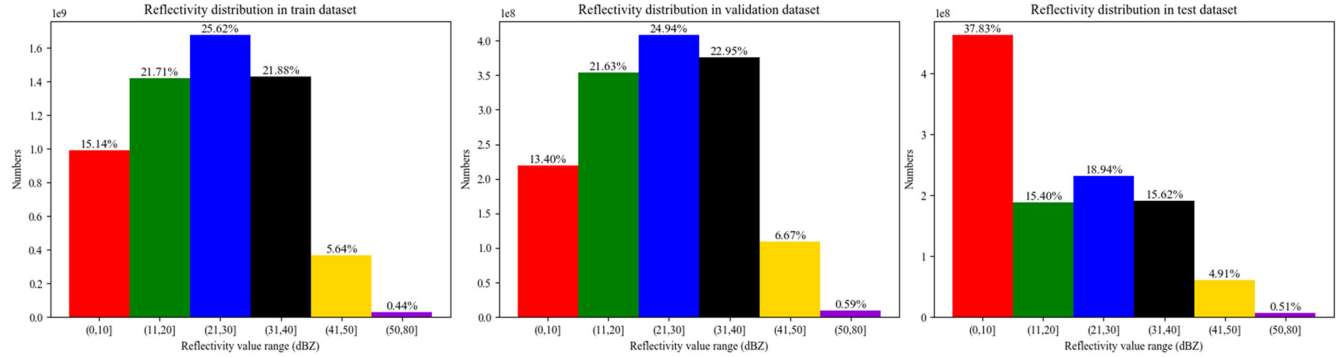


Fig. 7. Histograms of the pixel values in SRAD2018 dataset.

in SRAD2018. Considering the limited computing resources to process image sequence with such spatial sizes, we crop the central part of images in SRAD2018 to get echo images with the size of 384 × 384, which is the same with the echo image size in SEVIR.

We also preprocess SRAD2018 dataset by event splitting and echo images filtering. Since images in SRAD2018 are sampled in 6 min' time steps, we split each SRAD2018 event into three input-output subsequences, and each of them has $T = 20$ images spanning a 2 h of time. For the echo images filtering of SRAD2018, the filter size is also 1/8th of the image size and the threshold is set to 40 dBZ. Moreover, a nonlinear scaling operation [18] is performed on SRAD2018 dataset to make converting between the reflectivity values and integers as follows:

$$\text{dBZ} = p \times 95/255 - 10 \tag{19}$$

where p is the integers stored in the images after the conversion.

We divide the sequences into train dataset, validation dataset, and test dataset and finally we obtain 7258 training sequences, 1802 validation sequences, and 1200 test sequences for SRAD2018 dataset. The dataset distribution is shown in Fig. 7.

B. Experimental setups

1) Evaluation Protocols: We evaluate the models with nowcasting skill scores, pixel-wise image evaluation indexes, and perceptual image evaluation indexes.

First, for the nowcasting skill scores, we qualify the model performance using the widely adopted metrics in the area of pn, including the probability of detection (POD), critical success index (CSI), false alarm ratio (FAR), and the Heidke skill score (HSS) [61], [62]. Higher POD, CSI, and HSS values and lower FAR values indicate better nowcasting performance of the model. Note that the POD metric is biased to overestimating the size of precipitation areas while the FAR metric does the opposite [63]. The HSS and CSI metrics take into account of both the false alarm rate and probabilities of detection, hence HSS, CSI are mainly adopted for judging model performances while POD, FAR are mainly for analyzing why the HSS and CSI of a model are better or worse than another.

Above metrics are actually based on the binary classification, therefore the reflectivity (or VIL) threshold must be clarified for any metrics. As shown in Figs. 5 and 6, there are not enough samples satisfying the threshold of 50 dBZ and 12.0 kg/m^2 in the SRAD2018 and SEVIR dataset, respectively, hence for the SRAD2018 dataset, our adopted thresholds are 10, 20, 30, and 40 dBZ. For SEVIR dataset, our adopted thresholds for VIL are 0.140 kg/m^2, 0.700 kg/m^2, 3.50 kg/m^2, and 6.90 kg/m^2 [64]. We can easily modify these thresholds according to users' requirements in applications.

Additionally, we adopt peak signal to noise ratio (PSNR), MSE, and structural similarity (SSIM) [65] metrics to measure the pixel-level performances. What is more, the perceptual evaluation metrics including learned perceptual image patch similarity (LPIPS) [66] and Frechet inception distance (FID) [67] are also considered. LPIPS is a perceptual metric which indicates the perceptual similarity between two images ranging from 0 to 1, and it is considered to be similar with human

TABLE I

PERFORMANCE COMPARISONS OF DIFFERENT DL-BASED MODELS FOR PN TASK UNDER METRICS OF MSE, PSNR, SSIM, LPIPS, AND FID. ↑ DENOTES THE HIGHER THE BETTER, AND ↓ DENOTES THE LOWER THE BETTER. THE BEST PERFORMANCE UNDER SPECIFIC SETTINGS IS MARKED WITH **BOLD RED**. THE SECOND-BEST PERFORMANCE IS MARKED WITH **BOLD BLUE** (SAME BELLOW)

Models	SEVIR					SRAD2018				
	MSE↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	MSE↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
ConvLSTM	870.751	23.603	0.815	0.365	133.3	2254.65	19.114	0.760	0.455	107.1
PredRNN	834.326	23.626	0.789	0.403	169.2	2127.90	19.348	0.781	0.401	132.1
Conv-TT-LSTM	823.948	23.668	0.816	0.365	156.1	2128.29	19.278	0.779	0.431	141.6
IDA-LSTM	**817.982**	**23.807**	**0.820**	0.377	151.1	2010.91	19.548	**0.786**	0.409	135.7
U-Net	883.037	23.475	0.805	0.342	102.6	**1974.57**	**19.722**	0.781	0.418	110.9
rainymotion	1873.79	20.291	0.750	**0.189**	**54.69**	2700.46	18.381	0.750	**0.226**	**30.19**
ours w/o GAN loss	**784.802**	**24.187**	**0.827**	0.369	125.3	**1969.65**	**19.772**	**0.791**	0.372	116.6
ours w/ GAN loss	1218.42	22.232	0.755	**0.211**	**34.04**	3075.72	17.748	0.733	**0.292**	**58.34**

TABLE II

PERFORMANCE COMPARISONS OF DIFFERENT DL-BASED MODELS FOR PN TASK UNDER POD AND CSI METRICS ON SEVIR DATASET

Models	POD↑				CSI↑			
	$\geq 0.14\ kg/m^2$	$\geq 0.70\ kg/m^2$	$\geq 3.50\ kg/m^2$	$\geq 6.90\ kg/m^2$	$\geq 0.14\ kg/m^2$	$\geq 0.70\ kg/m^2$	$\geq 3.50\ kg/m^2$	$\geq 6.90\ kg/m^2$
ConvLSTM	0.8190	0.6951	0.3503	0.2125	0.7606	0.6397	0.3103	0.1928
PredRNN	0.8746	0.7597	0.3696	0.2280	0.7629	0.6622	0.3124	0.1934
Conv-TT-LSTM	0.8656	0.7601	0.3693	0.2025	0.7705	0.6633	0.3136	0.1801
IDA-LSTM	0.8617	0.7407	0.3706	0.2048	0.7732	0.6605	0.3175	0.1841
U-Net	0.8711	0.7364	0.2757	0.1373	0.7585	0.6466	0.2479	0.1277
rainymotion	**0.8782**	**0.8273**	**0.6293**	**0.4669**	0.7276	0.5908	0.2559	0.1357
ours w/o GAN loss	**0.8901**	**0.7931**	**0.4143**	**0.2541**	**0.7803**	**0.6817**	**0.3515**	**0.2268**
ours w/ GAN loss	0.8717	0.7656	0.4125	0.2752	**0.7781**	**0.6733**	**0.3602**	**0.2409**

TABLE III

PERFORMANCE COMPARISONS OF DIFFERENT DL-BASED MODELS FOR PN TASK UNDER HSS AND FAR METRICS ON SEVIR DATASET

Models	HSS↑				FAR↓			
	$\geq 0.14\ kg/m^2$	$\geq 0.70\ kg/m^2$	$\geq 3.50\ kg/m^2$	$\geq 6.90\ kg/m^2$	$\geq 0.14\ kg/m^2$	$\geq 0.70\ kg/m^2$	$\geq 3.50\ kg/m^2$	$\geq 6.90\ kg/m^2$
ConvLSTM	0.8105	0.7443	0.4452	0.2961	**0.0888**	**0.1175**	**0.2877**	**0.2999**
PredRNN	0.8080	0.7615	0.4493	0.2996	0.1456	0.1672	0.3271	0.3616
Conv-TT-LSTM	0.8163	0.7629	0.4528	0.2848	0.1267	0.1645	0.3167	0.3138
IDA-LSTM	0.8188	0.7608	0.4558	0.2892	**0.1197**	**0.1460**	**0.3145**	**0.3098**
U-Net	0.8035	0.7484	0.3720	0.2063	0.1483	0.1649	0.3214	0.4379
rainymotion	0.7701	0.6892	0.3737	0.2205	0.1951	0.3322	0.7053	0.8446
Ours w/o GAN loss	**0.8223**	**0.7761**	**0.4939**	**0.3457**	0.1394	0.1767	0.3266	0.3435
Ours w/ GAN loss	**0.8222**	**0.7700**	**0.5009**	**0.3640**	0.1243	0.1580	0.3185	0.3505

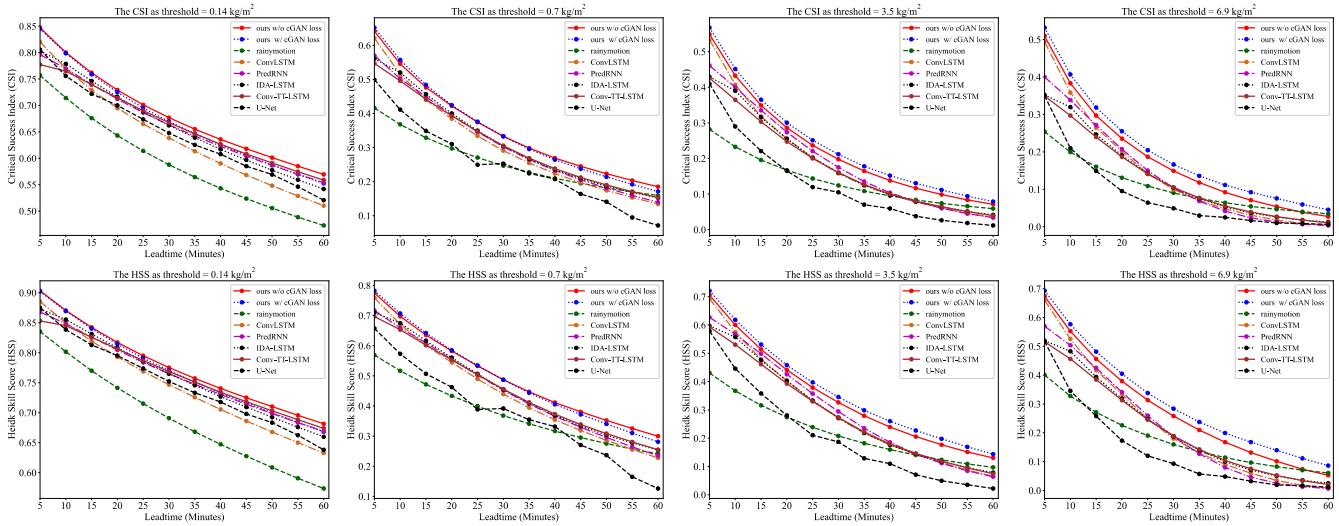


Fig. 8. CSI and HSS scores against different nowcasting lead times under different VIL thresholds on SEVIR dataset. Results in the upper row show CSI scores. Results in the lower row show HSS scores.

objects' recognition system. The FID indicates the generated images clarity.

2) Model Configurations: For the prediction branch, we adopt an encoder-forecaster structure with four ConvLSTM

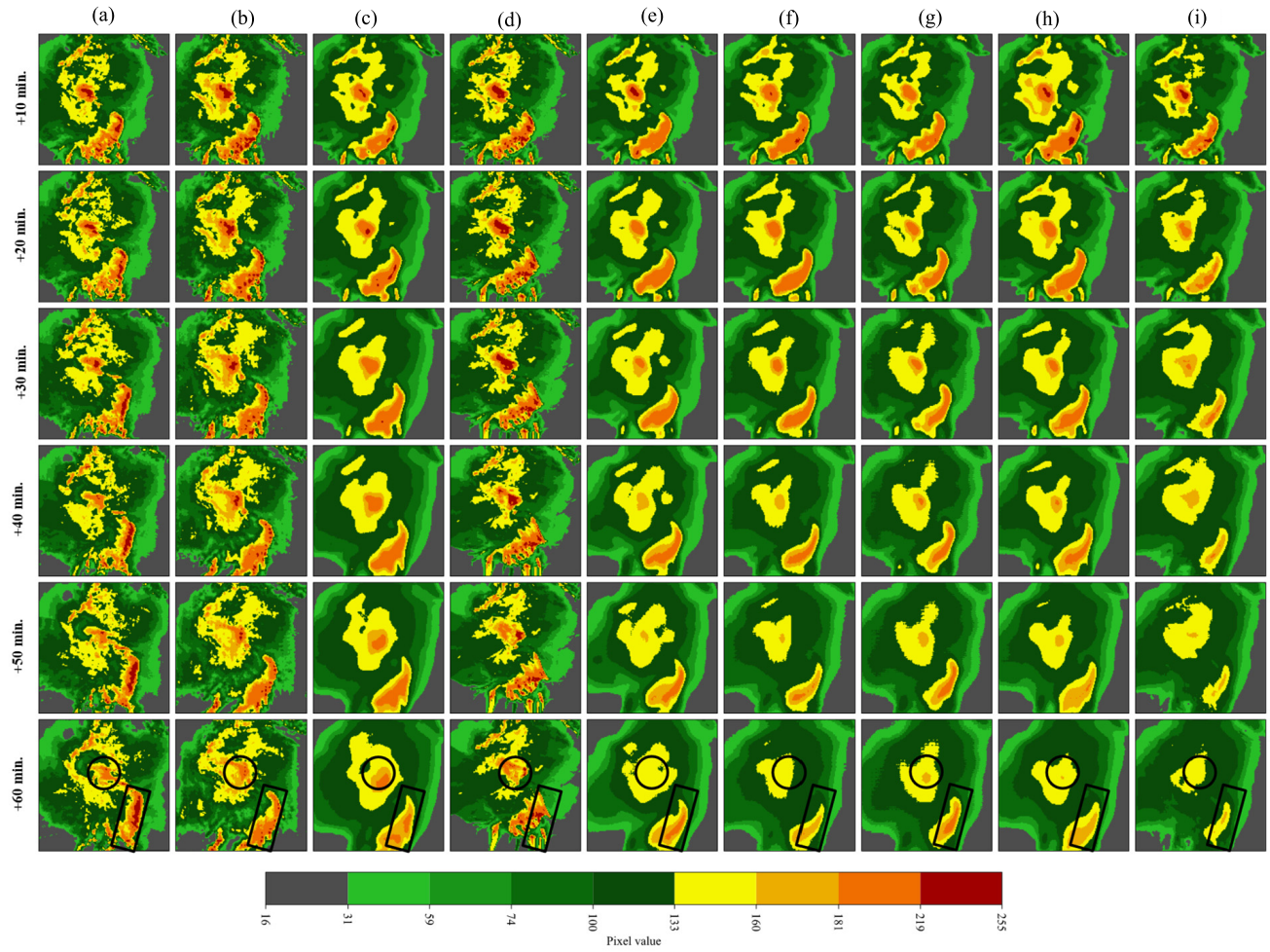


Fig. 9. Visualizations of a convective organization process in SEVIR dataset. (a) Observations. (b) Our proposed model trained with GAN strategy. (c) Our proposed model trained without GAN strategy. (d) rainymotion. (e) ConvLSTM. (f) PredRNN. (g) IDA-LSTM. (h) Conv-TT-LSTM. (i) U-Net.

TABLE IV
PERFORMANCE COMPARISONS OF DIFFERENT DL-BASED MODELS FOR PN TASK UNDER POD AND CSI METRICS ON SRAD2018 DATASET

Models	POD↑				CSI↑			
	$\geq 10dBZ$	$\geq 20dBZ$	$\geq 30dBZ$	$\geq 40dBZ$	$\geq 10dBZ$	$\geq 20dBZ$	$\geq 30dBZ$	$\geq 40dBZ$
ConvLSTM	0.7538	0.7122	0.6470	0.5241	0.6507	0.6161	0.5542	0.4423
PredRNN	0.7495	0.7178	0.6583	**0.5502**	0.6417	0.6079	0.5471	0.4427
Conv-TT-LSTM	0.7484	0.7171	0.6488	0.5172	0.6383	0.6051	0.5403	0.4241
IDA-LSTM	0.7577	**0.7259**	**0.6611**	0.5409	0.6527	0.6190	0.5563	0.4451
U-Net	**0.7619**	0.7249	0.6551	0.5076	0.6530	0.6178	0.5529	0.4285
rainymotion	**0.7652**	**0.7453**	**0.7051**	**0.6261**	0.6324	0.5999	0.5380	0.4376
Ours w/o GAN loss	0.7608	0.7251	0.6598	0.5467	**0.6589**	**0.6245**	**0.5611**	**0.4540**
Ours w/ GAN loss	0.7584	0.7173	0.6439	0.5228	**0.6586**	**0.6221**	**0.5580**	**0.4485**

layers (i.e., $J = 4$ in Fig. 1), with the number of hidden states for the RNNs setting to 16, 64, 128, and 128. For the motion module, the output channel of the first Res-block is set to 32. Then, each time the feature maps are passed through the down sampling block, their channel dimension is doubled. The rest three up sampling blocks keep the channel dimension unchanged. The first three $1 \times 1 \times 1$ convolutional layers are used to adjust channel dimensions for the skip connections. The channel dimension of the last $1 \times 1 \times 1$ convolutional layer is set to 2 for the 2-D optical flow vector estimation. For TDs, the channel dimensions of the input and output tokens are set to 128. We set the length of long-term sequence as the same with the total length of subsequences in training data.

TABLE V

PERFORMANCE COMPARISONS OF DIFFERENT DL-BASED MODELS FOR PN TASK UNDER HSS AND FAR METRICS ON SRAD2018 DATASET

Models	HSS↑				FAR↓			
	$\geq 10dBZ$	$\geq 20dBZ$	$\geq 30dBZ$	$\geq 40dBZ$	$\geq 10dBZ$	$\geq 20dBZ$	$\geq 30dBZ$	$\geq 40dBZ$
ConvLSTM	0.7418	0.7206	0.6761	0.5802	0.1816	**0.1893**	**0.2178**	**0.2797**
PredRNN	0.7344	0.7135	0.6695	0.5811	0.1896	0.2089	0.2457	0.3202
Conv-TT-LSTM	0.7317	0.7113	0.6640	0.5640	0.1931	0.2122	0.2453	0.3111
IDA-LSTM	0.7446	0.7238	0.6787	0.5845	**0.1808**	0.1991	0.2304	0.2972
U-Net	0.7441	0.7221	0.6749	0.5673	0.1860	0.2011	0.2306	**0.2816**
rainymotion	0.7227	0.7020	0.6551	0.5678	0.2240	0.2556	0.3184	0.4246
Ours w/o GAN loss	**0.7493**	**0.7278**	**0.6816**	**0.5907**	**0.1759**	0.1902	0.2215	0.2874
Ours w/ GAN loss	**0.7477**	**0.7273**	**0.6798**	**0.5894**	0.1823	**0.1900**	**0.2193**	0.2864

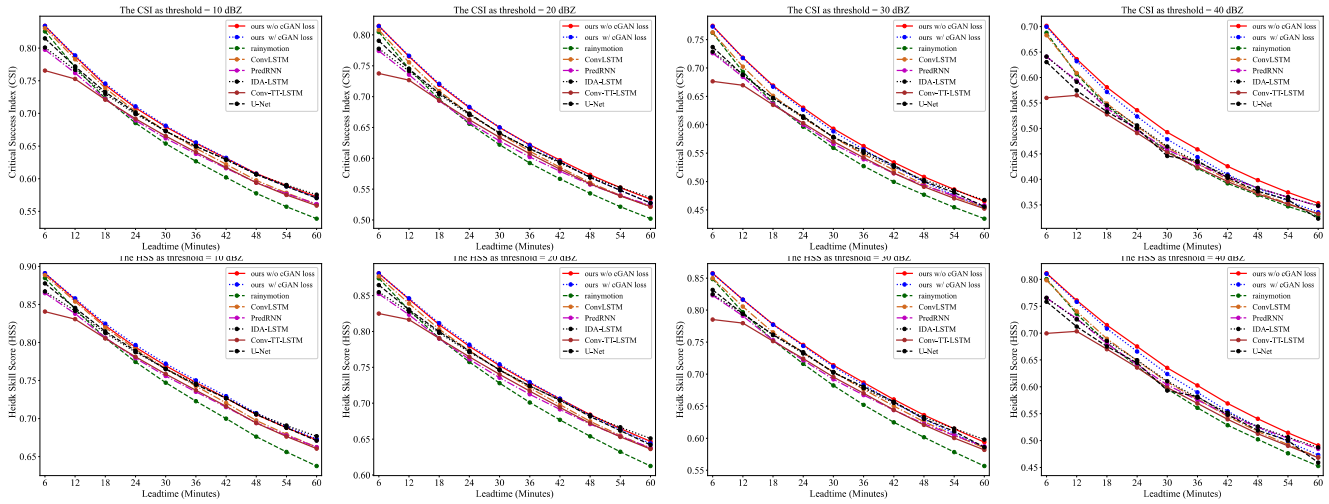


Fig. 10. CSI and HSS scores against different nowcasting lead times under different VIL thresholds on SRAD dataset. Results in the upper row show CSI scores. Results in the lower row show HSS scores.

Our adopted discriminator consists of six convolutional layers, of which the kernel size of the first five layers is set to 4, the stride is set to 2, and each of them is with a ReLU layer behind. The kernel size of the last convolution layer is set to 1 and the stride is set to 1. For the first layer, the output channel dimension is set to 64 and then after each convolutional layer, the number of channels is doubled.

Four other recent competitive ConvRNN-based models including ConvLSTM [12], PredRNN [14], convolutional tensor-train LSTM (Conv-TT-LSTM) [26], and IDA-LSTM [18] and one convolutional model U-Net [8] are implemented to compare performances. We also use an optical flow-based method (rainymotion) [7] for comparison.

The models are implemented by PyTorch framework on a server which is equipped with three NVIDIA TITAN RTX graphics cards. The Adam optimizer is adopted. Batch-size is set to 4 and learning rate is set to 0.0001. Except for U-Net, we train each of the other models for 10 000 iterations, since we observe obvious nonconvergence after 10 000 iterations on SRAD2018 dataset when training U-Net. Therefore, another 10 000 iterations are added when we train U-Net on SRAD2018 dataset.

Note that due to the adoption of different datasets in which the dataset size and data distribution are significantly various, and the limited computing resources, when implementing reference methods, sometimes we cannot keep the parameters consistent with that given in the literature and put the same effort to optimize all these models. Hence, there is still the possibility of performance biases and these biases are hard to be eliminated completely.

C. Quantitative and Qualitative Results

1) Results on SEVIR Dataset: We first analyze quantitative results. Table I lists performance comparisons under MSE, SSIM, PSNR, LPIPS, and FID metrics. Our model trained without GAN strategy achieves the best result for the pixel-wise metrics. When trained with the GAN strategy, our model is no longer optimal among pixel-wise indicators; however, it performs either the best or the second best in perceptual evaluation indicators including LPIPS and FID. The perceptual quality of predictions is significantly improved, which is comparable to the optical flow method.

Tables II and III list nowcasting skill score performance comparisons under POD, CSI, HSS and FAR metrics. Our proposed model when trained without GAN strategy outperforms all other reference methods in terms of HSS and CSI scores, and it maintains the FAR scores at a relatively low level at the same time. The HSS and CSI scores at

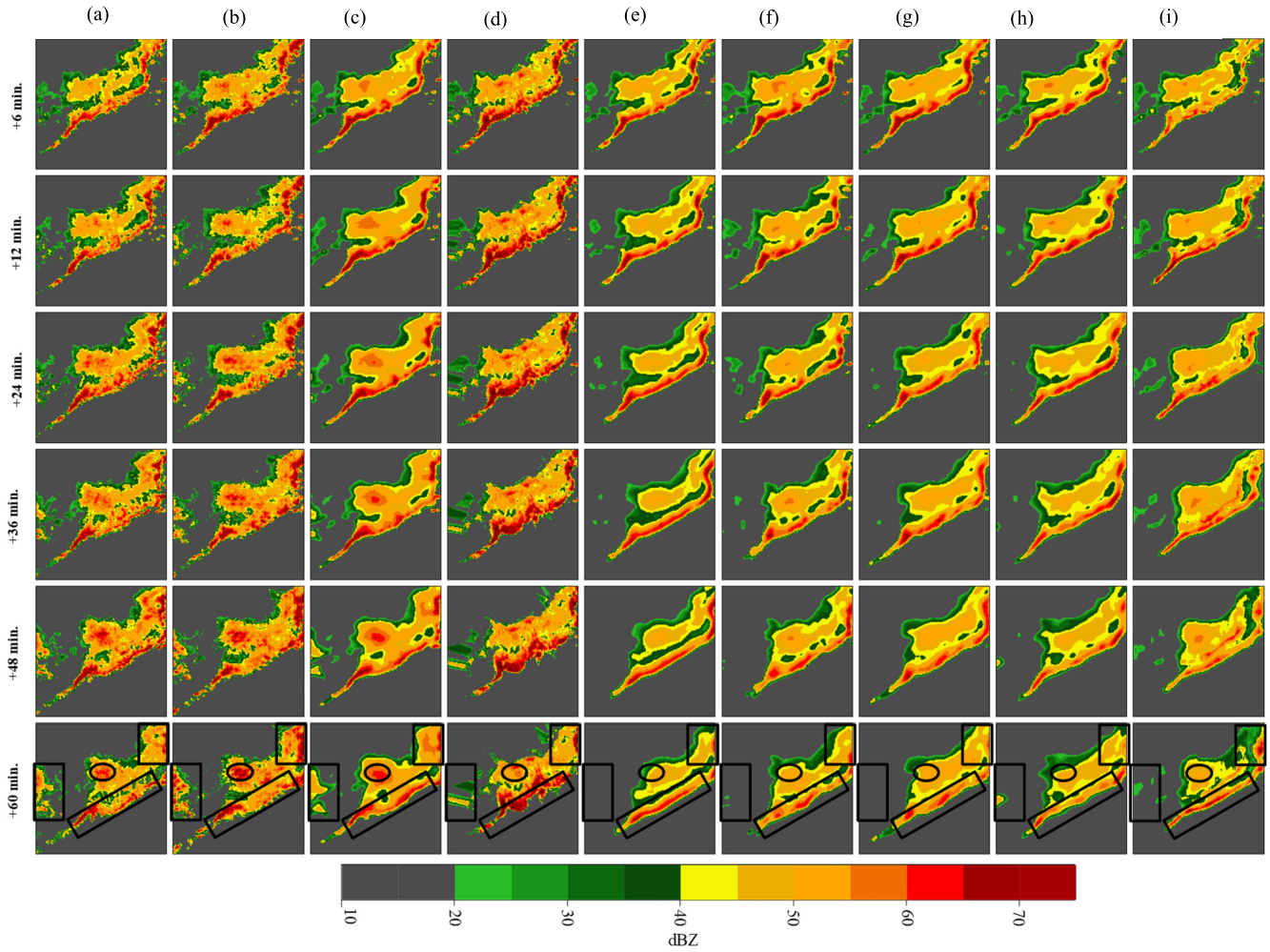


Fig. 11. Visualizations of a convective organization process in SRAD dataset. (a) Observations. (b) Our proposed model trained with GAN strategy. (c) Our proposed model trained without GAN strategy. (d) rainymotion. (e) ConvLSTM. (f) PredRNN. (g) IDA-LSTM. (h) Conv-TT-LSTM. (i) U-Net.

3.5 and 6.9 kg/m^2 thresholds are significantly better than other methods. These results imply that our model is more advanced in nowcasting heavy rainfalls, which is challenging for other models. When trained with the GAN strategy, there is a slight drop in HSS and CSI indicator except for the 3.5 and 6.9 kg/m^2 thresholds. We think this performance fluctuation when introducing the GAN losses is acceptable since the image clarity is ameliorated, and the predictions provides more precipitation details.

Fig. 8 shows CSI and HSS scores against different nowcasting lead times over different thresholds. From Fig. 8, we intuitively see that the U-Net performs the worst among all thresholds and during all nowcasting lead times. For other ConvRNN-based models, ConvLSTM get better nowcasting results than PredRNN, IDA-LSTM, and Conv-TT-LSTM in the first 12 min' lead time period. However, the performance of ConvLSTM drops sharply after 12 min. Performances of PredRNN, IDA-LSTM, and Conv-TT-LSTM on SEVIR dataset are similar and PredRNN gets slightly higher HSS and CSI scores than IDA-LSTM and Conv-TT-LSTM during 6 to 48 min' lead time. However, the results of all these models are inferior to those of our proposed method.

For qualitative analyses, we visualize a convective organization process selected from SEVIR dataset. As shown in Fig. 9, we focus on the area enclosed by the black circles and black rectangle. These areas are with strong rainfall intensities even exceeding a value of 32 kg/m^2 during the convective organization process. The rainfall field during this procedure has complex motion patterns. Our proposed model alleviates the problem of underestimating the intensity and spatial area of heavy rainfall, no matter with or without the introduction of GAN losses, while other models tend to underestimate the strong rainfall. In addition, the image clarity is improved significantly when trained with GAN loss, and there are more precipitation details compared with other blurry predictions. The optical flow method also predicts clear images; however, the distortions appear and worsen as nowcasting lead time increases. What is more, although the perceptual quantity is improved with the introduction of GAN loss, we could tell from Fig. 9 that there are still inaccurate nowcasting regions. This indicates that the image perceptual quality such as the

clarity is not always positively correlated with nowcasting skill scores.

2) Results on SRAD2018 Dataset: As shown in Tables IV and V, performances of our proposed model using HSS, CSI metrics under all the adopted thresholds are superior to those of other models. The optical flow method is characterized by high POD values while high FAR values at the same time, resulting in low HSS and CSI values. When trained with the GAN strategy, there is a slight performance drop in both HSS and CSI.

Fig. 10 shows the CSI and HSS scores against different nowcasting lead times under different reflectivity thresholds on SRAD2018 dataset. For the results under 40 dBZ threshold, ConvLSTM shows good scores before a nowcasting lead time of 18 min but its performance drops rapidly after 18 min. Conv-TT-LSTM does not obtain competitive results in the first 12 min but achieves similar performances with PredRNN and IDA-LSTM during 12 to 60 min, under both 30 and 40 dBZ thresholds. Different with results on SEVIR dataset, the U-Net model even gets competitive nowcasting scores with other ConvRNN-based models. Our model does not show significant improvements under the 30 dBZ threshold, where IDA-LSTM achieves the best scores at a nowcasting lead time of 1 h. However, when we raise the threshold to 40 dBZ, HSS and CSI scores of our model is obviously superior to others. The optical flow method still has good POD scores while poor FAR scores, as a result, there HSS and CSI scores are the worst at long nowcasting lead times.

Fig. 11 shows a nowcasting result comparison of a developing squall line system. We mainly focus on the four enclosed areas with solid black lines. The oblique black rectangle encloses the main structure of the squall line. All of the models nowcast the general structure of the squall line to a certain extent, but the prediction results of our models (trained with or without GAN loss) are more consistent with the real observations in terms of echo intensity and echo pattern evolutions. The left vertical black rectangle encloses an area of convection generation, and only our models successfully nowcast this convection initiation compared with other methods. The area enclosed by the horizontal black ellipse indicates the convective core strengthen, and only our models capture and predict this trend of echo intensity increasing precisely. This case study shows our model is more sensitive to complex echo motions such as convection generation and deformation, especially in strong rainfall cases, no matter with or without the introduction of GAN loss. It is interesting that although the optical flow method seems to predict sharp and clear images, however, the quantitative metrics are not so satisfied. Importantly, the predictions from the addition of GAN loss contains more echo structures although there are not obvious improvements in the nowcasting skill scores.

3) Effects of the Optical Flow Guidance: Fig. 12 shows an example of the learned optical flows and corresponding ground-truth optical flows calculated from EpicFlow. From Fig. 12, we know that the learned optical flow is consistent with the ground truth in the overall motion trend. This ensures a reasonable echo sequence MR learning. To further evaluate the importance of the optical flow guidance, we make the following ablation studies. In the aggregation branch of our model, we only adopt the encoder part of the motion module but without the decoder part and the supervision of the ground-truth optical flow. As listed in Table VI, for both datasets, after we remove the optical flow as motion guidance, the performances under heavy rainfall thresholds have different degrees of decline. This indicates the effectiveness of adopting a clear and reasonable motion guidance for improving nowcasting quality.

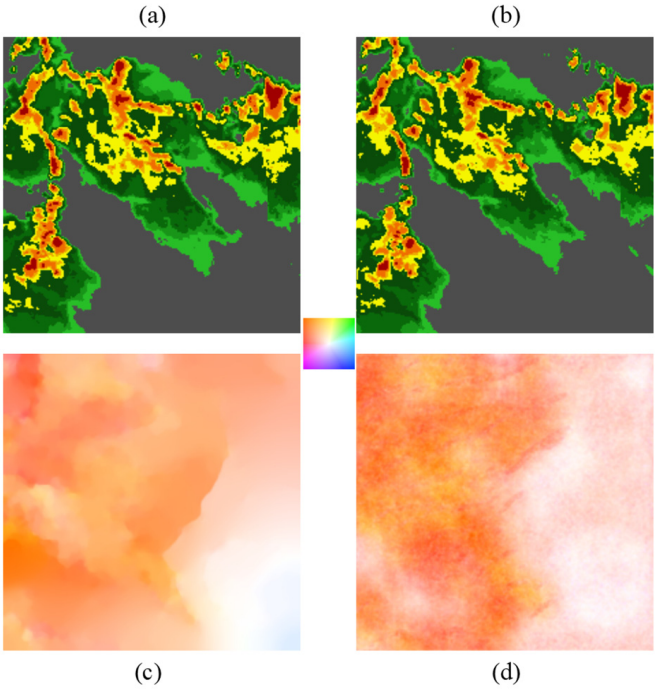


Fig. 12. (a) and (b) Two consecutive echo frames from SEVIR dataset. (c) Ground-truth optical flow calculated from Epicflow. (d) Learned optical flow by our motion module. In the middle, we show the color wheel. The different colors indicate the motion directions, and the color intensity represents the displacement's magnitude.

TABLE VI
EFFECTS OF ADOPTING OPTICAL FLOW AS MOTION GUIDANCE ON PERFORMANCE OF NOWCASTING QUALITIES

Models	SEVIR ($\geq 6.9kg/m^2$)				SRAD2018 ($\geq 40dBZ$)			
	HSS ↑	CSI ↑	POD ↑	FAR ↓	HSS ↑	CSI ↑	POD ↑	FAR ↓
Our model w/o optical flow guidance	0.3389	0.2171	0.3652	0.6757	0.5724	0.4439	0.5214	0.3462
Our model w/ optical flow guidance	0.3457	0.2268	0.2541	0.3435	0.5907	0.4540	0.5467	0.2874

V. DISCUSSION AND CONCLUSION

In this article, we proposed a motion-guided global–local aggregation Transformer network for improving pn quality. Different with previous convolutional structures or ConvRNN models for pn, on the one hand, we innovatively explore the Transformer architecture for an effective and efficient combination of spatiotemporal cues at different time scales, thereby further enhancing the global–local aggregation which is desperately required by pn task.

On the other hand, we notice that previous Transformer architecture lacks the guidance of motion information when performing attention calculations. To introduce reasonable motion guidance, we customize an end-to-end learning module for jointly extracting MR of echo sequences while estimating optical flow. This has the following benefits. First, an end-to-end learning manner avoids nontrivial computation burden of calculating optical flow. What is more, using optical flow as priori motion guidance forces our model learning latent MRs which are proper for nowcasting, and further benefits our model nowcasting fine-grained echo pattern evolutions precisely. Additionally, we do not use optical flow to extrapolate linearly. This enhances the nowcasting robustness of our model.

Furthermore, for the blurry prediction problem, we introduce the GAN training strategy to the proposed model. The experimental results show that the introduction of GAN loss help improve the predictions' perceptual quality and image clarity notably while the nowcasting skill scores are slightly unstable and maybe with acceptable performance drop.

For the evaluation metrics, many existing methods only adopts the pixel-wise indicators and the nowcasting skill scores for performance evaluation. As our experimental show, the blurry predictions sometimes have close HSS and CSI scores and better pixel-wise evaluation scores, compared with the clear results. However, the clear predictions which have rich echo structure details provide better reference for refined and accurate nowcasting. Hence, we propose to use nowcasting skill scores as the main evaluation metrics while use pixel-wise evaluation indicators (such as MSE) as optional insignificant references. What is more, when the skill score indicators of different models are almost equivalent, it is recommended to use perceptual indicators such as the clarity indicator FID for further judgment.

REFERENCES

[1] J. Johnson et al., "The storm cell identification and tracking algorithm: An enhanced WSR-88D algorithm," *Weather Forecast.*, vol. 13, no. 2, pp. 263–276, 1998.

[2] M. Dixon and G. Wiener, "TITAN: Thunderstorm identification, tracking, analysis, and nowcasting—A radar-based methodology," *J. Atmos. Ocean. Technol.*, vol. 10, no. 6, pp. 785–797, Dec. 1993.

[3] R. Rinehart and E. Garvey, "Three-dimensional storm motion detection by conventional weather radar," *Nature*, vol. 273, no. 5660, pp. 287–289, 1978.

[4] H. Sakaino, "Spatio–temporal image pattern prediction method based on a physical model with time-varying optical flow," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 3023–3036, May 2013.

[5] N. E. Bowler, C. E. Pierce, and A. Seed, "Development of a precipitation nowcasting algorithm based upon optical flow techniques," *J. Hydrol.*, vol. 288, nos. 1–2, pp. 74–91, Mar. 2004.

[6] W.-C. Woo and W.-K. Wong, "Operational application of optical flow techniques to radar-based rainfall nowcasting," *Atmosphere*, vol. 8, no. 12, p. 48, Feb. 2017.

[7] G. Ayzel, M. Heistermann, and T. Winterrath, "Optical flow models as an open benchmark for radar-based precipitation nowcasting (rainymotion v0. 1)," *Geosci. Model Develop.*, vol. 12, no. 4, pp. 1387–1402, 2019.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[9] K. Trebing, T. Stańczyk, and S. Mehrkanoon, "SmaAt-UNet: Precipitation nowcasting using a small attention-UNet architecture," *Pattern Recognit. Lett.*, vol. 145, pp. 178–186, May 2021.

[10] X. Pan, Y. Lu, K. Zhao, H. Huang, M. Wang, and H. Chen, "Improving nowcasting of convective development by incorporating polarimetric radar variables into a deep-learning model," *Geophys. Res. Lett.*, vol. 48, no. 21, Nov. 2021, Art. no. e2021GL095302.

[11] G. Ayzel, M. Heistermann, A. Sorokin, O. Nikitin, and O. Lukyanova, "All convolutional neural networks for radar-based precipitation nowcasting," *Proc. Comput. Sci.*, vol. 150, pp. 186–192, Jan. 2019.

[12] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 802–810.

[13] X. Shi et al., "Deep learning for precipitation nowcasting: A benchmark and a new model," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5622–5632.

[14] Y. Wang, M. Long, J. Wang, Z. Gao, and P. S. Yu, "PredRNN: Recurrent neural networks for predictive learning using spatiotemporal LSTMs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 879–888.

[15] Y. Wang, Z. Gao, M. Long, J. Wang, and S. Y. Philip, "PredRNN++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning," in *Proc. Int. Conf. Mach. Learn. (PMLR)*, 2018, pp. 5123–5132.

[16] Y. Wang, J. Zhang, H. Zhu, M. Long, J. Wang, and P. S. Yu, "Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9154–9162.

[17] H. Wu, Z. Yao, J. Wang, and M. Long, "MotionRNN: A flexible model for video prediction with spacetime-varying motions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15435–15444.

[18] C. Luo, X. Li, Y. Wen, Y. Ye, and X. Zhang, "A novel LSTM model with interaction dual attention for radar echo extrapolation," *Remote Sens.*, vol. 13, no. 2, p. 164, Jan. 2021.

[19] T. Yu, Q. Kuang, and R. Yang, "ATMConvGRU for weather forecasting," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[20] T. Xiong, J. He, H. Wang, X. Tang, Z. Shi, and Q. Zeng, "Contextual sa-attention convolutional LSTM for precipitation nowcasting: A spatiotemporal sequence forecasting view," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 12479–12491, 2021.

[21] C. Zhang, X. Zhou, X. Zhuge, and M. Xu, "Learnable optical flow network for radar echo extrapolation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1260–1266, 2021.

[22] J. Han et al., "Weight loss function for the cooperative inversion of atmospheric duct parameters," *Atmosphere*, vol. 13, no. 2, p. 338, Feb. 2022.

[23] L. Han, H. Liang, H. Chen, W. Zhang, and Y. Ge, "Convective precipitation nowcasting using U-Net model," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–8, 2022.

[24] J. Cuomo and V. Chandrasekar, "Developing deep learning models for storm nowcasting," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2022.

[25] W. Byeon, Q. Wang, R. K. Srivastava, and P. Koumoutsakos, "ContextVP: Fully context-aware video prediction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 753–769.

[26] J. Su, W. Byeon, J. Kossaifi, F. Huang, J. Kautz, and A. Anandkumar, "Convolutional tensor-train LSTM for spatio-temporal learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 13714–13726.

[27] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[28] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[29] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 2, no. 3, 2021, p. 4.

[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[31] A. Dosovitskiy et al., "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2758–2766.

[32] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8934–8943.

[33] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Deep end2end voxel2 Voxel prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 17–24.

[34] J. Y.-H. Ng, J. Choi, J. Neumann, and L. S. Davis, "ActionFlowNet: Learning motion representation for action recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1616–1624.

[35] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Aug. 2020, pp. 213–229.

[36] J. He, L. Zhao, H. Yang, M. Zhang, and W. Li, "HSI-BERT: Hyperspectral image classification using the bidirectional encoder representation from transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 165–178, Sep. 2019.

[37] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2022.

[38] Q. Wang, W. Huang, X. Zhang, and X. Li, "Word–sentence framework for remote sensing image captioning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 12, pp. 10532–10543, Dec. 2021.

[39] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016, pp. 1–48.

[40] S. Ravuri et al., "Skilful precipitation nowcasting using deep generative models of radar," *Nature*, vol. 597, no. 7878, pp. 672–677, Sep. 2021.

[41] Y. Hu, L. Chen, Z. Wang, X. Pan, and H. Li, "Towards a more realistic and detailed deep-learning-based radar echo extrapolation method," *Remote Sens.*, vol. 14, no. 1, p. 24, Dec. 2021.

[42] P. Xie et al., "An energy-based generative adversarial forecaster for radar echo map extrapolation," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[43] C. Wang, P. Wang, P. Wang, B. Xue, and D. Wang, "Using conditional generative adversarial 3-D convolutional neural network for precise radar extrapolation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 5735–5749, 2021.

[44] Y. Kim and S. Hong, "Very short-term rainfall prediction using ground radar observations and conditional generative adversarial networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–8, 2022.

[45] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.

[46] M. Wolfson et al., "Tactical 0–2 hour convective weather forecasts for FAA," in *Proc. 11th Conf. Aviation, Range Aerosp. Meteorol.*, 2004, pp. 1–35.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2016, pp. 630–645.

[48] M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, "The importance of skip connections in biomedical image segmentation," in *Deep Learning and Data Labeling for Medical Applications*. Cham, Switzerland: Springer, 2016, pp. 179–187.

[49] E. Orhan and X. Pitkow, "Skip connections eliminate singularities," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–22.

[50] S. Lee, H. G. Kim, D. H. Choi, H.-I. Kim, and Y. M. Ro, "Video prediction recalling long-term motion context via memory alignment learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3054–3063.

[51] D. Gong et al., "Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1705–1714.

[52] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 3–19.

[53] K. Xu et al., "Show, attend and tell: Neural image caption generation with visual attention," *Comput. Sci.*, vol. 2015, pp. 2048–2057, Feb. 2015.

[54] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1412–1421.

[55] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "EpicFlow: Edge-preserving interpolation of correspondences for optical flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1164–1172.

[56] I. J. Goodfellow et al., "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 1–47.

[57] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.

[58] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 694–711.

[59] M. Veillette, S. Samsi, and C. Mattioli, "SEVIR: A storm event imagery dataset for deep learning applications in radar and satellite meteorology," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 33, 2020, pp. 22009–22019.

[60] *IEEE ICDM 2018 Global AI Challenge on Meteorology*. Accessed: Nov. 5, 2022. [Online]. Available: https://tianchi.aliyun.com/competition/entrance/231662/information?lang=en-us

[61] M. Chen, B. Bica, L. Tüchler, A. Kann, and Y. Wang, "Statistically extrapolated nowcasting of summertime precipitation over the eastern Alps," *Adv. Atmos. Sci.*, vol. 34, no. 7, pp. 925–938, Jul. 2017.

[62] R. Donaldson, R. M. Dyer, and M. J. Kraus, "An objective evaluator of techniques for predicting severe weather events," in *Proc. 9th Conf. Severe Local Storms*, vol. 321326, 1975, pp. 321–326.

[63] R. J. Hogan, C. A. Ferro, I. T. Jolliffe, and D. B. Stephenson, "Equitability revisited: Why the 'equitable threat score' is not equitable," *Weather Forecasting*, vol. 25, no. 2, pp. 710–726, 2010.

[64] M. Robinson, J. Evans, and B. Crowe, "En route weather depiction benefits of the NEXRAD vertically integrated liquid water product utilized by the corridor integrated weather system," in *Proc. 10th Conf. Aviation, Range Aerosp. Meteorol.*, 2002, pp. 1–4.

[65] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.

[66] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.

[67] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," *Proc. Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 6629–6640.

Xichao Dong (Member, IEEE) received the B.S. degree in electrical engineering and the Ph.D. degree in target detection and recognition from the Beijing Institute of Technology (BIT), Beijing, China, in 2008 and 2014, respectively.

From 2011 to 2013, he was a Research Assistant with CTCD, University of Sheffield, Sheffield, U.K. From 2014 to 2017, he was in a postdoctoral position with the School of Information and Electronics, BIT. In 2017, he joined the Teaching Staff of BIT. From December 2019, he was also with the Beijing Institute of Technology Chongqing Innovation Center, Chongqing, China. Since 2021, he has been an Associate Professor at BIT. His research interests include geosynchronous synthetic aperture radar, signal processing, and weather radar.

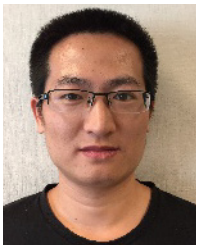
Dr. Dong was a recipient of the IEEE Chinese Institute of Electronics (CIE) International Radar Conference Excellent Paper Award in 2011 and the Chinese Institute of Electronics Youth Conference Poster Award in 2014.

Zewei Zhao was born in Shandong, China, in 1997. He received the B.S. degree in information engineering from the Beijing Institute of Technology, Beijing, China, in 2018, where he is currently pursuing the Ph.D. degree in signal and information processing with the Radar Research Laboratory.

His research interests mainly include signal processing and weather radar.

Yupei Wang received the Ph.D. degree from the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, in 2019.

In July 2019, he joined the Beijing Institute of Technology, Beijing, where he is currently an Assistant Professor. His research interests include computer vision, deep learning, semantic segmentation, and remote sensing image analysis.



Jianping Wang received the Ph.D. degree in electrical engineering from the Delft University of Technology, Delft, The Netherlands, in 2018.

From August 2012 to April, 2013, he worked as a Research Associate with the University of New South Wales, Sydney, Australia, on frequency modulated continuous wave synthetic aperture radar signal processing for formation flying satellites. He is currently a Post-Doctoral Researcher with the Group of Microwave Sensing, Signals and Systems (MS3), Delft University of Technology. His research interests include microwave imaging, signal processing, and antenna array design.

Dr. Wang was a Technical Program Committee (TPC) Member of the Institution of Engineering and Technology (IET) International Radar Conference, Nanjing, China, in 2018. He was a Finalist of the Best Student Paper Award in International Workshop on Advanced Ground Penetrating Radar (IWAGPR), Edinburgh, U.K., in 2017, and the International Conference on Radar, Brisbane, Australia, in 2018. He has served as a reviewer of many IEEE journals.

Cheng Hu (Senior Member, IEEE) received the B.S. degree in electronic engineering from the National University of Defense Technology, Changsha, China, in July 2003, and the Ph.D. degree in target detection and recognition from the Beijing Institute of Technology (BIT), Beijing, China, in July 2009.

He was a Visiting Research Associate with the University of Birmingham, Birmingham, U.K., for 15 months from 2006 to 2007. In September 2009, he joined the School of Information and Electronics, BIT, where he was promoted to be a Full Professor in 2014. He has published over 60 science citation index (SCI)-indexed journal articles and over 100 conference papers. His main research interests include the new concept of synthetic aperture radar imaging, biological detection radar systems, and signal processing.