NowCasting-Nets: Representation Learning to Mitigate Latency Gap of Satellite Precipitation Products Using Convolutional and Recurrent Neural Networks

Mohammad Reza Ehsani, Ariyan Zarei, Hoshin Vijai Gupta, Kobus Barnard, Eric Lyons, and Ali Behrangi

Abstract—Accurate and timely estimation of precipitation is critical for issuing hazard warnings (e.g., for flash floods or landslides). Current remotely sensed precipitation products have a few hours of latency, associated with the acquisition and processing of satellite data. By applying a robust nowcasting system to these products, it is (in principle) possible to mitigate this latency and improve their applicability, value, and impact. However, the development of such a system is complicated by the chaotic nature of the atmosphere, lack of sufficient knowledge about the evolution of precipitation systems based on previous observations, and the consequent rapid changes that can occur in the structures of precipitation systems. In this work, we develop two approaches (hereafter referred to as NowCasting-nets) that use recurrent and convolutional deep neural network (DNN) structures to address the challenge of precipitation nowcasting.

A total of five models are trained using global precipitation measurement (GPM) Integrated MultisatelliteE Retrievals for GPM (IMERG) precipitation data over the Eastern contiguous United States (CONUS) and then tested against independent data for the Eastern and Western CONUS. The models were designed to provide forecasts with a lead time of up to 1.5 h, and by using a feedback loop approach, the ability of the models to extend the forecast time to 4.5 h was also investigated. The performance of the models was compared against the random forest (RF) and linear regression (LR) machine learning (ML) methods, a persistence benchmark (BM) that uses the most recent observation as the forecast, and optical flow (OF). Independent IMERG observations were used as a reference, and experiments were conducted to examine both overall statistics and case studies involving specific precipitation events. Overall, the forecasts provided by the NowCasting-net models are superior, with the convolutional NowCasting-net (CNC) achieving 42%, 24%, 18%, and 16% improvement on the test set mean squared error (MSE) over the BM, LR, RF, and OF models, respectively, for the Eastern CONUS. Results of further testing over the Western CONUS (which was not part of the training data) are encouraging and indicate the ability of the proposed models to learn the dynamics of precipitation systems without having explicit access to motion vectors and other auxiliary features and then to generalize to different hydro-geo-climatic conditions.

Index Terms—Convolutional long short-term memory (ConvLSTM), deep learning (DL), deep neural networks (DNNs), global precipitation measurement (GPM), Integrated MultisatelliteE Retrievals for GPM (IMERG), precipitation nowcasting, UNET.

I. INTRODUCTION

Precipitation has long been one of the most difficult aspects of weather to forecast. Just as ancient people were limited by environmental conditions when planning a hunt, modern people plan their everyday activities around cloudiness and the chance of rain. As weather patterns continue to be altered by climate change and the frequency of extreme weather events increases, it becomes even more important to provide actionable predictions at sufficiently fine spatiotemporal resolutions to be practically useful. Precipitation is a crucial driver for many natural hazardous events (e.g., landslides and flash floods), and its early prediction plays a crucial role in the development of new warning systems [11]–[8].

The term “nowcasting” reflects the need for timely and accurate predictions of risky situations related to the development of severe meteorological events (World Meteorological Organization 2020). Such predictions facilitate effective planning, crisis management, and reduction of loss to life and property. Most importantly, climate change has also recently led to more frequent catastrophic hydrometeorological events such as flash floods in various parts of the world, caused by more intense precipitation events [10]. Given that intense precipitation can cause severe damage to life and property, accurate and reliable nowcasting of precipitation and other hydrometeorological forcings is extremely important [11], [12] and has emerged as a hot research topic in the hydrometeorology community. However, real-time, large-scale, and fine-grained precipitation nowcasting is a challenging task, mainly due to the inherent complexities of the dynamics of the atmosphere and lack of sufficient knowledge about the evolution of precipitation systems [13].

Manuscript received August 15, 2021; revised January 12, 2022 and February 24, 2022; accepted March 9, 2022. Date of publication March 10, 2022; date of current version April 12, 2022. This work was supported in part by NASA MEASURES under Grant NNH17ZDA01N-MEASURES and in part by NASA Weather and Atmospheric Dynamics under Grant NNH19ZDA001N-ATDM. (Mohammad Reza Ehsani and Ariyan Zarei are co-first authors.) (Corresponding author: Ali Behrangi.)

Mohammad Reza Ehsani, Hoshin Vijai Gupta, and Ali Behrangi are with the Department of Hydrology and Atmospheric Sciences, The University of Arizona, Tucson, AZ 85721 USA (e-mail: rehsani@email.arizona.edu; hoshin@email.arizona.edu; behrangi@email.arizona.edu).
Ariyan Zarei and Kobus Barnard are with the Department of Computer Science, The University of Arizona, Tucson, AZ 85721 USA (e-mail: ariyanzarei@email.arizona.edu; kobus@email.arizona.edu).
Eric Lyons is with the School of Plant Science, The University of Arizona, Tucson, AZ 85721 USA (e-mail: ericlyons@email.arizona.edu).
Digital Object Identifier 10.1109/TGRS.2022.3158888
Conventional methods for precipitation nowcasting include the use of storm-scale numerical weather prediction (NWP) models, radar echo extrapolation, and radar extrapolation coupled with NWP and/or stochastic field perturbations [14]–[37]. The radar-based forecasts are limited by ground clutter jamming, beam anomaly, signal attenuation, limited coverage over land, lack of coverage over the ocean, and installation costs. NWP methods must deal with the computational expense of numerically solving partial differential equations subject to dynamic and thermodynamic laws, sensitivity to various kinds of noise, dependence upon initialization, the inability to exploit big data, and the prediction latency resulting from the numerical simulation and data assimilation steps. Accordingly, the development of reliable and timely nowcasting systems remains a critical need to prepare for natural hazards and to conduct related scientific investigations [38]–[41].

Satellite remote sensing is the main observational source for near-real-time global precipitation fields with high spatiotemporal resolution. Satellite products rely mainly on precipitation estimates from infrared (IR) and passive microwave (PMW) observations. IR sensors aboard the geosynchronous-Earth-orbiting (GEO) satellites provide high spatiotemporal sampling, but precipitation estimates based on IR data are indirect and are mainly inferred from the cloud-top temperature. Meanwhile, PMW sensors offer more direct information related to precipitation but are typically mounted only on the low-Earth-orbiting (LEO) satellites and, therefore, provide less frequent sampling [42], [43].

To use the strength of both IR and PMW information, merged precipitation products that combine IR- and PMW-based estimates, such as the global precipitation measurement (GPM) Integrated Multisatellite Retrievals for GPM (IMERG), have been developed. However, the latency associated with these near-real-time global precipitation products is a few hours after the time of observation, which limits their application for many real-time tasks such as flash-flood and landslide warning systems that require timely information [44], [45]. It is (in principle) possible to reduce this latency gap and improve their applicability, value, and impact of these products by applying a robust nowcasting system.

Here, we discuss some recent advances in deep learning (DL) that can help advance precipitation nowcasting. Increased computational power, and the emergence of novel DL structures and techniques, has led to breakthrough applications of deep neural networks (DNNs) to areas such as autonomous vehicles, medical image analysis, object detection and tracking, semantic segmentation of images and point clouds, speech recognition, and natural language processing [46]–[49], among many others.

One area of computer vision that has attracted much attention is the analysis of spatiotemporal data [50]. For example, Reda et al. [51] proposed a model for video frame prediction using spatially displaced convolutions, and Li et al. [52] used a two-phase algorithm for the same purpose by first predicting the flow in the video and then generating the desired frame using the predicted flow information. In this context, note that precipitation nowcasting is essentially a spatiotemporal sequence forecasting problem, where the sequence of recent precipitation maps serves as the input and the consequent sequence of a fixed number of future maps is the desired output. By examining recent advances in DL, with particular attention to recurrent neural networks (RNNs), and more specifically long short-term memory (LSTM) models and convolutional neural networks (CNNs), we can obtain useful insights into how this problem can be addressed.

Machine learning (ML) and DL techniques have already been successfully applied in many areas of geoscience such as precipitation retrieval [42], [43], time-series gap-filling [53], rainfall-runoff modeling [54], [55], hydraulic conductivity estimation [56], and inferring surface water and groundwater exchanges [57]. Many studies have also investigated precipitation nowcasting through DL [35]–[37], [45], [46], [56]–[62]. For example, Shi et al. [62] showed that a convolutional LSTM (ConvLSTM) model using radar maps outperformed the real-time optical flow by variational methods for echoes of radar (ROVER) method for precipitation nowcasting when tested over a small region covered by a single radar.

Ravuri et al. [64] also presented a deep generative model (DeepMind) for the probabilistic nowcasting of precipitation from radar to improve DNN application for nowcasting. While these studies demonstrated that DL-based methods outperform the conventional nowcasting methods such as optical flow (OF), most of them use radar maps, focused on a very small region, and trained the model using data from a specific type of precipitation storm (e.g., convective).

In this study, we propose and test two DL-based precipitation nowcasting approaches (hereafter referred to as NowCasting-nets) trained using IMERG precipitation fields. In theory, these structures should help address the dynamical spatiotemporal nature of the precipitation nowcasting problem. Our study area is the entire contiguous United States (CONUS) that represents a much larger area than the range of a single radar, and the data are representative of different kinds of precipitation events (frontal, convective, etc.), which enables us to assess the potential for real-world large-scale applicability of the approach.

For comparison, we test two conventional ML models, random forest (RF) and linear regression (LR), a persistence benchmark (BM) model that uses the most recent observation as the forecast, and an OF model to demonstrate the potential for improved accuracy and timeliness of precipitation nowcasting achievable using DL. To the best of our knowledge, this is the first study that applies nowcasting to fill in the latency gap between when satellite observations are made and when the precipitation estimate becomes available. Therefore, the main objective of this study is to check whether DL can help to improve the timeliness of the IMERG precipitation product and mitigate its latency gap through DL-based nowcasting.

II. STUDY AREA

The philosophy underlying DL suggests that if we propose a reasonable end-to-end representation for the model and have sufficient data to properly train it, we can obtain a good solution to the input–output mapping problem. For this
study, we have chosen to focus on the CONUS and to use sequences of precipitation maps over the Eastern CONUS to train and test the NowCasting-nets and other models. To further assess the capabilities of the models, the models were also tested over the Western CONUS, data from which were not used during the training and validation stages (see Fig. 1). In contrast to previous studies that used very local regions for model training and testing, our study includes data from a large area and therefore incorporates different types of precipitation events, such as frontal, hurricane, and local convective systems. So, the task is more challenging and can potentially have more real-world applications.

III. DATASET

Precipitation data were obtained from the GPM IMERG product [65], obtained from the Goddard Earth Sciences Data and Information Services Center (GES DISC) at https://disc.gsfc.nasa.gov. IMERG provides three gridded products with different levels of timeliness: early run, late run, and final run. The IMERG early and late runs are quasi-real-time products, released with a lag of 4 and 12 h from real time, respectively. The final run is a research-level product that is bias-adjusted using the Global Precipitation Climatology Centre (GPCC) monthly rain gauge data product [66] that has a time latency of about three to four months. All IMERG products are available at half-hourly 0.1° resolution and go back to June 2000. For this study, six years (2015–2020) of IMERG V06B early run products were used. More information on training and test sets is provided under Section IV.

IV. METHODOLOGY

DNNs have been extensively used in a variety of different fields. One of the novel aspects of recent DNN technologies is their ability to effectively detect and extract the input data features that are most relevant and beneficial to the task at hand. This relieves the user from the need to perform the cumbersome, time-consuming, and challenging task of identifying and extracting the input data features that have the best explanatory power. For the problem of spatiotemporal precipitation prediction, this task becomes even more important, given that the problem of how to relate spatiotemporal features and patterns seen in the data to the dynamics of how precipitation fields evolve is not well understood. Our strategy, therefore, was to perform experiments using the state-of-the-art neural network structures and then to design new models inspired by them to accommodate our domain-specific needs. In particular, we investigated the ConvLSTM [62] and UNET [46] architectures, described below, and proposed five new models that are inspired by them.

These five models, together with four other models (i.e., LR, RF, BM, and OF) enable us to assess and compare the performance of a total of nine models for precipitation nowcasting. Note that the previous studies have already demonstrated the superiority of the DL-based models over conventional methods for precipitation nowcasting (e.g., OF) [39], [61]. The significant contributions of this study are as follows.

1) Explore five DL models that have the ability to simultaneously handle spatiotemporal data over large scales (i.e., CONUS).
2) Provide a way of measuring uncertainty estimates of the DL models.
3) Experiment with learning residuals between the current and next precipitation maps instead of learning actual precipitation values.
4) Produce a valuable dataset for precipitation nowcasting and implementation of the models (that will be publicly available to enable continued investigation into this hot topic, thereby serving the goals of open science).
5) And most importantly, explore the ability of these models to mitigate the latency gap associated with the IMERG/satellite precipitation products.

The nine models investigated in this study are described below.

A. Convolutional-Based Models

UNET models have generally been used for semantic segmentation of 2-D and 3-D images [46]. This network architecture consists of an encoder that maps the input image into a lower dimension space and a decoder that takes a tensor from this space and upsamples it to generate an output mask of the same size as the input image. The underlying intuition is that the lower dimensional encoding causes the model to learn a set of useful features that contain information relevant to generating the desired output. The encoder uses convolution and maxpooling layers to generate the lower dimensional hidden space representation, and the decoder uses deconvolution (transpose convolution) to expand the hidden space and generate the output mask. Several skip or residual connections [67] connect the encoder blocks to the decoder ones; these connections simply concatenate the filters from one of the encoder layers to the inputs of one of the decoder layers. One benefit of such connections is to alleviate the “vanishing gradient” problem and stabilize network training. The following equations explain how the UNET model works:

\[
B_i^D = \text{MaxPool}(W_i^D \ast W_{i1}^D \ast I_i^D) \quad (1)
\]

\[
B_i^U = W_i^{UT} \otimes W_i^U \ast W_{i1}^U \ast I_i^D \oplus I_i^U. \quad (2)
\]

The first equation corresponds to the downsampling operation, where \(B_i^D\) represents the output of the \(i\)th block, \(W_i^D\) and \(W_{i1}^D\) represent the weights of the first and second convolution layers in the \(i\)th block, \(I_i^D\) is the input of the \(i\)th block, and \(\ast\) is the convolution operation. Similarly, the second equation corresponds to the upsampling operation, where \(B_i^U\) represents the output of the \(i\)th block, \(W_i^U\) and \(W_{i1}^U\) represent the weights of the first and second convolution layers in the \(i\)th block, \(W_i^{UT}\) is the weight of the deconvolution layer, \(I_i^U\) is the input of the \(i\)th block, \(\otimes\) represents deconvolution, and \(\oplus\) represents concatenation.

Inspired by this structure, in three of our proposed models, we implement a 3-D network architecture to handle spatiotemporal data and to generate an output sequence of precipitation maps given an appropriate input sequence.

1) Convolutional NowCasting-Net (CNC):

Inspired by the UNET architecture, we designed one of our base models, which we refer to as the CNC. In a manner similar to UNET, the CNC model maps the input to a lower dimensional space and then back to the higher dimensional space of the output sequence. However, unlike UNET, this model takes spatiotemporal data as inputs; specifically, it takes a time sequence (with length \(n\)) of precipitation maps as an input and produces another time-sequence (with length \(k\)) of precipitation maps as an output. The encoder performs spatial downsampling of the input and learns a compact tensor representation of the input sequence. The decoder performs a spatial upsampling of the compact tensor representation using the ConvTranspose layer and adjusts the temporal dimensions (sequence length) by appropriate padding during the convolution layer operations. Like UNET, residual connections link the encoder parts of the network to the decoder parts.

Red dashed box in Fig. 2 shows the structure of this model in detail. Based on the structure of the network and the kernel, padding, and stride of the layers, the intuition is that the encoder learns spatial features from the input sequence and the decoder learns a mixture of both temporal and spatial features. However, squishing the input into the lower dimensional hidden space is thought to reduce the spatial resolution [67]. To overcome this potential limitation, we used the CNC base model as a backbone and proposed two related architectures that we call CNC with residual head (CNC-R) and CNC with dual head CNC-D.

2) Convolutional NowCasting-Net With Residual Head (CNC-R):

This model uses the CNC architecture as a foundation, but instead of learning the actual spatiotemporal features required to generate the desired outputs, it learns the features that help generate the “residual” between each time step of the output and the final time step of the input sequence. The residuals are then added to the output of the last input time step to calculate the expected outputs. Green dashed box in Fig. 2 illustrates the architecture of this network. To implicitly learn the residuals of the output with respect to the last time step of the input, instead of altering the input and output data, an addition layer is added to the output of the CNC model, which enables the backpropagation algorithm to properly update the weights of the internal layers for learning the residuals.

3) Convolutional NowCasting-Net With Dual Head (CNC-D):

By combining the CNC and CNC-R model architectures, we obtain a hybrid model that can generate predictions of both the actual outputs and the residuals with respect to the last time step of the input. This model has two branches corresponding to each of the aforementioned models and their outputs. The fork-like structure of this model includes a shared encoder for learning the spatial features and two branches for learning the spatiotemporal features required to generate the actual outputs and the residuals. The ultimate output of the model is computed as the average of the two intermediate outputs of the two branches. Fig. 2 illustrates the structure of this model.

B. Recurrent-Based Models

ConvLSTM is a specialized class of neural networks that is specifically designed to handle spatiotemporal data. It combines the architectures of CNNs, which are designed to handle image (spatial) data, and LSTM networks, which are designed to handle temporal data and are a subclass of RNNs. So,
like the LSTM, which sequentially processes temporal data while maintaining a memory of important aspects of what has previously been given to the network, the ConvLSTM maintains a memory of important aspects of the previous input images and states and uses convolution as the internal operation instead of matrix multiplication. The following equations represent the flow of data within one block of the ConvLSTM model:

\begin{align*}
    i^t &= \sigma(W_{xi}x^t + W_{ai}a^{t-1} + W_{ci}c^{t-1} + b_i) \\
    f^t &= \sigma(W_{xf}x^t + W_{af}a^{t-1} + W_{cf}c^{t-1} + b_f) \\
    c^t &= o^t \circ c^{t-1} + i^t \circ \tanh(W_{xc}x^t + W_{ac}a^{t-1} + b_c) \\
    o^t &= \sigma(W_{xo}x^t + W_{ao}a^{t-1} + W_{co}c^t + b_o) \\
    a^t &= o^t \circ \tanh(c^t)
\end{align*}

where \( x^t \) is the input sequence at time step \( t \), \( W_{xx} \) is a weight matrix, \( a^t \) is the hidden state at time step \( t \), \( * \) is the convolution operation, and \( \circ \) is hadamard product. More details about the equations and the ConvLSTM structure can be found in [62].

1) Recurrent NowCasting-Net (RNC): This model uses the ConvLSTM as a foundation for dealing with spatiotemporal data and implements ConvLSTM blocks to enable precipitation nowcasting. Each ConvLSTM block receives a current time step of the input and the previous memory (state) and produces one time step of output and an updated memory state that feeds into the next step. Using this basic block, we created new models designed to learn precipitation patterns, by stacking and unrolling ConvLSTM on sequences of precipitation maps. Red dashed box in Fig. 3 illustrates our proposed model.

2) Recurrent NowCasting-Net With Residual Head (RNC-R): Similar to the CNC-R architecture that learns spatiotemporal features relevant to generating residuals of the output with respect to the last time step of the input, it is possible to have an RNC model with a residual head (i.e., RNC-R). We used the same technique as in the CNC-R to force the model to learn the residuals. The architecture of this model is shown in the green box in Fig. 3.

A property of ConvLSTM models is their large need for memory, during training, to enable the internal convolution operations to be performed. This computational barrier poses limitations to the development of a dual branch model similar to the CNC-D structure for RNC.

C. Models Serving as Benchmarks for Comparison

As mentioned earlier, in addition to the DL structures described above, we developed two other ML-based models—RF, LR, a persistence BM, and an OF. The capacity of RF and LR to learn spatiotemporal features is very limited, and such methods are typically used for simple classification and simple trend prediction tasks. The RF and LR models were fed with the same data as the DL models, with the difference that they were fed nine-step sequences of data pixel by pixel, rather than the entire images, and were trained to output three-step sequences at each pixel which were then assembled to create
Fig. 3. General architecture of the RNC and RNC-R models. As it is described in the figure, the red dashed box corresponds to RNC and the green dashed box to RNC-R.

precipitation maps. The RF and LR models were developed using Scikit learn library [68].

The BM model simply repeats the last final time step of the input sequence as the prediction for the next three time steps. While this might seem rather simple to beat, it is worth noting that given the relatively short forecast lead time (1.5 h), the BM prediction is robust and difficult to beat. In addition, OF model was implemented using pySTEPS package [63]. Motion vectors were calculated using the Lucas–Kanade (LK) method [69] and predictions were generated using the “extrapolate” method using these motion vectors [63]. The LK method used to generate the motion vectors is a differential method for OF estimation. Compared with other approaches, this method is less sensitive to pixel noise, which is an advantage since IMERG early maps tend to be noisy. The LK assumes that the flow is essentially constant in a local neighborhood of the pixels under consideration and solves the OF equations for all the pixels in that neighborhood by the method of least squares. Due to this correct assumption, this system has more equations than unknowns, and thus, it is usually overdetermined, which can be properly solved by using least squares.

D. Training, Validation, and Test Sets

As mentioned under Section III, six years (i.e., 2015–2020) of IMERG early half-hourly precipitation products over CONUS were used in this study that equals to $6 \times 365 \times 48 = 105120$ precipitation maps. For each instance in the dataset, we needed 18 consecutive precipitation maps (nine precipitation maps as an input sequence and nine as an output sequence). This means that there are a total of $105120/18 = 5840$ possible instances in our dataset. Some of these instances did not have a much precipitation system, so we had to exclude them from the dataset by manual quality control. At the end of the qualification check, we had 2665 instances in our dataset, each containing 18 precipitation maps. We used 80% of the instances/events for training and validation of the models and the remaining 20% (i.e., 533 events) as an independent test set. First nine time steps were used as input to the model, the next three time steps were used for prediction for +90-min lead time and the next six time steps were used for recursive loop experiments. Each map covers a square area with a length of 25.6°, which is much larger compared to a radar coverage. After training and testing the models with the maps from the Eastern CONUS, the models were also tested against completely independent instances from the Western CONUS to check whether they can learn the dynamics of the precipitation systems for potential application to other regions.

E. Model Training

The aforementioned models were each fed with nine time steps of the IMERG early product and trained to predict the subsequent three-time steps, resulting in a forecast with 1.5-h lead time. To initialize the weights of the DL models, the Xavier method [70] was used, whose basic idea is to keep the variance of the inputs and outputs consistent, to prevent all output values from tending to zero. The Adam optimizer was used to train the network weights by seeking to minimize the total precipitation nowcasting loss. For batch
normalization, the moving average decay was set to 0.9, and a small float of 1E-5 was added to the variance term to avoid division by zero. The learning rate was initialized to 1E-3 and the batch size for training was set to 8. These hyperparameter values were selected based on exhaustive testing of different values via a grid search. All experiments are conducted using a TensorFlow V2 library [71], on a system with AMD EPYC 7542 32-Core Processor, 1008 GB of RAM, and NVIDIA A100 40-GB GPU. Dataset used for training and testing the NowCasting-nets is freely available at https://github.com/ariyanzri/NowCasting-nets.

V. RESULTS

Several experiments were designed to assess the performance of the nowcasting models. The assessment included the analysis of overall statistics based on all events in the independent test set and several case studies using a selection of individual events. The purpose of the models examined here is to provide short-term nowcasts of the dynamic spatiotemporal evolution of precipitation events. A performance evaluation based on metrics that measure grid-by-grid differences [e.g., mean squared error (MSE), correlation coefficient (CC), etc.] can be incomplete because they may not provide a good assessment of the models’ abilities to reproduce the shapes and organizational structures of the events. A more comprehensive evaluation should also include a visual assessment of the actual precipitation maps and their spatiotemporal evolution. We provide such an assessment through several case studies. These events were selected because their spatiotemporal development (change in shape and intensity of the system) was significant, and they could help to compare the models. In addition, they represent convective, frontal, or a combination of both precipitation systems.

Fig. 4 shows nine consecutive half hourly precipitation maps from six different precipitation events (rows A–F) that fell into our study domain. The precipitation maps are from the IMERG early product. As can be seen, the shapes and intensities of the precipitation systems change significantly through each sequence. These maps are fed as an input to each of the models, and the results are reported and discussed in the following. Storm events A and B will be used for assessing predictions with +90-min lead times, events C and D will be examined for extended prediction with +4.5-h lead times, event E is used to estimate the uncertainty of the model predictions, and event F is used for testing the models over the Western United States, where no training was performed.
A. Prediction With +90-min Lead Time

We first checked the performance of the models on an independent testing set that consists of 533 precipitation events (from both summer and winter), each with an input sequence of nine time steps and the output consisting of three time steps. Two types of metrics were used—continuous and categorical (see Fig. 5). The continuous metrics include the MSE, bias ratio (RBIAS), $R^2$, and the CC; see definitions provided in Appendix.

1) Overall Performance of Models on the Testing Set: The CNC-D and CNC models achieved the best MSE (optimal value is 0.0) performance [see Fig. 5(a)], followed by OF for which the three lead time forecasts have mean MSE values of 0.819, 0.833, and 0.980, respectively. Importantly, CNC’s MSE performance does not degrade with increasing lead time and the corresponding mean performance degradation for CNC-D is relatively small compared to the other models. In contrast, MSE performance of the persistence BM scenario degrades significantly with increasing lead time (from...
to deal with spatiotemporal sequences. In general, the CNC models, arising from the ability of the CNC architectures to provide the best performance (mean RBIAS is 1.03, 0.95, 1.06, and 1.07, respectively). The RNC model exhibits significant underestimation (mean RBIAS is 0.640, 0.623, and 0.636 for the first, second, and third lead time forecasts, respectively), while inclusion of the residual head (RNC-R) results in improved bias (mean RBIAS is 1.027, 1.052, and 1.083 for the three lead times). This indicates that the addition of skip connections to the RNC model helps reduce its predictive bias. However, the overestimation bias of the RNC-R predictions does get progressively larger with increasing lead time. In contrast, the underestimation bias of the CNC model (∼0.95) becomes smaller as the lead time increases, indicating that its performance is stable over time. CNC-D model underestimates slightly while CNC-R model overestimates.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.

In terms of RBIAS [see Fig. 5(b); optimal value is 1.0], values larger/smaller than 1.0 indicate overestimation/underestimation], the OF, CNC, BM, and RF models have the best performance (mean RBIAS is 0.76, 0.77, 1.06, and 1.07, respectively). The CNC models exhibit superior MSE performance than the RNC models.
Fig. 6. Precipitation nowcasting for storm A on July 18, 2015, using different models developed in this study up to 1.5-h lead time. Each row shows a time step, and each column represents a model.

CNC and CNC-D architectures provide ∼35% better forecasts in terms of CC and ∼60% better forecasts in terms of MSE compared to the persistence BM. OF has the best performance at the first time step, but its performance drops as the lead time increases. In addition, OF, due to working with motion vectors, has some “nan” values in the precipitation maps, which is clearly undesirable and would damage the results much further if “nan” values are replaced by average values. Overall, Fig. 6 indicates that the models based on the CNC architecture are superior to the others at predicting the dynamic spatiotemporal development of the system with CNC providing the best performance in this case.

The model predictions for event B (see the second row of Fig. 4) are shown in Fig. 7, which corresponds to the spatiotemporal development of a tropical storm system (Hurricane Fred) over the ocean east of the Florida Peninsula. The input sequence indicates that the system is strengthening with time. As with event A, the RNC architecture underestimates the storm intensities (RBIAS is ∼0.55 at all time steps) while capturing its spatiotemporal development acceptably well (CC is ∼0.75 at all time steps). The addition of skip connections improves the ability of RNC-R to predict the intensity of the system, but the MSE progressively degrades with increasing lead time (MSE is 1.12, 1.92, and 3.03 mm/h for the three forecast lead times). It appears that the skip connections interfere with the internal state of the ConvLSTM cells. Both the CNC-D and CNC-R architectures predict the development of the system well, while the CNC without skip connections has slightly better performance (RBIAS is almost perfect and the CC is highest). The simpler LR and RF models underestimate the system precipitation intensities while showing skill in capturing the development of its shape/structure (CC is 0.87, 0.78, and 0.67 for both models). As expected, skill of the BM is poor due to the rapid development of the system (MSE increases from 1 to 3.03 mm/h). Similar to previous case study, OF has the best performance at the first time step, but its performance drops as the lead time increases. In addition, as mentioned before, there are some “nan” values in OF nowcasts. Overall, the CNC-based models perform relatively well, compared to the others.

B. Extending the Nowcast Lead Time to +4.5 h by Feeding the Models With Their Predictions (Feedback Loop)

The IMERG early product is available with about 4-h delay past real time to allow the inclusion of all or most of the input satellite precipitation products into it. For several applications, such as hazard warning (e.g., for flood and landslide) or lumped hydrological modeling, the 4-h latency can negatively impact the usefulness of the product. Here, we investigate the skill of each model to predict precipitation with a +4.5-h lead time, so that if applied to IMERG early, it would enable a user to access “real-time” estimates of precipitation with no latency.

To achieve this, each model was first fed with the input sequence, and then, its predictions were recursively fed back into it as an input until the lead time of 4.5-h was reached.
This requires running each model three times, each time producing a nowcast with a 1.5-h lead time. This experiment helps to assess the range of the model’s capabilities and to check whether the models can reproduce the dynamical evolution of the precipitation systems. It is noteworthy to mention that the initial goal of this study was to develop models that can nowcast nine time steps at once; however, due to computational power issues, we had to limit our model to nowcast only three time steps.

1) Overall Performance of Models on the Testing Set: We first examine the performance of the models over the entire testing set and then investigate two selected case studies (Fig. 4; events C and D). Fig. 8 shows how the mean value of model performance (in terms of four continuous metrics) varies with increasing lead time in half an hour increments. Fig. 8(a) shows that while CNC-R and CNC-D have similar MSE performance at the first three time steps, which was also observed in Fig. 5, as the lead time increases, the CNC-D architecture achieves better performance (less rapid decline) than CNC-R. CNC model also shows the most consistent performance (slower decline) alongside CNC-D model. This suggests that the extrapolation performance of CNC-D is affected by the head, similar to the CNC model. The performance of RNC-R is no better than that of the BM, indicating that adding skip connection significantly affects its performance. The RF and LR models achieve similar performance to each other, as was also observed in previous experiments. OF’s performance also degrades significantly as the lead time increases.

In terms of RBIAS, the RNC model consistently underestimates, as was also observed in the case studies investigated. The CNC-R architecture, although performing relatively well for the first few time steps, progressively overestimates with increasing lead time. This might be due to adding the last time step of the input sequence as the skip connection to the predictions, which is not very beneficial as the lead time increases considering the rapid development in the intensity and deformation of the systems (as observed in the input sequences shown in Fig. 4). CNC and OF have the best performance in terms of RBIAS.

In terms of $R^2$, the CNC and CNC-D models have similar performance, suggesting that CNC models may better reproduce the spatiotemporal dynamics of precipitating systems. As before, the performance of RNC-R and BM drops sharply as the lead time increases. The addition of the skip connections in the RNC-R probably interferes with the internal state of the ConvLSTM, which is the core of the RNC model. The CNC and CNC-D models also have the best performance in terms of CC. Overall, the CNC model outperforms other models at predicting the dynamical spatiotemporal evolution of the precipitation systems with increasing lead times, as was observed over multiple metrics.

2) Case Studies and Visual Assessment: Next, we examine feedback loop experiment for two case studies (i.e., a frontal and a convective system). Fig. 9 shows the results for a frontal system event on August 12, 2015 (see the third row of Fig. 4 for the input sequence of this system). Each row shows a model, and each column represents a time step starting at...
Fig. 8. Comparison of the models’ performance on the test set for the feedback loop experiment using (A) MSE, (B) relative bias (RBIAS), (C) $R^2$, and (D) CC. Here, the lead time is increased up to 4.5 h by recursively feeding the model by its prediction three times.

18:00 UTC. From the input sequence and the observations (last column of Fig. 9), we can see that the precipitation system moved, deformed, and had different intensities over time.

The RNC architecture largely underestimated the precipitation intensities and is unable to capture the development of the system. This is consistent with the overall performance of the RNC model on the entire test set. The performance of RNC-R is also poor, affected by the skip connections as seen in previous experiments. The performance of CNC-D is better than that for the RNC models. The CNC model seems to have the best performance among all models, although a tendency to underestimate persists with increasing forecast lead time. The CNC is best at capturing the change in the shape of the system followed by CNC-D. CNC-R overestimates the precipitation intensity, which is affected by the skip connections. The simpler LR and RF models capture the fact that the system is getting weaker over time, but both tend to underestimate, and RF produces noisy maps with light precipitation. BM overestimates significantly with increasing lead time and does not track the decay of the precipitation intensity over time. OF has similar performance to BM and produces incomplete maps. Overall, Fig. 10 shows that forecasting precipitation events beyond a few hours is very challenging for all models tested here, which is perhaps not surprising given: 1) the highly chaotic space time evolution of real storms and lack of sufficient knowledge about its evolution based on previous observations and 2) the fact that models were trained to optimize forecasting performance over three
VI. DISCUSSION

This section discusses the test results over the western CONUS (transferability), uncertainty associated with NowCasting-nets, and “real-time” applications and challenges (tradeoff between accuracy and latency).

A. Testing for Storms in the Western CONUS

All models in this study were trained with precipitation maps from the Eastern CONUS. In this section, we investigate how well they perform for precipitation systems over the Western CONUS to assess the extent to which the relationships learned by the models can be generalized to other regions. In other words, we assess whether the dynamical precipitation patterns learned for the Eastern CONUS can be used to predict the dynamics of precipitation systems over the Western part.
of the country that has a different hydro-geo-climate. First, we examine overall model performance on an independent test set composed of 533 events over the Western CONUS using continuous metrics. A case study is then examined for visual assessment.

1) Overall Performance of Models on the Testing Set: Fig. 11 indicates that the CNC, CNC-D, and RNC models exhibit consistent MSE performance (slower decline) with increasing lead time. In contrast, other models perform well only for the first few time steps and lose skill thereafter. Although models such as OF do not have a training phase and are based on the previous time steps, their performance is still worse than that of DL-based models, which are trained using the Eastern CONUS precipitation maps. CNC and CNC-D exhibit the best performance among all models. This behavior is consistent with their performance on the Eastern CONUS. CNC also has good RBIAS performance, while LR and RF have a poor performance by significantly overestimating as the lead time increases. CNC-D also underestimates as the lead time increases. CNC-D and CNC have the best $R^2$ performance and outperform other models (especially RF, OF, and BM). CNC-D, CNC, and RNC, have the best

Fig. 10. Performance of different models for the feedback loop experiment for an event on March 8, 2016. Each row shows a model, and each column represents a time step starting at 12:00 UTC.
CC performance. The summary of metrics indicates that the CNC model achieves the best overall performance, generally consistent with the previous results. However, by comparing the CNC-R prediction with +4.5-h lead time with a prediction with +30-min lead time, a 77% reduction in CC and 130% increase in MSE can be seen.

2) Case Studies and Visual Assessment: A precipitation event in the Western CONUS is used to visually compare the performance of the method studied for precipitation prediction, with lead times up to +4.5 h, through the repeated feedback loop experiment (see Fig. 12). The input sequence for this case study is presented in the sixth row of Fig. 4. This storm features large changes in intensity and shape and moves toward the Southeast. As before, RNC underestimates the storm intensities significantly while RNC-R overestimates. The CNC-D architecture captures the decay of the system but cannot capture its movement very good. CNC-R significantly overestimates the system. CNC outperforms other models in reproducing the dynamics of precipitation intensity but is unable to accurately track the intensity of the system. The simpler LR and RF models exhibit very little skill in tracking the development of the system as the lead time increases. The BM persistence benchmark performs poorly due to its inherent inability to track variations in intensity and location of the system. OF also has a very poor performance, similar to BM, despite the fact that it uses data from previous time steps (i.e., inputs) to track the changes in the intensity and the shape of the system.

Overall, the predictions are weaker in the Western CONUS compared to the Eastern CONUS, which is expected for ML- and DL-based models given that the models were trained with data from the Eastern CONUS. Since OF uses local information, it was expected to have better performance over the Western CONUS compared to the other models. However, OF performance depends on the quality of the motion vectors and the method used for generating them. It is interesting that the CNC model shows some skill in capturing the movement patterns and intensity dynamics of the system, suggesting that the model can learn some common aspects of the precipitation dynamics of the Eastern and Western CONUS. In general, CNC exhibited superior performance compared to other models developed in this study.

B. Uncertainty Estimates of DL Nowcasts

Modern nowcasting methods such as those mentioned above in the introduction can generate ensembles to represent uncertainty. This is critically important, particularly with limited accuracy of satellite precipitation products (compared to
The uncertainty question has two aspects. First, what is the predictive uncertainty around the estimates provided by the models (i.e., how precise are the IMERG-like nowcasts and can we put prediction intervals around them, for example, by generating ensembles?). The second aspect, which is harder to tackle, pertains to what is the uncertainty associated with these nowcasts relative to real precipitation? To assess the latter would necessitate access to another separate, higher quality dataset such as stage IV precipitation estimates, which is not within the scope of this article.

To assess the predictive uncertainty associated with the nowcasts generated by DL models developed in this study, we used Monte Carlo Dropout technique [72], which can provide a measure of model uncertainty (in addition to boosting the performance of any trained dropout model without having to retrain it or even modify it at all). In short, when using Monte Carlo Dropout, the model will predict values in the training mode, meaning that the dropout layers are active and randomly drop incoming inputs to them. Therefore, running the model multiple times results in slightly different predictions each time. Using this approach, we generated 20 predictions with dropout over the test set and averaged over them to obtain a Monte Carlo estimate that is more reliable than the result of a single prediction generated without dropout. In addition,
we use the normalized standard deviation of model predictions at each pixel as a measure of model uncertainty; lower values indicate higher certainty associated with the model predictions.

Fig. 13 shows the CNC model predictions (i.e., our best model according to the overall statistical scores and visual assessments), associated uncertainty and ground truth for a frontal system and a convective system (see the fifth row of Fig. 4 for the input sequence of this system). The first row shows the observations for three lead time forecasts, the second row shows the mean of ensemble CNC predictions, and the third row depicts the uncertainty of the CNC predictions. This result indicates that the mean of the ensemble CNC predictions is skillful at capturing the spatiotemporal development of the system. As can be seen by looking at the associated uncertainty map, larger uncertainties are associated with higher precipitation intensities (which is perhaps not surprising).

C. Applications and Challenges

As discussed before, nowcasting performance inevitably degrades with increasing lead time, and this is the case for both radar- and satellite-based methods. Given limitations of satellite precipitation products such as latency gap and limited accuracy especially for heavy precipitation, one can question the usefulness/meaningfulness of IMERG/satellite nowcasts. In other words, why and how satellite-based nowcasting can be made useful? The answer to this question really goes back to the intention of the IMERG developers to produce an early version due to the importance of the early products at the expense of lower quality. While the early product may not benefit from all the available sensors due to the latency issue, it can be improved by bias adjustment using monthly climatology, similar to what is being implemented in the “IMERG Final” product. The need for real-time precipitation...
products is obvious, and a latency in the production of these products will remain because it takes time to collect enough satellite data, transfer them, and estimate precipitation using them. So, in any case, there is a need to fill in the “latency gap” in IMERG or other similar datasets. While radars can provide smaller latency, their coverage is still limited, and so, efforts to improve products such as IMERG remain important for global applications. Real-time applications and challenges (tradeoff between accuracy and latency) of satellite-based precipitation nowcasting need further attention.

VII. Conclusion

Precipitation nowcasting is critical for the successful implementation of hazard warning systems (e.g., for flood and landslide). We developed and tested two state-of-the-art DL approaches for precipitation nowcasting, including five different models, and compared them against classical ML techniques (RF and LR), OF, and a persistence BM. The proposed RNC architecture utilizes ConvLSTM cells, while the CNC architecture is based on the UNET structure and uses convolutional layers. Five variations with and without skip connections were developed, and together with RF, LR, OF, and BM, a total of nine nowcasting models were compared against observations for forecast lead times up to 1.5 h. By using a recursive feedback loop approach, the capability of the models to extend the forecasts up to +4.5 h was also investigated.

Our results indicate that models based on the CNC architecture provide better results than those based on the RNC architecture. Overall, the best performance is achieved by the CNC model, which uses a 3-D CNN to learn spatiotemporal features that are useful for predicting the output. In general, the proposed models were able to learn useful features relevant to the evolution of precipitation patterns in time and space. Note that our approach used only the information provided directly by precipitation maps for nowcasting, and other potentially useful auxiliary data such as motion vectors were not used in this study.

The models based on the CNC architecture, and in particular CNC model, showed good overall predictive skill in terms of categorical (POD, FAR, HSS, and ACC) and continuous (MSE, CC, $R^2$, and RBIAS) skill score metrics on all the events in the independent test set [including 533 events (convective, frontal, etc.]) and individual case studies. To recap, the CNC model provided approximately 42%, 24%, 18%, and 16% better MSE performance than BM, LR, RF, and OF models, respectively. Similarly, the CNC model shows 27%, 36%, 44%, and 1255% better performance in terms of $R^2$ than RF, LR, OF, and BM models. These gains in performance arise from the ability of the CNC model to learn from spatiotemporal data and to extract prediction-relevant features from sequences of precipitation maps.

For the five case studies investigated, the CNC model consistently demonstrated superior performance over the other models, in terms of skill scores, and in the ability to track the changes in shape and intensity of the precipitation systems over time. CNC also showed reasonably good performance at extending the forecasts up to +4.5 h, when implemented in recursive feedback mode, for the Eastern and Western CONUS, indicating its ability to learn the spatiotemporal dynamics of weather systems evolution.

While this and previous studies have demonstrated the superiority of DL-based precipitation nowcasting over conventional nowcasting approaches (such as NWP-based and radar echo extrapolation), some challenges need specific attention and should be addressed in future studies. First, given that use of DL for precipitation nowcasting is still in its early stage, it is not yet clear how models should be evaluated to take into account the needs of real-world applications. Previous studies mainly used precipitation images from a single radar observatory that covers small regions. Here, the study area includes the entire CONUS, and our experiments were designed to test the model at this scale, so that it is likely to be more relevant for real-world applications. However, computational power will be of serious concern when training models for real-world applications.

For effective nowcasting, high-dimensional spatiotemporal sequences with multistep predictions must be made. Because effective training of DNNs requires massive amounts of data and computation power, the computational demands can limit testing of potentially promising approaches. For example, we were unable to train and test a recurrent NowCasting-net with dual head (RNC-D) architecture due to the exorbitant computational expense. While the enhanced skill of DL models comes with additional computational costs that can inhibit their development, when trained, they are often quick and efficient to implement, providing a clear advantage over conventional methods.

Third, rapid changes in intensity, shape, and direction of precipitation systems can complicate the prediction task. Arguably, the problem of precipitation nowcasting is considerably more challenging than comparable tasks in the computer science field such as video prediction. This is mainly because the combined effects of the changes in intensity, deformation, and movement of objects/fields are not common in video prediction. Further, the chaotic characteristics of atmospheric systems tend to result in declining predictive skill with increasing forecast lead time. Of course, this latter issue is not specific to DL-based models and physics-based approaches must also contend with the same issue.

Fourth, the problem of blurriness of images that is widely reported in the DL literature remains unresolved. In this study, we have taken advantage of skip connection and the use of several blocks with batch normalization to help diminish this issue, but there are rooms for improvement.

Last but not least, both DL and ML models require good quality training datasets. Noisy and erroneous observations can adversely affect the ability of the models to extract information regarding the processes that control the spatiotemporal evolution of precipitation patterns. With more powerful computational resources and more accurate and informative sources of information (e.g., from remote sensing satellite platforms), a bright future for the nowcasting of precipitation using DL methods can be expected.
APPENDIX

DEFINITIONS OF THE STATISTICAL METRICS

Quantitative statistics are obtained using estimated (est.) and observed (obs.) quantities. Categorical statistics are obtained using the contingency table. The quantitative and categorical statistics used in the present work are calculated as indicated below, where \( N \) is the total number of observed and estimated precipitation pairs, and \( H \), \( F \), \( M \), and \( Z \) are the numbers of hits, false alarms, misses, and correct negatives, respectively.

\[
\text{POD} = \frac{H}{H + F} \\
\text{FAR} = \frac{F}{H + F} \\
\text{HSS} = \frac{2(HZ - FM)}{(H + M)(M + Z) + (H + F)(F + Z)} \\
\text{ACC} = \frac{N - \sum_{i=1}^{N} (\text{obs}_i - \text{est}_i)^2}{\sum_{i=1}^{N} (\text{obs}_i - \text{est}_i)^2} \\
\text{RBIAS} = \frac{\sum_{i=1}^{N} \text{obs}_i - \sum_{i=1}^{N} \text{est}_i}{\sum_{i=1}^{N} \text{obs}_i} \\
\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (\text{obs}_i - \text{est}_i)^2 \\
R^2 = 1 - \frac{\sum_{i=1}^{N} (\text{obs}_i - \text{est}_i)^2}{\sum_{i=1}^{N} (\text{obs}_i - \text{est}_i)^2}
\]

ACKNOWLEDGMENT

The authors are grateful to the four anonymous reviewers who provided comments and suggestions that helped to improve this article, and especially to reviewer 1 who suggested providing a comparison with the optical flow method and also the generation of ensembles to represent uncertainty, and also provided a comparison with the optical flow method. The authors are also grateful to reviewer 4 who provided valuable and constructive feedback. The authors are also grateful to reviewer 4 who provided valuable and constructive feedback.

REFERENCES

[1] A. Arabzadeh and A. Behrangi, “Investigating various products of IMERG for precipitation retrieval over surfaces with and without snow and ice cover,” Remote Sens., vol. 13, no. 14, p. 2726, Jul. 2021, doi: 10.3390/rs13142726.

[2] A. Arabzadeh, M. R. Ehsani, B. Guan, S. Helfan, and A. Behrangi, “Global intercomparison of atmospheric river precipitation in remote sensing and reanalysis products,” J. Geophys. Res., Atmos., vol. 125, no. 21, Nov. 2020, doi: 10.1029/2020JD033021.

[3] M. R. Ehsani and A. Behrangi, “On the importance of gauge-undercatch correction factors and their impacts on the global precipitation estimates,” Tech. Rep., Jun. 2021. [Online]. Available: https://www.preprints.org/manuscript/202106.0179/v1, doi: 10.20944/preprints202106.0179.v1.

[4] V. Lebedev et al., “Precipitation nowcasting with satellite imagery,” in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Jul. 2019, pp. 2680–2689, Quart. J. Roy. Meteorol. Soc., vol. 144, no. 711, pp. 404–413, 2018, doi: 10.1002/qj.3211.

[5] Y. Song, P. D. Brotxton, M. R. Ehsani, and A. Behrangi, “Assessment of snowfall accumulation from satellite and reanalysis products using SNOTEL observations in Alaska,” Remote Sens., vol. 13, no. 15, p. 2922, Jul. 2021, doi: 10.3390/rs13152922.

[6] G. A. Meehl, F. Zwiers, J. Evans, T. Knutson, L. Mearns, and P. Whetton, “Trends in extreme weather and climate events: Issues related to modeling extremes in projections of future climate change range,” Bull. Amer. Meteorol. Soc., vol. 81, no. 3, pp. 427–436, 2000, doi: 10.1175/1520-0477(2000)081<0427:TIEWAC>2.3.CO;2.

[7] A. Gobiet, S. Kotlarski, M. Beniston, G. Heinrich, J. Rajczak, and M. Stoffel, “21st century climate change in the European Alps—A review,” Sci. Total Environ., vol. 493, pp. 1138–1151, Sep. 2014, doi: 10.1016/j.scitotenv.2013.07.050.

[8] C. F. Durnan, J. M. Gregoire, D. C. Hassell, R. G. Jones, and J. M. Murphy, “A comparison of European daily precipitation simulated by a global and a regional climate model for present and future climates,” Quart. J. Roy. Meteorol. Soc., vol. 127, no. 573, pp. 1005–1015, Apr. 2001, doi: 10.1002/qj.49712757316.

[9] World Meteorological Organization, WMO Statement on the Status of the Global Climate in 2019, WMO, Geneva, Switzerland, 2020, p. 1248.

[10] F. Zhang, J. A. Biederman, M. P. Dunneberg, D. Y. Cao, S. Reed, and W. K. Smith, “Five decades of observed daily precipitation reveal longer and more variable drought events across much of the western United States,” Geophys. Res. Lett., vol. 48, no. 7, Apr. 2021, Art. no. e2020GL092293, doi: 10.1029/2020GL092293.

[11] E. Coppola, D. I. F. Grimes, M. Verdeccia, and G. Visconti, “Validation of improved TAMANN neural network for operational satellite-derived rainfall estimation in Africa,” J. Appl. Meteorol. Climatol., vol. 45, no. 11, pp. 1557–1572, Nov. 2006, doi: 10.1175/JAM2426.1.

[12] Y. Cao et al., “Precipitation nowcasting with star-bridge networks,” 2019, arXiv:1907.08069.

[13] J. Sun et al., “Use of NWP for nowcasting convective precipitation: Recent progress and challenges,” Bull. Amer. Meteorol. Soc., vol. 95, no. 3, pp. 409–426, 2014, doi: 10.1175/BAMS-D-14-00263.1.

[14] Q. Yan, F. Ji, K. Miao, Q. Wu, Y. Xia, and T. Li, “Convolutional residual attention: A deep learning approach for precipitation nowcasting,” Adv. Meteorol., vol. 2020, pp. 1–15, Feb. 2020, doi: 10.1155/2020/84812.

[15] B. J. Turner, I. Zawadzki, and U. Germann, “Predictability of precipitation from continental radar images. Part III: Operational nowcasting implementation (MAPLE),” J. Appl. Meteorol. Climatol., vol. 43, no. 2, pp. 231–248, 2004, [Online]. Available: https://journals.ametsoc.org/view/journals/jamc/43/2/1520-JAMC-D-04-04312.00.0000/latest.

[16] Y. Liu, D.-G. Xi, Z.-L. Li, and Y. Hong, “A new methodology for pixel-quantitative precipitation nowcasting using a pyramidal Lucas Kanade optical flow approach,” J. Hydrol., vol. 529, pp. 354–364, Oct. 2015, doi: 10.1016/j.jhydrol.2015.07.042.

[17] Z. Li, S. P. Ballard, and D. Simonin, “Comparison of 3D-Var and 4D-Var data assimilation in an NWP-based system for precipitation nowcasting at the met office,” Quart. J. Roy. Meteorol. Soc., vol. 144, no. 711, pp. 404–413, 2018, doi: 10.1002/qj.3211.

[18] L. Li et al., “Subpixel-based precipitation nowcasting with the pyramidal Lucas–Kanade optical flow technique,” Atmosphere, vol. 9, no. 7, p. 260, 2018, doi: 10.3390/atmos9070260.

[19] H. Hohi, J. Koistinen, P. Nurmi, E. Saltikoff, and K. Holmlund, “Precipitation nowcasting using radar-derived atmospheric motion vectors,” Phys. Chem. Earth, B, Hydrocl., Ocean. Atmos., vol. 25, no. 10–12, pp. 1323–1327, 2000, doi: 10.1016/S1464-1909(00)00020-1.

[20] L. Han, S. Fu, L. Zhao, Y. Zheng, H. Wang, and Y. Lin, “3D convective storm identification, tracking, and forecasting—An enhanced TITAN algorithm,” J. Atmos. Ocean. Technol., vol. 26, no. 4, pp. 719–732, 2009, doi: 10.1175/2008JTECHA1084.1.

[21] L. Foresti, M. Reyniers, A. Seed, and L. Delobbe, “Development and verification of a real-time stochastic precipitation nowcasting system for urban hydrology in Belgium,” Hydrol. Earth Syst. Sci., vol. 20, no. 1, pp. 505–527, Jan. 2016, doi: 10.5194/hess-20-505-2016.
A. Kumar, Y. Sekimoto, C. Mattmann, and B. Wilson, “ConvNet: an embedded convolutional LSTM based architecture for precipitation nowcasting using satellite data,” PLOS ONE, vol. 15, no. 3, 2020, Art. no. e0230114, doi: 10.1371/journal.pone.0230114.

S.-F. Chang, Y.-C. Liou, J. Sun, and S.-L. Tai, “The implementation of the ice-phase microphysical process into a four-dimensional variational Doppler radar analysis system (VDRAS) and its impact on parameter retrieval and quantitative precipitation nowcasting,” J. Atmos. Sci., vol. 73, no. 3, pp. 1015–1038, Mar. 2016, doi: 10.1175/JAS-D-15-0184.1.

N. E. Bowler, C. E. Pierce, and A. W. Seed, “STEPS: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled NWP,” Quart. J. Roy. Meteorol. Soc., vol. 132, no. 620, pp. 2127–2155, Oct. 2006, doi: 10.1256/qj.05.100.

A. Bellon and G. L. Austin, “The evaluation of two years of real-time operation of a short-term precipitation forecasting procedure (SHARP),” J. Appl. Meteorol., vol. 17, no. 12, pp. 1778–1787, Dec. 1978. [Online]. Available: https://journals.ametsoc.org/view/journals/apme/17/12/1520-0450_1978_017_1778_teotoyo_2_0_co_2.xml, doi: 10.1175/1520-0450(1978)017<1778:etswts>2.0.co;2.

R. Barillé and D. Cornford, “Data assimilation for precipitation nowcasting using Bayesian inference,” Adv. Water Resour., vol. 32, no. 7, pp. 1050–1065, Jul. 2009, doi: 10.1016/j.adwres.2008.09.004.

A. Ayzel, M. Heistermann, and T. Winterrath, “Optical flow techniques,” Quart. J. Roy. Meteorol. Soc., vol. 132, no. 620, pp. 2127–2155, Oct. 2006, doi: 10.1002/qj.2020.

S. Mecklenburg, J. Joss, and W. Schmid, “Improving the nowcasting operation of a short-term precipitation forecasting procedure (SHARP),” J. Hydrometeorol., vol. 23, no. 1, pp. 1015–1038, Mar. 2016, doi: 10.1175/WAF-D-15-0063.1.

M. A. Moghaddam, T. P. A. Ferre, M. R. Ehsani, J. Klakovich, and H. V. Gupta, “Can deep learning extract useful information about energy dissipation and effective hydraulic conductivity from gridded conductivity fields?” Water, vol. 13, no. 12, p. 1668, Jun. 2021, doi: 10.3390/w13121668.

M. A. Moghaddam, T. P. A. Ferre, X. Chen, K. Chen, and M. Reza Ehsani, “Application of machine learning methods in inferring surface water groundwater exchanges using high temporal resolution temperature measurements,” 2022, arXiv:2201.00726.

A. Adhikari, M. R. Ehsani, Y. Song, and A. Behrangi, “Comparative assessment of snowfall retrieval from microwave humidity sounders using machine learning methods,” Earth Space Sci., vol. 7, no. 11, Nov. 2020, doi: 10.1002/2020EA001357.

G. Ayzel, T. Scheffer, and M. Heistermann, “RainNet V1.0: A convolutional neural network for radar-based precipitation nowcasting,” in Geoscientific Model Development Discussions, 2020, pp. 1–20. [Online]. Available: https://gmd.copernicus.org/preprints/gmd-2020-30/gmd-2020-30.pdf, doi: 10.5194/gmd-2020-30.

C.-C. Tsai, S.-C. Yang, and Y.-C. Liou, “Improving quantitative precipitation nowcasting with a local ensemble transform Kalman filter radar observations,” in Geoscientific Model Development Discussions, 2020, Art. no. e0230114, doi:10.5194/gmd-2020-30.

S. Khan, D. B. Kirschtbaum, and T. Stanley, “Investigating the potential of a global precipitation forecast to inform landslide prediction,” Weather Climate Extremes, vol. 33, Sep. 2021, Art. no. 100364, doi: 10.1016/j.wace.2021.100364.

G. Skofronick-Jackson et al., “The global precipitation measurement (GPM) mission for science and society,” Bull. Amer. Meteorol. Soc., vol. 98, pp. 1679–1695, Aug. 2017, doi: 10.1175/BAMS-D-15-00306.1.

O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” 2015, arXiv:1505.04597.

S. Subramanian, R. Li, J. Pilaul, and C. Pal, “On extractive and abstractive neural document summarization with transformer language models,” 2019, arXiv:1909.01386.

C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,” 2016, arXiv:1612.00593.

S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” 2015, arXiv:1506.01497.

D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3D convolutional networks,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 4489–4497, doi: 10.1109/ICCV.2015.510.

F. A. Reda et al., “S-DCNet: Video prediction using spatially-displaced convolution,” Lecture Notes in Computer Science (Including Subseries Lecture Notes in Computer Science Artificial Intelligenct Lecture Notes in Computer Science Bioinformatics) (Lecture Notes in Computer Science), vol. 11211, 2018, pp. 747–763. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-01234-2_44, doi: 10.1007/978-3-030-01234-2_44.

Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang, “Flowgrounded spatial-temporal video prediction from still images,” 2018, arXiv:1807.09755.

M. R. Ehsani et al., “2019–2020 Australia fire and its relationship to hydroclimatological and vegetation variabilities,” Water, vol. 12, no. 10, p. 3067, 2020, doi: 10.3390/w12103067.

Z. Xiang, J. Yan, and I. Demir, “A rainfall-runoff model with LSTM-based sequence-to-sequence learning,” Water Resour. Res., vol. 56, no. 1, Jan. 2020, Art. no. e2019WR025326, doi: 10.1029/2019WR025326.

C. Hu, Q. Wu, H. Li, S. Jian, N. Li, and Z. Lou, “Deep learning with a long short-term memory networks approach for rainfall-runoff simulation,” Water, vol. 10, no. 11, p. 1543, Oct. 2018, doi: 10.3390/w10111543.

M. A. Moghaddam, P. A. T. Ferre, M. R. Ehsani, J. Klakovich, and H. V. Gupta, “Can deep learning extract useful information about energy dissipation and effective hydraulic conductivity from gridded conductivity fields?” Water, vol. 13, no. 12, p. 1668, Jun. 2021, doi: 10.3390/w13121668.

M. A. Moghaddam, T. P. A. Ferre, X. Chen, K. Chen, and M. Reza Ehsani, “Application of machine learning methods in inferring surface water groundwater exchanges using high temporal resolution temperature measurements,” 2022, arXiv:2201.00726.

X. Shi et al., “Deep learning for precipitation nowcasting: A benchmark and a new model,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5618–5628.

B.-Y. Yan, C. Yang, F. Chen, K. Takeda, and C. Wang, “FDNet: A deep learning approach with two parallel cross encoding pathways for precipitation nowcasting,” 2021, arXiv:2105.02585.
[61] Q. K. Tran and S. K. Song, “Computer vision in precipitation nowcasting: Applying image quality assessment metrics for training deep neural networks,” *Atmosphere*, vol. 10, no. 5, p. 244, 2019, doi: 10.3390/atmos10050244.

[62] 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.*, 2015, pp. 802–810.

[63] S. Pulkkinen et al., “Pysteps: An open-source Python library for probabilistic precipitation nowcasting (v1.0),” *Geoosci. Model Develop.*, vol. 12, no. 10, pp. 4185–4219, Oct. 2019, doi: 10.5194/gmd-12-4185-2019.

[64] S. Ravuri et al., “Skilful precipitation nowcasting using deep generative models of radar,” *Nature*, vol. 597, no. 7878, pp. 672–677, Sep. 2021, doi: 10.1038/s41586-021-03854-z.

[65] G. J. Huffman et al., “Integrated multi-satellite retrievals for the global precipitation measurement (GPM) mission (IMERG),” in *Satellite Precipitation Measurement (Lecture Notes in Earth Syst. Sci. Data)*, vol. 10.1007/978-3-319-46493-0_38.

[66] K. Schamm et al., “Global gridded precipitation over land: A description of the new GPCC first guess daily product,” *Earth Syst. Sci. Data*, vol. 6, no. 1, pp. 49–60, Jan. 2014, doi: 10.5194/essd-6-49-2014.

[67] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence Lecture Notes in Computer Science Bioinformatics)* (Lecture Notes in Computer Science), vol. 9908, 2016, pp. 630–645. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-303-24566-9_19, doi: 10.1007/978-3-303-24566-9_19.

[68] K. Schamn et al., “Global gridded precipitation over land: A description of the new GPCC first guess daily product,” *Earth Syst. Sci. Data*, vol. 6, no. 1, pp. 49–60, Jan. 2014, doi: 10.5194/essd-6-49-2014.

[69] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” *Tech. Rep.*, 1981. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.421.4619

[70] W. Sun, F. Su, and L. Wang, “Improving deep neural workflows with multilayer maxout networks,” in *Proc. IEEE Vis. Commun. Image Process. Conf.*, Dec. 2014, pp. 334–337, doi: 10.1109/VCPIC.2014.7051574.

[71] M. Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous distributed systems,” 2016, arXiv:1603.04467.

[72] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” 2015, arXiv:1506.02142.

Mohammad Reza Ehsani received the B.Sc. degree in civil and environmental engineering and the M.Sc. degree in water resources management from the Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2013 and 2019, respectively. He is currently pursuing the Ph.D. degree in hydrology with The University of Arizona, Tucson, AZ, USA. His research interests include remote sensing of precipitation and developing precipitation retrieval algorithms utilizing machine learning and deep learning.

Ariyan Zarei received the B.Sc. degree in computer science from Shahid Beheshti University, Tehran, Iran, in 2018, and the M.Sc. degree from The University of Arizona, Tucson, AZ, USA, in 2020, where he is currently pursuing the Ph.D. degree in computer science. His research interests include the applications of computer vision and machine learning in plant sciences and remote sensing.

Hoshin Vijay Gupta received the B.S. degree in civil engineering from IIT Bombay, Mumbai, India, in 1979, and the M.S. and Ph.D. degrees in systems engineering from Case Western Reserve University, Cleveland, OH, USA, in 1982 and 1984, respectively. He is currently a Regents Professor of hydrology and atmospheric sciences with The University of Arizona, Tucson, AZ, USA. His research interest includes combining physics-based knowledge with machine learning (ML, via information theory) to develop earth and environmental systems models that can learn from interactions with the environment.

Dr. Gupta is a Fellow of the American Geophysical Union and the American Meteorological Society. He was ranked in the top 1% on the Clarivate “Highly Cited Researchers List” for Environment/Ecology in 2017 and 2018. He was a recipient of the American Meteorology Society Robert E. Horton Lecture Award in 2017 and the European Geosciences Union Dalton Medal in 2014. He has served as an Editor for *Water Resources Research* from 2009 to 2013.

Kobus Barnard received the Ph.D. degree in computer science from Simon Fraser University, Burnaby, BC, Canada, in 2000, specializing in color constancy.

He was a Post-Doctoral Fellow of computer vision with the University of California at Berkeley, Berkeley, CA, USA. He is currently a Professor of computer science with The University of Arizona, Tucson, AZ, USA, and has appointments with electrical and computer engineering, cognitive science, statistics, and applied math. His research interests include building and learning explanatory probabilistic models for computer vision and interdisciplinary applications.

Eric Lyons is currently an Associate Professor with the School of Plant Sciences, The University of Arizona, Tucson, AZ, USA. He has authored over 100 peer-reviewed articles and book chapters and teaches students how to use large-scale computing to solve problems and answer questions in biology. His research focuses on scalable computational systems and infrastructure to support and accelerate life science research. To support this, he is a co-PI on CyVerse, a $115M project funded by the National Science Foundation to provide cyberinfrastructure for life science research.

Dr. Lyons serves on several boards of nonprofit companies and research institutions and has worked in biotech, pharma, and software companies around the San Francisco Bay Area. He has served as a Program Director for the National Science Foundation in the Plant Genome Research Program.

Ali Behrangi received the B.S. degree in civil engineering and the M.Sc. degree in water resources engineering from the Sharif University of Technology, Tehran, Iran, in 1999 and 2002, respectively, and the Ph.D. degree in hydrometeorology from the University of California Irvine, Irvine, CA, USA, in 2009.

From 2012 to 2018, he was a Scientist with the NASA Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA. He is currently a Professor of hydrology and atmospheric sciences with The University of Arizona, Tucson, AZ, USA. His research interests include remote sensing of precipitation and water cycle, weather and climate extremes, machine learning, hydrologic applications, and water and energy budget analysis.

Dr. Behrangi received the NASA Early Career Achievement Medal in 2016 and NASA New Investigator Program (NIP) Award in 2014. He is a Fellow of the Kavli Frontiers of Science, National Academy of Sciences and has been a Lead Member of the WCRP Weather and Climate Extreme Grand Challenge and a Member of the WMO/WCRP joint Expert Team on Climate Change Detection and Indices (ETCCDI). He received a title of Distinguished Scholar from The University of Arizona in 2020.