Title

Short-Term Precipitation Forecast Based on the PERSIANN System and LSTM Recurrent Neural Networks

Journal

Journal of Geophysical Research: Atmospheres, 123(22)

ISSN

2169-897X

Authors

Akbari Asanjan, A
Yang, T
Hsu, K
et al.

Publication Date

2018-11-27

DOI

10.1029/2018JD028375

Journal of Geophysical Research: Atmospheres

RESEARCH ARTICLE
10.1029/2018JD028375

Short-Term Precipitation Forecast Based on the PERSIANN System and LSTM Recurrent Neural Networks

Ata Akbari Asanjan[1], Tiantian Yang[1,2], Kuolin Hsu[1,3], Soroosh Sorooshian[1], Junqiang Lin[4], and Qidong Peng[4]

[1]Department of Civil and Environmental Engineering, Center for Hydrometeorology and Remote Sensing, University of California, Irvine, CA, USA, [2]School of Civil Engineering and Environmental Science, University of Oklahoma, Norman, OK, USA, [3]Center for Excellence for Ocean Engineering, National Taiwan Ocean University, Keelung, Taiwan, [4]China Institute of Water Resources and Hydropower Research, Beijing, China

Key Points:
- Artificial intelligence techniques are useful tools in support of forecasting complex precipitation in short range (0–6 hr)
- Long Short-Term Memory structure is capable of learning spatial and temporal correlations, efficiently
- The framework provides accurate precipitation forecasts, especially for the convective systems that have complex evolving dynamics

Correspondence to:
T. Yang,
tiantiay@uci.edu

Citation:
Akbari Asanjan, A., Yang, T., Hsu, K., Sorooshian, S., Lin, J., & Peng, Q. (2018). Short-term precipitation forecast based on the PERSIANN system and LSTM recurrent neural networks. *Journal of Geophysical Research: Atmospheres*, *123*, 12,543–12,563. https://doi.org/10.1029/2018JD028375

Received 23 JAN 2018
Accepted 21 OCT 2018
Accepted article online 29 OCT 2018
Published online 19 NOV 2018

Author Contributions:
Conceptualization: Ata Akbari Asanjan, Tiantian Yang, Kuolin Hsu
Funding acquisition: Soroosh Sorooshian
Investigation: Tiantian Yang, Kuolin Hsu
Methodology: Ata Akbari Asanjan, Tiantian Yang
Resources: Kuolin Hsu, Soroosh Sorooshian
Supervision: Tiantian Yang, Kuolin Hsu, Soroosh Sorooshian, Junqiang Lin, Qidong Peng
Validation: Ata Akbari Asanjan
Visualization: Ata Akbari Asanjan
Writing - original draft: Ata Akbari Asanjan, Tiantian Yang
Writing – review & editing: Tiantian Yang, Soroosh Sorooshian, Junqiang Lin, Qidong Peng

Abstract Short-term Quantitative Precipitation Forecasting is important for flood forecasting, early flood warning, and natural hazard management. This study proposes a precipitation forecast model by extrapolating Cloud-Top Brightness Temperature (CTBT) using advanced Deep Neural Networks, and applying the forecasted CTBT into an effective rainfall retrieval algorithm to obtain the Short-term Quantitative Precipitation Forecasting (0–6 hr). To achieve such tasks, we propose a Long Short-Term Memory (LSTM) and the Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN), respectively. The precipitation forecasts obtained from our proposed framework, (i.e., LSTM combined with PERSIANN) are compared with a Recurrent Neural Network (RNN), Persistency method, and Farneback optical flow each combined with PERSIANN algorithm and the numerical model results from the first version of Rapid Refresh (RAPv1.0) over three regions in the United States, including the states of Oregon, Oklahoma, and Florida. Our experiments indicate better statistics, such as correlation coefficient and root-mean-square error, for the CTBT forecasts from the proposed LSTM compared to the RNN, Persistency, and the Farneback method. The precipitation forecasts from the proposed LSTM and PERSIANN framework has demonstrated better statistics compared to the RAPv1.0 numerical forecasts and PERSIANN estimations from RNN, Persistency, and Farneback projections in terms of Probability of Detection, False Alarm Ratio, Critical Success Index, correlation coefficient, and root-mean-square error, especially in predicting the convective rainfalls. The proposed method shows superior capabilities in short-term forecasting over compared methods, and has the potential to be implemented globally as an alternative short-term forecast product.

1. Introduction

Precipitation is one of the crucial elements of the hydrological cycle, and an important weather phenomenon, which affects human lives in many aspects. Accurate and timely information regarding the upcoming, especially short-term, precipitation events can prevent financial and life losses. In particular, short-term precipitation forecasts, referred to 0–6 hr of lead time, are acute for flash flood warning, flood forecasting, and other hydrological applications (Ganguly & Bras, 2003; Kuligowski & Barros, 1998b; Liu et al., 2017; Vasiloff et al., 2007; Zhu et al., 2017).

Previous studies had shown that the Numerical Weather Prediction (NWP) models and the extrapolation-based methods are frequently used in the short-term precipitation forecast (Bright & Mullen, 2002; Golding, 1998; Kuligowski & Barros, 1998a; Nam et al., 2014; Robertson et al., 2013; Zahraei et al., 2013). The NWP models are using physical characteristics of related atmospheric processes to simulate the dynamics of many meteorological properties, and one of the most important variables is precipitation (Faridzad et al., 2018; Golding, 1998; Ritter & Geleyn, 1992; Warner et al., 1997; Yang, Tao, et al., 2017; Zahraei et al., 2012, 2013). In recent years, the NWP models have improved significantly in many aspects (Ballard et al., 2016; Sun et al., 2014; Wang et al., 2016). For example, the temporal and spatial resolutions of the NWP forecasts have been enhanced by advanced computational resources (Bližňák et al., 2017; Wang et al., 2016). The high-resolution NWP forecasts enabled researchers to improve prediction skills for convective systems (Sun et al., 2014). Despite the improvements attained, the NWP models are still facing many limitations in the short-term forecasting (Foresti et al., 2016). The NWP models with short-range

forecasting purposes require more observational data with higher quality and more effective assimilation methods to overcome the spin-up problems (Shrestha et al., 2013; Wang et al., 2016). Moreover, convective systems require more than sophisticated assimilation methods and sufficient data (Sun et al., 2014). Fine spatial and temporal forecasts are also required for convective studies in order to improve the uncertainties associated with model spin-up and rapid error growth problems (Sun et al., 2014; Wang et al., 2016). Despite the deficiencies of NWP models, they are vastly trusted and used by governmental agencies. In this study, we use the first version of Rapid Refresh (RAPv1.0) model forecasts from National Oceanic and Atmospheric Administration (NOAA)/National Centers for Environmental Prediction (https://rapidrefresh.noaa.gov/) as a numerical baseline comparison to the proposed model.

The RAP is an hourly updated regional NWP model over North America. The RAP model benefits from the community-based Advanced Research version of the Weather Research and Forecasting model and NOAA's Gridpoint Statistical Interpolation analysis system (GSI; Benjamin et al., 2016). The RAPv1.0 forecasts are presented with 13 × 13-km spatial resolution and hourly temporal resolution, which makes the model a proper benchmark candidate.

Different from the NWP models, extrapolation-based methods, or data-driven methods, belong to another category of tools to predict precipitation. The main difference between the NWP models and extrapolation-based methods is that the extrapolation-based methods use statistical approaches to extrapolate the current state of precipitation while the NWP models imitate the rainfall dynamics by using physical-based governing equations. However, the extrapolation-based methods have the advantage of achieving higher forecasting skills during the first few hour(s) of precipitation events with a relatively lower computational cost than the NWP models (Zahraei et al., 2012, 2013). Kuligowski & Barros (1998b) implemented a Three Layer Feed Forward Neural Networks to forecast hourly rainfall in 0- to 6-hr range over Pittsburgh, Pennsylvania. The authors used radiosonde-based 700-hPa wind direction and the historical data of a precipitation gauge network surrounding the target gauge. The results show improvement in rainfall forecasts up to 6 hr. Zahraei et al. (2012) introduced a pixel-based short-term forecasting algorithm. The method tracks severe precipitation events using an iterative algorithm to obtain the advection of mesh cells in space and time (Zahraei et al., 2012). In addition, authors also extrapolated the storm events up to 3 hr using advection field information and a pixel-based Lagrangian dynamic model (Zahraei et al., 2012) and compared the results with the Watershed-Clustering Nowcasting and Persistency methods for 10 storm events over the continental United States (CONUS). The authors evaluated the forecasts from their model along with the two benchmarks using Q2 radar observations. The results obtained by Zahraei et al. (2012) showed promising improvement in terms of verification metrics. As suggested by the above literatures, machine learning and statistical methods have shown their potential to improve the short-term forecasting skills and become popular methods to forecast precipitation events.

The objective of this study is to propose an advanced deep learning algorithm, termed Long Short-Term Memory (LSTM), to forecast the next time step of Cloud-Top Brightness Temperature (CTBT) images from infrared (IR) channel of GOES satellites and iteratively feed the forecasted CTBT image as input to obtain precipitation forecasts with up to 6 hr ahead of time. To demonstrate this concept, we compare the results from the proposed LSTM method with a number of classical extrapolation-based methods, including the Recurrent Neural Networks (RNNs) introduced by Elman (1990), the Farneback Optical Flow method developed by Farnebäck (2003), and Persistency method.

The first generation of RNNs was introduced by Jordan (1997) and then Elman (1990) to find the temporal structure of time-dependent variables. Elman (1990) added a *context unit* to FeedFroward Neural Network (FFNN) architecture in order to represent the information from previous time steps. This structure, often called Elman Networks, has an internal memory of the past events and extends the learned information with an assumption of consistent characteristics of data over time. In general, RNNs are capable of providing a better internal state in comparison with FFNN models and other existing temporal models (Connor et al., 1994; Graves & Schmidhuber, 2005). Despite the soundness of the Elman Networks, the model faces many challenges in learning sequential patterns. According to Guo (2013), one of the challenges is that the RNNs cannot sufficiently be trained by classical backpropagation (BP) algorithm. Hochreiter et al. (2001) and Hochreiter and Schmidhuber (1997a) also indicate the inability of RNNs in learning long-term dependencies even with using more effective learning algorithms like BP Through Time (BPTT). The reason for this

shortcoming in RNNs is the limitations within the BP, which is not suitable for sequential models (Guo, 2013). Many approaches were attempted to improve the general learning capabilities of the Neural Networks (Rumelhart et al., 1985; Werbos, 1988; Williams & Zipser, 1989; Yang, Asanjan, Faridzad, et al., 2017; Yang, Asanjan, Welles, et al., 2017). Despite the improvements in training procedure of RNNs, they cannot capture the temporal features especially the long-term dependencies (Hochreiter et al., 2001; Hochreiter & Schmidhuber, 1997a).

To improve the sequential learning skills of the RNNs, the first LSTM was proposed by Hochreiter and Schmidhuber (1997b). The proposed architecture includes a concept of *control gates* to control the flow of information and prevent the possible model perturbations caused by useless data. Later Gers et al. (2000) introduced a LSTM with abilities to forget useless memories from the memory cell by adding a forget gate to deal with uninformative memory contents.

The overall architecture of LSTM block differs from traditional RNN in two major aspects: (1) the LSTM block tends to excel in learning skills using a sophisticated gated approach where one gate learns the relevance of the input information (input gate), and the other gate learns the importance magnitude of the relevance information (network gate). Putting the above-mentioned gates along with forget gate, which clears the memory of the LSTM block from useless information, creates an efficient and effective learning scheme. On the contrary, traditional RNNs have simple learning scheme, which affects the performances, and (2) the gating structure of the LSTM block allows the model to prevent the gradient decay problem which exists in the traditional RNNs. Thus, allowing LSTMs to learn more complex and longer-range behaviors comparing to traditional RNNs.

In general, LSTM models have shown groundbreaking skills on complex sequential tasks (Byeon et al., 2015; Eck & Schmidhuber, 2002; Graves, 2013). For example, Sundermeyer et al. (2012) used a LSTM model for an English and French language modeling task. The authors found 8% incomprehension improvements as compared to the standard RNNs (Elman Networks) using English Treebank-3 Corpus and the French corpora data sets. Srivastava et al. (2015) designed a LSTM autoencoder to reconstruct and forecast patch sequences of YouTube videos from the Sports-1M data set.

Despite the broad application of LSTM models across different research areas, there are limited studies investigating the applications of LSTM variants in the short-term precipitation forecasting. Shi et al. (2015) introduced a new Neural Network layer, which is an integration of convolutional layer and LSTM layer, termed as ConvLSTM, to better capture the spatiotemporal characteristics of precipitation events. The authors used an encoding-forecasting structure, and radar precipitation data over Hong Kong to forecast precipitation in short range (0–6 hr). Heye et al. (2017) used a similar approach to that used by Shi et al. (2015) and used the ConvLSTM autoencoder to overcome the short-term precipitation forecasting problem. Heye et al. (2017) used NEXRAD radar precipitation data to forecast the upcoming precipitation events.

The aim of current study is to introduce a precipitation-forecasting algorithm that has potentials of becoming an accurate short-term precipitation forecasting product in quasi-global coverage. To address the above-mentioned bottleneck, we used the CTBT data set from GOES satellites, which is a homogenous and continuous data set instead of directly using precipitation data. As compared to the rainfall data, the CTBT data obtained from the GOES satellites provide continuous values for each pixel, and less randomness in the time dependencies on each pixel as the changes of temperature follows the continuity governing law of heat transfer. In contrast, the rainfall characteristics of each pixel are relatively discrete and lack the temporal dependency, especially for small rainfall events. In addition, CTBT data set, provides high temporal and spatial resolution microphysical information regarding the cloud locations and cloud-top temperature (Arkin et al., 1994; Behrangi et al., 2009), which are essential information for precipitation forecasting purposes. The high frequency of CTBT images, which are available for the quasi-global domain, makes the CTBT data set unique and popular for capturing fast varying precipitation fields (Xu et al., 1999).

In order to take advantage of CTBT data set, an effective precipitation retrieval algorithm is required. Due to the indirect relationship of CTBT and precipitation rates (Arkin et al., 1994; Behrangi et al., 2009; Xu et al., 1999), nonlinear mapping functions such as Artificial Neural Networks (ANN) have shown promising potentials in estimating rainfall intensities from CTBT information. In this study, an effective ANN-based precipitation retrieval algorithm termed as Precipitation Estimation from Remotely Sensed Information using ANNs

(PERSIANN) is used. The PERSIANN algorithm is a suitable candidate for estimating precipitation from short-term CTBT forecast due to the capability of the model in estimating high-resolution half-hourly rainfall rate maps where other precipitation retrieval models have coarser temporal resolution (AghaKouchak et al., 2011; Behrangi et al., 2009).

The PERSIANN algorithm introduced by Hsu et al. (1997) is an effective and efficient approach in retrieving rainfall using CTBT data in quasi-global coverage (60°N to 60°S). For instance, Katiraie-Boroujerdy et al. (2013) investigated the performance of CMORPH, PERSIANN, adjusted PERSIANN, and TRMM-3B42 V6 algorithms over Iran, and their results demonstrate that adjusted PERSIANN and TRMM-3B42 V6 are more reliable than other tested products over their case study. AghaKouchak et al. (2011) evaluated the performance of CMORPH, PERSIANN, TMPA-RT, and TMPA-V6 in detecting the extreme precipitation events over the central United States. The authors compared the satellite precipitation data sets to the Stage IV radar data set and found out that CMORPH and PERSIANN have better Probability of Detection (POD) skills comparing to the other products; however, their False Alarm Ratio (FAR) and intensities are higher than TMPA-RT and TMPA-V6. Therefore, we utilized the PERSIANN algorithm to estimate precipitation from forecasted CTBT data.

Besides the methods implemented in this study, two frequently used baseline methods termed as Persistency and Farneback optical flow methods are also implemented to compare the performances of extrapolation-based methods for CTBT and precipitation forecasts. The Persistency method takes the last observed information and assumes no changes will happen during all the lead times (French et al., 1992; Hall et al., 1999; Zahraei et al., 2013). Farneback optical flow method takes last two most recent data and extracts the advection flows for each pixel using the dense optical flow technique. The method assumes steady advection throughout the forecasts and uses the same advection to predict further in time. In this study, first, we forecasted CTBT images using the LSTM, RNN, Persistency, and Farneback models, and then used the PERSIANN algorithm to estimate the corresponding precipitation fields.

Many studies have focused on variations of precipitation types over the United States (Fovell, 1997; Higgins et al., 1998; Wallace, 1975). Based on the different precipitation regimes and types, we select three case studies to evaluate the performances of the proposed LSTM model jointly used with the PERSIANN algorithm under different precipitation mechanisms. Based on the existing precipitation classifications, most of the rainfall events over the states of Oklahoma and Florida are associated with convective systems. In additional, the state of Oregon is also studied to investigate the capabilities of the proposed model under advection-dominant orographic precipitation pattern. The study is designed into two sets of experiments. In the first set of experiment, we compare the CTBT forecast skills of the LSTM, RNN, Persistency model, and the Farneback method for the whole testing period. In addition, the corresponding precipitation forecasts obtained from the combination of the models with the PERSIANN algorithm (respectively referred to as LSTM-PER, RNN-PER, Persist-PER, and Farne-PER) alongside RAPv1.0 numerical forecasts are compared for the whole testing period. In the second set of experiments, we investigate the visual precipitation forecasts from LSTM-PER, RNN-PER, Persist-PER, Farne-PER, and RAPv1.0 for a single precipitation event over each case study region. The rest of this paper is organized as follows. Section 2 describes the implementation steps for obtaining forecasts from the RNN, LSTM, and rainfall estimations for corresponding CTBT forecasts using the PERSIANN algorithms; section 3 provides the details for the data sets and case studies, and parameters of the employed models; section 4 presents the results and discussion. Lastly, the main findings, conclusions, limitations, and future works are presented in section 5.

2. Methodology

2.1. RNNs

The RNNs are capable of encoding information from past events in an internal state (Lipton et al., 2015). The Elman Networks or the Elman-type RNN is one of the most commonly used RNNs (Mao et al., 2014). The Elman-type RNN introduces a *context unit* in the hidden layer, which is able to learn the time-dependent information (Elman, 1990). In this paper, a standard Elman-type Network (Figure 1) is employed. The employed RNN consists of an input layer, which is connected to the hidden layer of the model. The hidden layer has a one-on-one connection to the context layer, which carries the temporal information. In other words, the context layer is another input to the model to represent the time-varying characteristics by

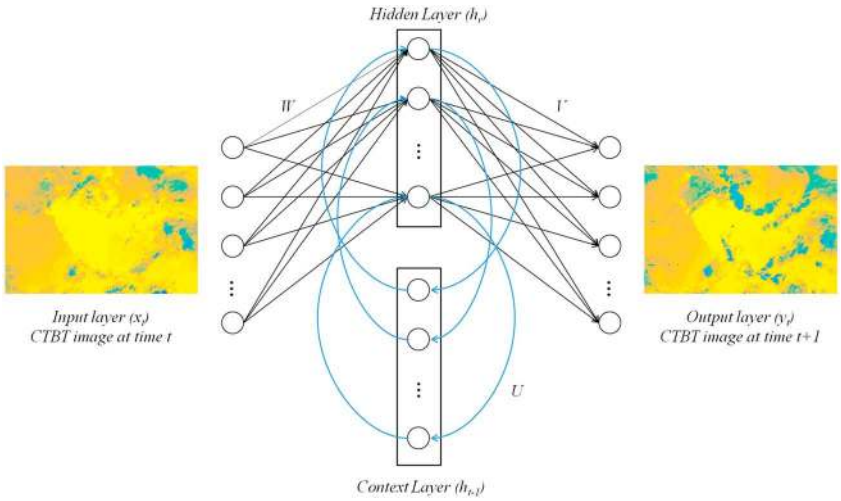

Figure 1. The illustration of an Elman-type Three-Layer Recurrent Neural Network: x_t, h_t, h_{t-1}, and y_t represent the normalized input, hidden state, context state, and the output forecast at time t, respectively. W, U, and V are input-hidden, context-hidden, and hidden-output weights, respectively. CTBT = Cloud-Top Brightness Temperature.

creating an inner loop. The hidden layer is also connected to the output layer to forecast the next time step (Figure 1). In details, a sequence of the CTBT images retrieved from the GOES satellites is first normalized to a range between 0 and 1. The normalized CTBT images at each time step (e.g., at time t) are fed into the RNN model to forecast the next time step (e.g., time $t + 1$). To further discuss the structure of the implemented RNN, let us assume $\boldsymbol{x} = (\boldsymbol{x_1}, \ldots, \boldsymbol{x_T})$ to represent the sequence of CTBT images from time step $\boldsymbol{t} = \boldsymbol{1}$ to $\boldsymbol{T}$, respectively. As shown in Figure 1, the normalized input image $\boldsymbol{x_t}$ is fed into the model, and the RNN model computes the hidden vector sequence $\boldsymbol{h} = (\boldsymbol{h_1}, \ldots, \boldsymbol{h_T})$ by

$$h_t = \sigma\left(W_{ji}x_t + U_{jj}h_{t-1} + b_j\right) \tag{1}$$

$$\begin{aligned} \sigma(a) &= (1 + \exp(-a))^{-1} \\ i &= 1, \ldots, n_0 \qquad n_0 = \text{number of input nodes} \\ j &= 1, \ldots, n_1 \qquad n_1 = \text{number of hidden nodes} \end{aligned} \tag{2}$$

where $\boldsymbol{W}$ is the input-hidden weight vector, $\boldsymbol{U}$ is the context unit weight vector, and $\boldsymbol{b}$ denotes the bias vector. The calculation of outputs is shown in the following equation (3):

$$y_t = V_{kj}h_t + b_k \tag{3}$$

where the vector $\boldsymbol{y} = (\boldsymbol{y_1}, \ldots, \boldsymbol{y_T})$ is the output of the model for time step t = 1, 2, …, T; and $\boldsymbol{V}$ is the hidden-output weight vector.

The cost function is set to the mean square error (MSE) function detailed as follows:

$$E = \frac{\sum_{i=1}^{N}(y_i - o_i)^2}{N} \tag{4}$$

where E, o_i, and N are the output layer error term, target values of ith sample, and the total number of samples, respectively. The described RNN model is trained using the BPTT scheme (Rumelhart et al., 1988).

In order to use the above model for forecasting multiple time steps, an iterative forecast scheme is used in which the model output is fed into the model as the next time step input for desired multiple times. For instance, the first input at time t = 1 is fed into the model, and the output is the forecasted CTBT image at t = 2. Then, the forecasted CTBT image will be fed into the model as input and the output will forecast the CTBT image at time t = 3, and so on.

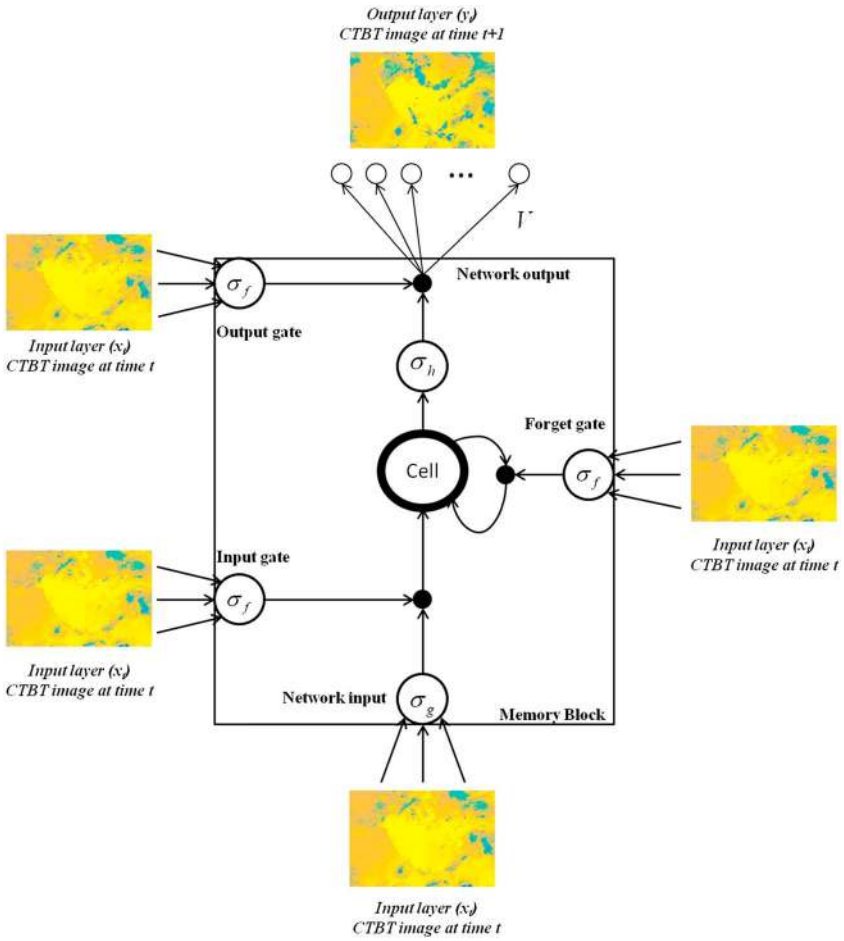

Figure 2. The conceptual illustration of employed LSTM: The x_t and y_t represent the normalized input and the output forecast at time t, respectively. The V represents the LSTM output weights. The Network, Input, Forget and Output gate and Cell are the main components of the Memory Block. The σ_f, σ_g, and σ_h represent the activation functions used for different gates. Note that σ_g and σ_h both represent the *tanh* activation function in this study. LSTM = Long Short-Term Memory; CTBT = Cloud-Top Brightness Temperature.

2.2. LSTM Algorithm

LSTM is a complex recurrent model developed by Hochreiter and Schmidhuber (1997b) to address the deficiencies of RNNs. As mentioned in the previous section, RNN models have simple hidden structures made of a context layer. However, LSTMs consist of one or many memory blocks as its fundamental units, and the memory blocks contain memory cell(s) and gates to control the information flow of the system. According to the literature, there have been many types of LSTM developed to improve the performance of the original model, such as LSTM with Forget gate (Gers et al., 2000), LSTM with peephole connections (Gers & Schmidhuber, 2000), and GRU (Cho et al., 2014). Greff et al. (2017) investigated eight variants of LSTM on different tasks and concluded that there was no significant difference with regard to the performance of different versions of LSTM algorithms. The employed LSTM is from Gers et al. (2000), and a conceptual illustration of the implemented model is shown in the following Figure 2.

As it is shown in Figure 2, a LSTM block consists of an input gate, a forget gate, a memory cell, and an output gate. The following equations (5)–(9) represent the mathematics behind the LSTM Memory Block:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{5}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{6}$$

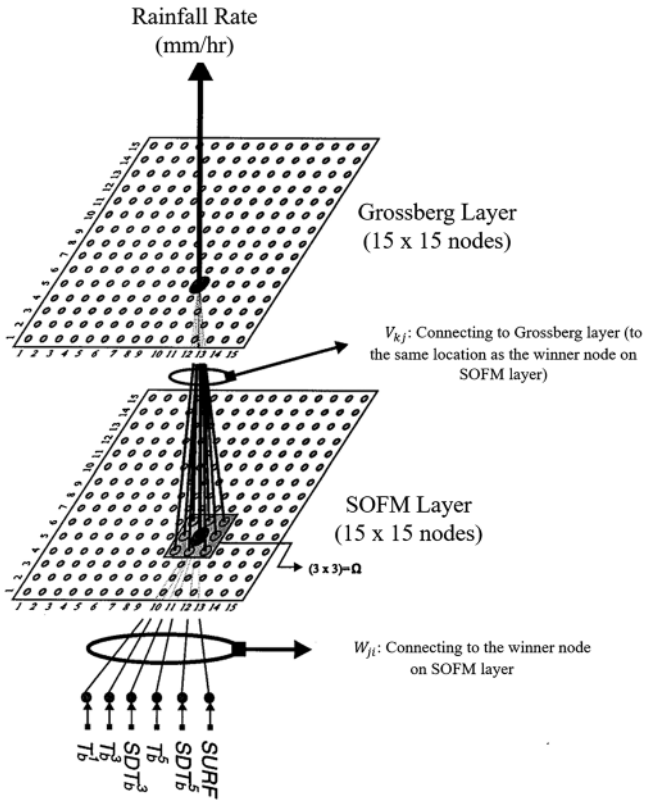

Figure 3. The simple structure of Precipitation Estimation from Remotely Sensed Information using Artificial Neural Network algorithm (Hsu et al., 1997). The bottom layer is the Self-Organizing Map, and the upper layer is the regression layer. The 15 × 15 is the size of hidden layer, and the neighborhood of 3 × 3 is used. The inputs $T_b^1, T_b^3, SDT_b^3, T_b^5, SDT_b^5$, and *SURF* are relatively the Cloud-Top Brightness Temperature (CTBT) of the pixel, mean of 3 × 3 CTBT pixel window, standard deviation of CTBT in 3x3 pixel window, mean of 5 × 5 CTBT pixel window, standard deviation of CTBT in 5 × 5 pixel window, and the index of land, coast, or ocean of the pixel.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (7)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (8)$$

$$h_t = o_t \odot \tanh(c_t) \quad (9)$$

where i, f, c, o, and h are the input gate, forget gate, cell, output gate, and the hidden output, respectively. W_x and W_h in equations (5)–(8) are the input and hidden weights for the gates or cells with the corresponding subscripts, respectively. For example, W_{xf} is the input to forget gate weight matrix. The $\odot$ symbol in equations (7) and (9) represents the inner product of matrices. Note that σ and tanh in equations (5)–(9) represent the Sigmoidal and Hyperbolic Tangent activation functions, respectively. The regression layer of the network is similar to the regression layer used in the RNN model. In this study, the LSTM block is also trained using the BPTT method, and the cost function is identical to that used in the RNN model. A detailed summary of the LSTM training, validation, applications, and additional information is available at Gers et al. (2000) for interested readers.

The implemented LSTM uses a similar autoregressive scheme as in the RNN to forecast up to several time steps in lead time.

2.3. PERSIANN Algorithm

The PERSIANN is an efficient and effective precipitation estimation algorithm, which was originally developed by (Hsu et al., 1997). The heart of PERSIANN consists of a three-layer FFNN with a Self-Organizing Feature Map (Kohonen, 1982) and a regression layer (Figure 3).

The PERSIANN algorithm takes the CTBT images along with the 3 × 3 and 5 × 5 spatial mean and standard deviation of the CTBT image, and the location index of each pixel whether if it is land, coast, or ocean as inputs, and provides precipitation information. The Self-Organizing Feature Map layer uses an unsupervised technique to classify different patterns of input data. Then, a linear regression will estimate precipitation based on the most relevant feature of the hidden layer and its neighborhood.

The PERSIANN has been validated over different regions worldwide, and the reported results indicate promising performances of accurate precipitation estimations (AghaKouchak et al., 2011; Hsu et al., 1997; Katiraie-Boroujerdy et al., 2013; Moazami et al., 2013; Romilly & Gebremichael, 2011).

2.4. Farneback Dense Optical Flow Method

The Farneback dense optical flow method is used in this paper to investigate the performance of optical flow methods, and from now on will be called Farneback method. The Farneback method is a robust algorithm, which takes two sequences of images to estimate the displacement of each pixel. The Farneback method, first, approximates the neighborhood of each pixel using a quadratic polynomial equation:

$$f_1(x) = x^T A_1 x + b_1^T x + c_1 \quad (10)$$

where A_1, b_1, and c_1 are a symmetric matrix, a vector, and a scalar coefficient, respectively. The weights for above equation are estimated using a weighted least squares fit. By creating a new signal f_2 with a global displacement of d, the following equations can be obtained:

$$\begin{aligned} f_2(x) &= f_1(x-d) \\ f_2(x) &= (x-d)^T A_1 (x-d) + b_1^T (x-d) + c_1 \\ f_2(x) &= x^T A_1 x + (b_1 - 2A_1 d)^T x + d^T A_1 d - b_1^T d + c_1 \\ f_2(x) &= x^T A_2 x + b_2^T x + c_2 \end{aligned} \quad (11)$$

The key outcome of the above equilibrium is

$$b_2 = b_1 - 2A_1 d \tag{12}$$

from which the translation d can be solved if A_1 is non-singular

$$d = -\frac{(b_2 - b_1)}{2A_1} \tag{13}$$

In addition to the above-mentioned strategy, a series of fine-tuning is applied to minimize the errors and provide high-performance dense advection fields (Farnebäck, 2003).

2.5. RAP NWP Model

The RAP models are one of the operational assimilation and forecasting models developed by NOAA in response to the need for an accurate short-range NWP model. The first generation of RAP models (RAPv1.0) was developed to increase the short-range forecasting accuracy and replaced the Rapid Update Cycle (RUC) models, which were serving as a preparedness model over the United States and some portions of Canada and Mexico. In May 2012, the previous RUC model was replaced by RAPv1.0, which consisted of more advanced data assimilation techniques and covered larger portion over North America. The RAPv1.0 uses more advanced model components, assimilation components, and horizontal domain in comparison with RUC model. The RAPv1.0 model benefits from the modified GSI, which assimilates radar reflectivity and boundary layer-related observations in hourly resolution. In addition to GSI, the community-based regional Weather Research and Forecasting model is used in RAPv1.0 to include additional precipitation type information. Using the above-mentioned advanced structure, the RAPv1.0 model provides hourly updated forecasts up to 18 hr ahead.

2.6. Statistical Metrics

In order to quantify the capabilities of the presented extrapolation-based models in this study (i.e., Persistency, Farneback, RNN, and LSTM) in forecasting CTBT images, we used correlation coefficient (CC) and root-mean-square error (RMSE) indices for CTBT comparison. The equations used for the comparison indices are

$$RMSE = \sqrt{\frac{\Sigma_{i=1}^{N}(\widehat{y}_i - y_i)^2}{N}}, \quad i = 1, \ldots, N \tag{14}$$

$$CC = \frac{\Sigma_{i=1}^{N}((\widehat{y}_i - \widehat{\mu})(y_i - \mu))}{\sqrt{\left(\Sigma_{i=1}^{N}(\widehat{y}_i - \widehat{\mu})^2\right)\left(\Sigma_{i=1}^{N}(y_i - \mu)^2\right)}} \quad , \qquad i = 1, \ldots, N \tag{15}$$

where $\widehat{y}$, y, $\widehat{\mu}$, and μ are the forecast, observation, mean of the forecast, and the mean of observation, respectively. N is the number of pixels in the study area. For the precipitation result comparison, we chose POD, FAR, and Critical Success Index (CSI) indices on top of the RMSE and CC. The following equations (16)–(18) describe the indices, respectively.

$$POD = \frac{TP}{TP + MS} \tag{16}$$

$$FAR = \frac{FP}{TP + FP} \tag{17}$$

$$CSI = \frac{TP}{TP + FP + MS} \tag{18}$$

where TP, MS, and FP are the number of True Positive, the number of Missed, and the number of False Positive pixels in an event, respectively.

3. Study Regions, Data, and Model Settings

In this paper, we investigated the performances of the proposed model along with baseline models over three case study regions with different precipitation regimes. The case study areas are selected as

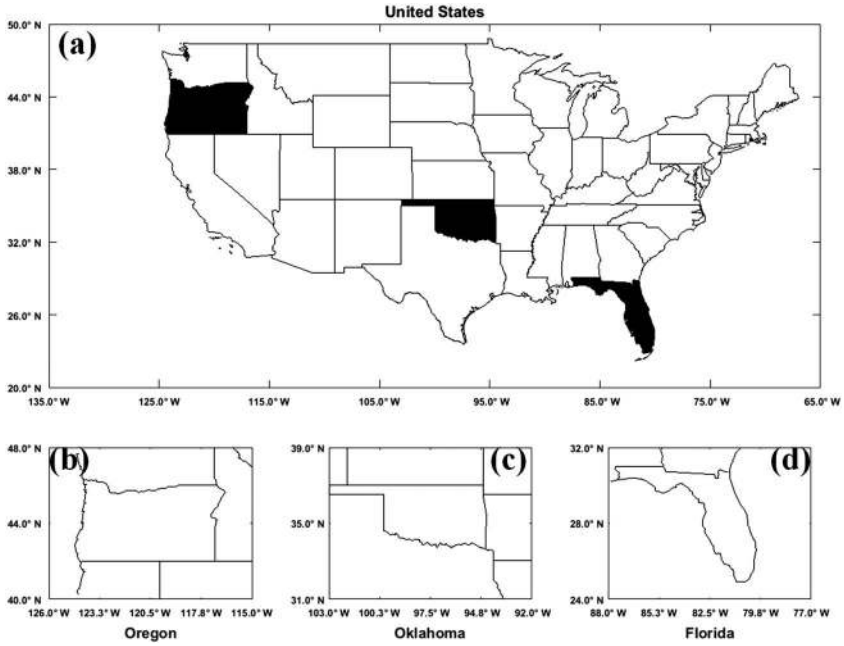

Figure 4. Contiguous United States map (a) and the selected study areas of the states of (b) Oregon, (c) Oklahoma, and (d) Florida.

rectangular regions, which include states of Oregon, Oklahoma, and Florida. The corresponding coordinates of the rectangular regions over state of Oregon, Oklahoma, and Florida are 126°W to 115°W and 40°N to 48°N, 103°W to 92°W and 31°N to 39°N, and 88°W to 77°W and 24°N to 32°N, respectively (Figure 4).

The input data used in the LSTM, RNN, Farneback, and Persistency models is the longwave IR channel of GOES provided by Climatic Prediction Center (http://www.cpc.ncep.noaa.gov/products/global_precip/html/wpage.merged_IR.html). The data provides continuous quasi-global CTBT images with spatial resolution of 0.04° × 0.04°, every 30 min. The combination of GOES East and GOES West produces the CTBT data for the CONUS and adjacent oceans. For the purpose of this study, the CTBT data were regridded using bilinear interpolations to match the resolution of the PERSIANN algorithm, which is 0.25° × 0.25°.

For the ground truth reference, we use the National Mosaic and multisensory Quantitative Precipitation Estimation system (Q2; http://nmq.ou.edu; Zhang et al., 2011). Note that the Q2 radar precipitation estimation is bias corrected using different in-situ observations and quality control algorithms (Lakshmanan et al., 2007; Zhang et al., 2011). The temporal and spatial resolutions for the Q2 data set are 5 min and 0.01° × 0.01°, respectively. In this study, the Q2 data were regridded to 0.25° × 0.25° resolution using bilinear interpolations in order to match the resolution of PERSIANN algorithm.

For comparison purposes, the first version of RAPv1.0 model (https://rapidrefresh.noaa.gov/) is retrieved to be compared with the precipitation forecast produced by the Persist-PER, Farne-PER, and RNN-PER models and the proposed LSTM-PER model. The RAPv1.0 model, with an hourly updating mechanism, is one of the most frequently updated models over North America. The model provides 13 × 13 km, and 51 vertical levels since May of 2012 by NOAA/National Centers for Environmental Prediction (Benjamin et al., 2016). The RAPv1.0 model provides various variables consisting of Atmospheric and Land surface variables. We used the Precipitation rate surface variable, which provides the hourly rainfall rate. We regridded the obtained RAPv1.0 data, using bilinear interpolation, to 25 × 25 km to match the PERSIANN results.

To train the RNN model for each case study region, we implemented a three-layer neural network consisting of an Elman-type RNN layer and a fully connected layer as described in section 2.1. An exhaustive trial-and-error process was conducted to find the optimal hyper-parameters for the model, and the resulted optimal hyper-parameters are 2,000 nodes for the size of the hidden layer, and the RMSProp (Tieleman & Hinton, 2012) optimizer with a learning rate of 0.002 and momentum of 0.9. Moreover, a 25% dropout in the forward

layer and recurrent layer of the RNN, and early-stopping scheme were implemented to prevent the model from overfitting and *simply copy the input* problem.

A similar architecture to the RNN was imposed for the LSTM model with three layers consisting of an LSTM layer and a fully connected layer (described in section 2.2). Based on the outcomes of trial-and-error hyper-parameter search, a same number of nodes and optimizer used in the RNN model was selected for the LSTM network. The RMSProp optimizer was performing the best when the learning rate was set to 0.001 with the momentum of 0.9. Similar to the RNN model, the 25% dropout in the forward layer and recurrent layer of the LSTM was implemented, and the early-stopping scheme was used to prevent the model from overfitting.

The Farneback model was implemented using the last two CTBT observations to estimate the advection, and from there, the same advection field was applied to the last output of the Farneback model to forecast the next time steps. We used a 5×5 neighborhood for the polynomial expansion function, and the model was fine-tuned using 10 iterations.

Due to the stochasticity of Neural Networks including LSTMs and RNNs, the LSTM and RNN models implemented in this study were trained separately over each case study regions for 30 independent runs. For all the study regions, the CTBT data from 1 May 2011 to 1 May 2012 and from 1 May 2012 to 1 May 2013 are used as training and testing data sets, respectively. The Persistency and Farneback models are assumed deterministic, which yield in unique forecasts.

The PERSIANN algorithm was separately trained over each case study region using the same period of training and testing data as used in the LSTM and RNN models. All the hyper-parameters of the PERSIANN algorithm used in this study are identical to those used by (Hsu et al., 1997, 2002).

4. Results

In this section, the results will be presented in two parts: (1) General performances of the CTBT forecast models along with their corresponding precipitation forecasts obtained from jointly use of the models with the PERSIANN algorithm and RAPv1.0 (Figures 5–7) and (2) event-based visual comparison of precipitation forecasts over the case study regions (Figures 8–10).

4.1. General Forecasting Skills

The performance averages of each model forecast in each lead time are calculated for 12 time steps (6 hr) ahead, and the results are presented in Figures 5–7 for the states of Oregon, Oklahoma, and Florida, respectively. Figures 5a and 5b present the average performances of the LSTM, RNN, Farneback, and Persistency methods over the state of Oregon for 12 CTBT prediction time steps in terms of RMSE and CC, respectively. The results show lower RMSE and higher CC for the LSTM forecasts (red line) comparing to the other models. The Farneback model (green line) has the second lowest RMSE values in all the lead time. In Figure 5b, the Farneback method shows higher CC comparing to the LSTM model for the first hour of the forecast; however, the LSTM model deteriorates slower than Farneback model for the remainder of the forecast lead time. The RNN (blue line) shows better performances compared to Persistency method (black line) but has the highest RMSE and the least CC comparing to the LSTM and Farneback method. It is noteworthy that the LSTM model yields more certain results from the 30 independent runs compared to the RNN model.

In Figures 5c–5g, the performances of the corresponding precipitation from the LSTM-PER, RNN-PER, Farne-PER, and Persist-PER along with RAPv1.0 forecasts are compared. It is noteworthy that the presented metrics in Figures 5–7, subplots c–g, are calculated in 30-min intervals for LSTM-PER, RNN-PER, Farne-PER, and Persist-PER. The performance metrics for RAPv1.0 hourly forecasts are also calculated in 30-min intervals by comparing each hourly forecasts from RAPv1.0 to the immediate previous and the same 30-min observation data from Q2 radar data. The corresponding precipitation data were first processed to remove all the no rain events, and then the results with rainy events were compared in terms of RMSE, CC, POD, FAR, and CSI. In order to remove the no rain events, we removed the events where the maximum rain rate of the event was lower than 10% of the maximum rain rate value of that month, and the number of the rainy pixels in each event was less than 10 (i.e., the area of precipitation over each case study was less than 6,250 km^2). Figure 5c shows superior performance of the LSTM-PER model in terms of RMSE except for the first time step.

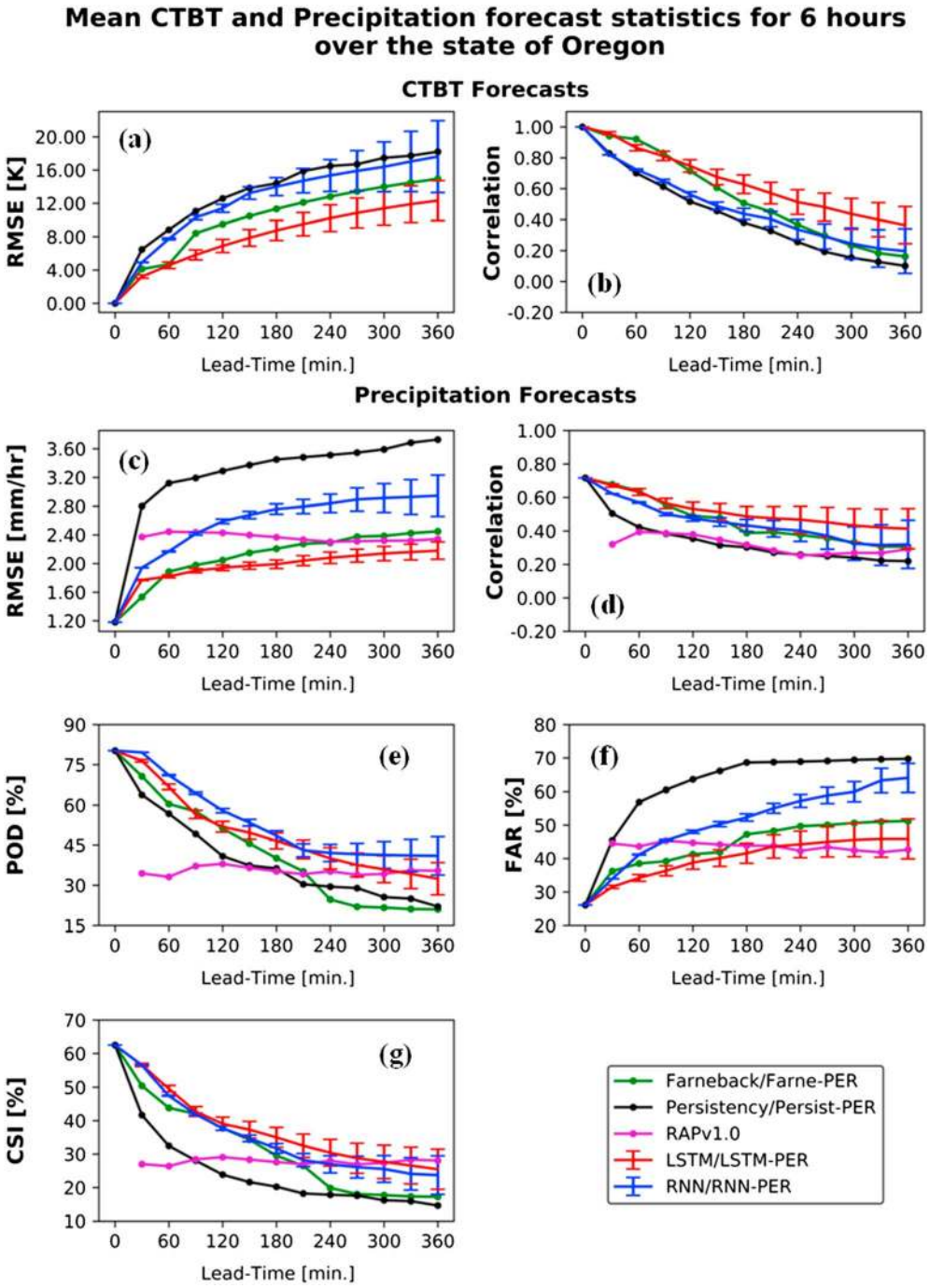

Figure 5. The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (c–g) The average of RMSE, CC, Probability of Detection (POD), False Alarm Ratio (FAR), and Critical Success Index (CSI) indices for the precipitation forecasts from ersist-PER, Farne-PER, LSTM-PER, RNN-PER, and RAPv1.0 models over the state of Oregon.

The Farne-PER model show the second lowest RMSE for the first 4 hr of forecasts and from there the RAPv1.0 (magenta line) becomes the second lowest RMSE. The RNN-PER and Persist-PER are the second and first worst models in terms of RMSE, respectively. In Figure 5d, the LSTM-PER has slightly higher CC comparing to the Farne-PER and RNN-PER models. In Figure 5e, the RNN-PER model has the highest POD values with

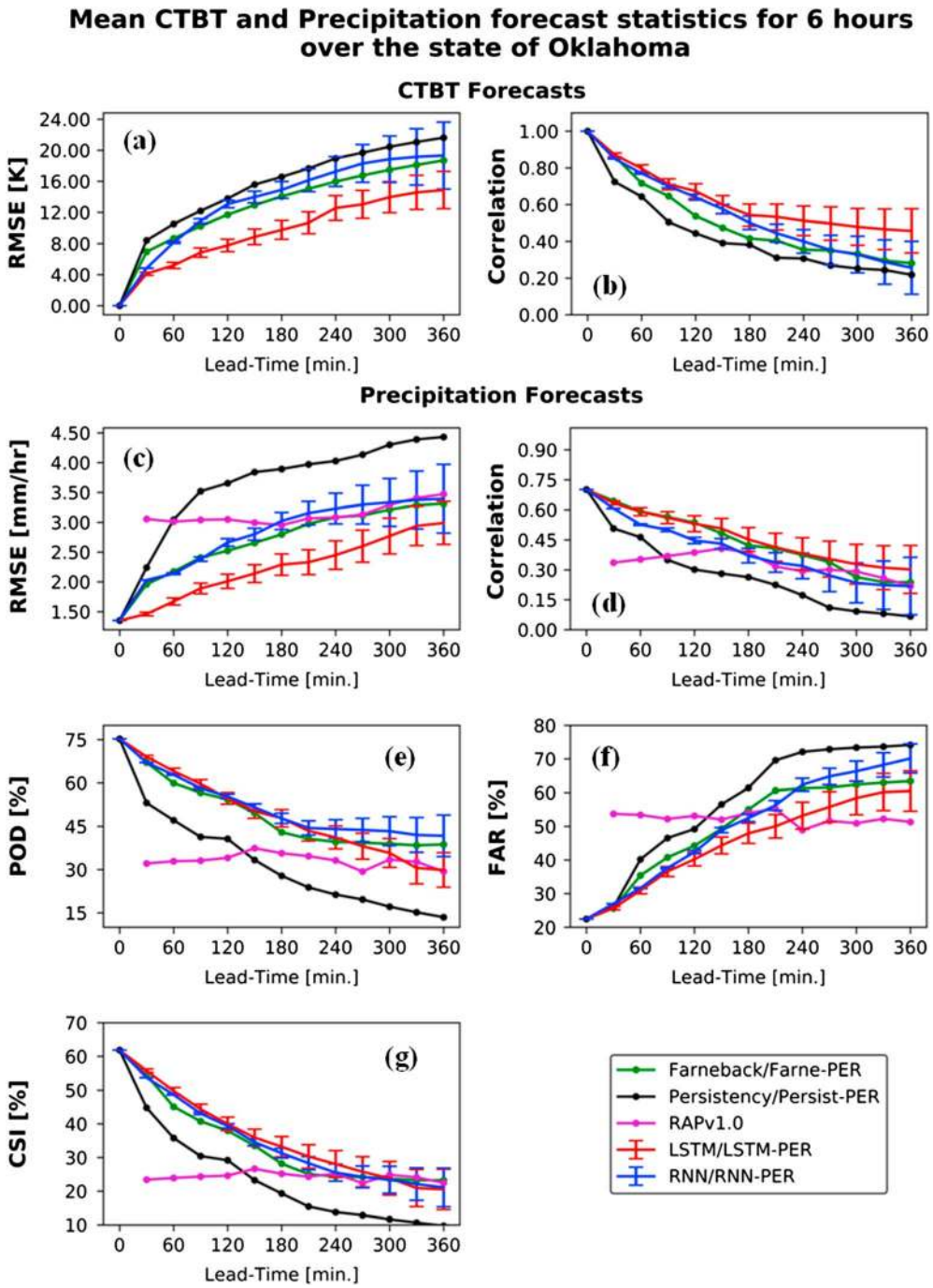

Figure 6. The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (c–g) The average of RMSE, CC, Probability of Detection (POD), False Alarm Ratio (FAR), and Critical Success Index (CSI) indices for the precipitation forecasts from Persist-PER, Farne-PER, LSTM-PER, RNN-PER, and RAPv1.0 models over the state of Oklahoma.

increase of lead time where the LSTM-PER model has the second highest POD values in most lead times. The Farne-PER has higher POD values than Persist-PER and RAPv1.0 up to 4 hr of lead time, and from fourth to sixth hour of forecast, the Farne-PER has the lowest POD values comparing to all baselines. The RAPv1.0 forecasts yield to a steady POD value of 40% throughout the forecast time. In Figure 5f, the LSTM-PER model has the lowest FAR values up to fourth hour of forecast and from there the RAPv1.0 model provides

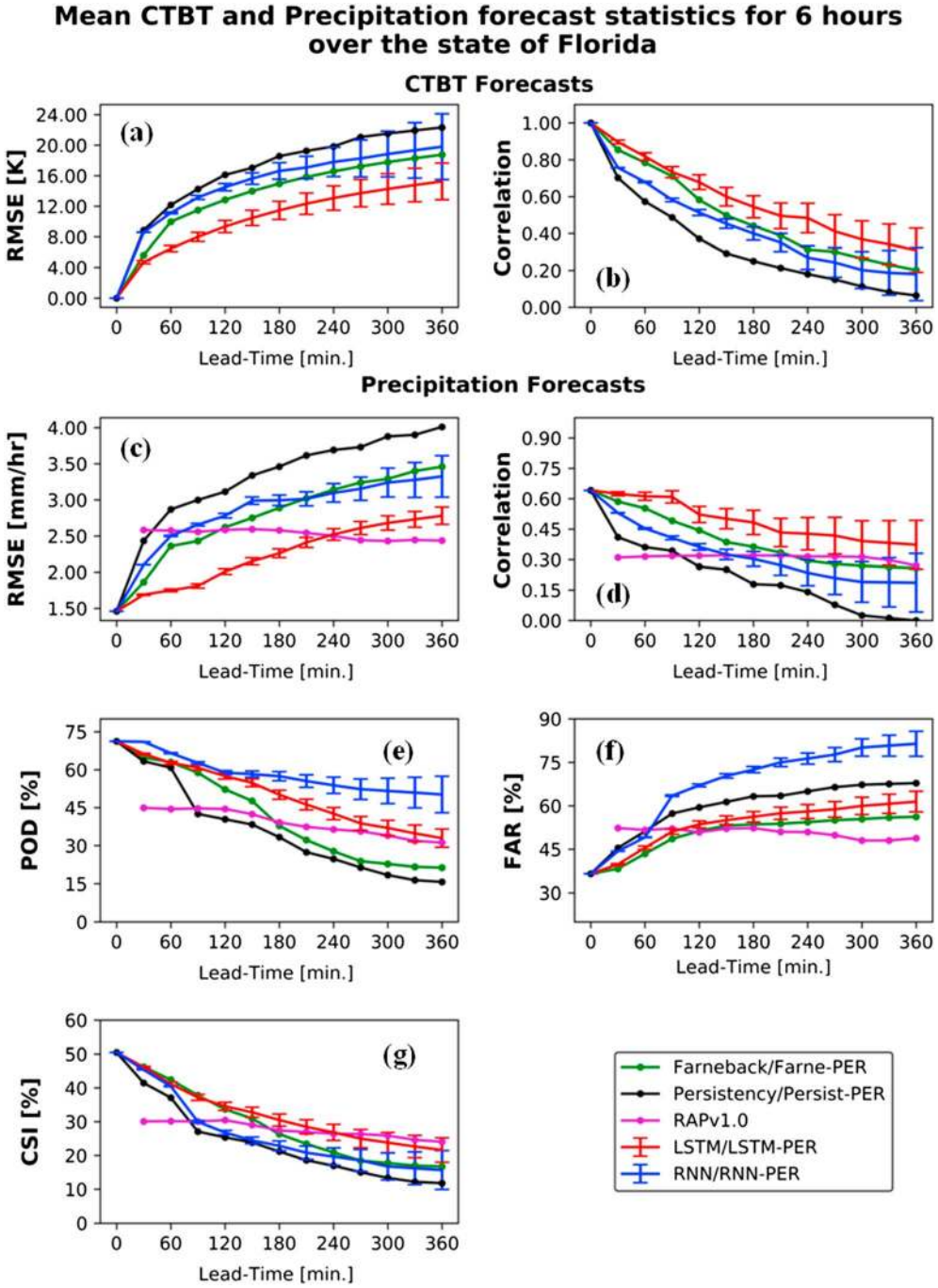

Figure 7. The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (c–g) The average of RMSE, CC, Probability of Detection (POD), False Alarm Ratio (FAR), and Critical Success Index (CSI) indices for the precipitation forecasts from Persist-PER, Farne-PER, LSTM-PER, RNN-PER, and RAPv1.0 models over the state of Florida.

the lowest FAR values. The Farne-PER, RNN-PER, and Persist-PER models have the third, second, and first highest FAR values, respectively. Figure 5g demonstrates the superiority of the LSTM-PER model comparing to the other models with the increase in lead time. The RNN-PER and Farne-PER methods have the second and third highest CSI values up to the last hour of forecast. The RAPv1.0 model has a steady CSI value of 30% with the increase in lead time.

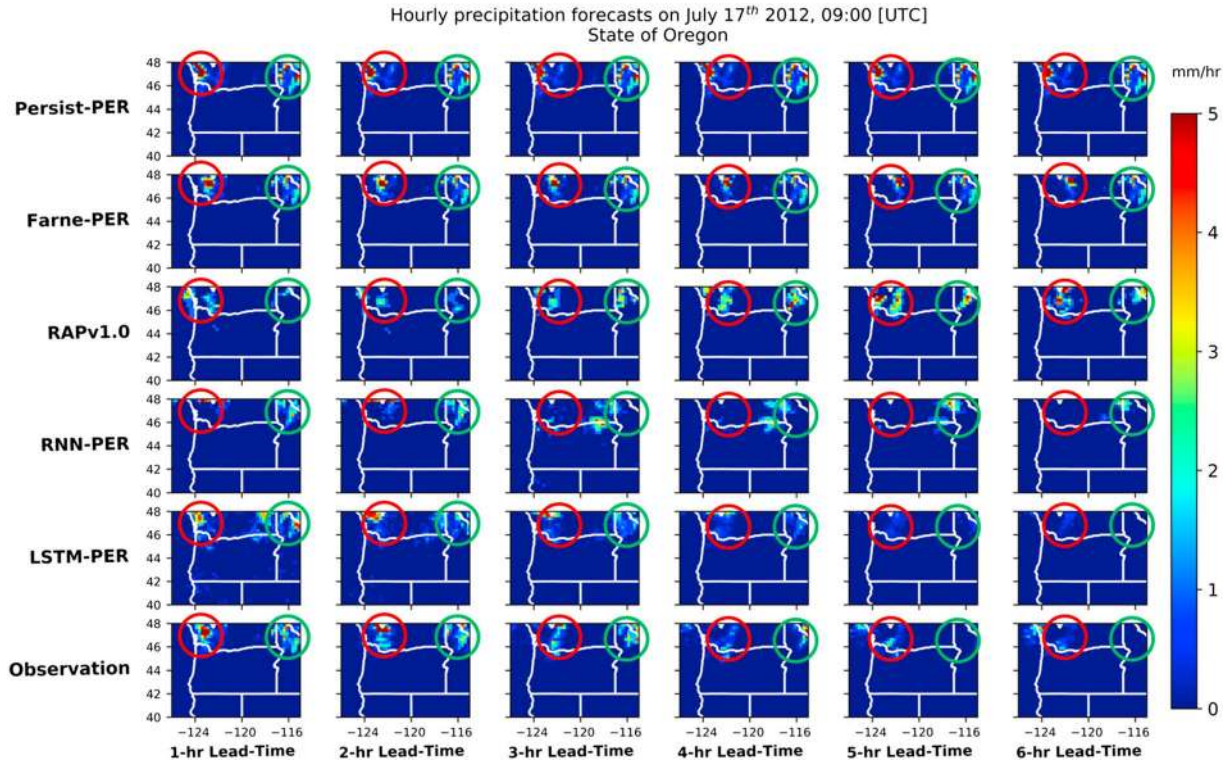

Figure 8. Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Oregon made on 17 July 2012 09:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

Comparing the 30 independent runs for the RNN-PER and LSTM-PER, the forecasts become more uncertain with the increase in lead time. However, the certainty of the RNN-PER model deteriorates faster than the LSTM-PER model in all comparison metrics (Figures 5c–5g).

Figures 6a and 6b presents the average CTBT performances in terms RMSE and CC over the state of Oklahoma. The LSTM model provides forecasts with lowest RMSE and highest CC in all lead times. The RNN model has slightly higher RMSE values comparing to Farneback forecasts; however, the RNN yields higher CC values comparing to the Farneback model. The Persistency provides the highest RMSE and lowest CC values in comparison to other models. The 30 independent runs for the RNN model shows higher uncertainty compared to the 30 independent runs for the LSTM model.

Similar to Figures 5c–5g, Figures 6c–6g show the precipitation forecasting skills over the state of Oklahoma. Figures 6c and 6d show lower RMSE and higher CC for the LSTM-PER model comparing to the other models. The performances of the RNN-PER and Farne-PER models are similar, but the Farne-PER model has slightly lower RMSE and higher CC compared to the RNN-PER results. The RAPv1.0 has higher RMSE and lower CC than the other models except for the Persist-PER results up to 3 hr of the forecast, and after that, the RAPv1.0 model yields similar RMSE and CC compared to the RNN-PER and Farne-PER (Figures 6c and 6d). In Figures 6e, the LSTM-PER, RNN-PER, and Farne-PER models have similar POD values for the first 4 hr of the forecast, and for the rest of lead time, the RNN-PER, Farne-PER, and LSTM-PER have higher POD values, respectively. The RAPv1.0 performs similarly with the increase in lead time in terms of the POD (Figure 6e). In Figure 6f, the LSTM-PER shows lowest FAR up to 4 hr of the forecast, and from fourth to sixth hour of forecast the RAPv1.0 model yields the least FAR values and the LSTM-PER has the second lowest FAR. The RNN-PER and Farne-PER have similar FAR values with increase in lead time, and Persist-PER has the highest FAR (Figure 6e). Figure 6g shows similar performances of the LSTM-PER, RNN-PER, and Farne-PER in terms of CSI with slightly higher skills for the LSTM-PER. In comparison to the other models, except Persist-PER, the RAPv1.0 model has lower CSI values in the first 4 hr of forecast and has similar

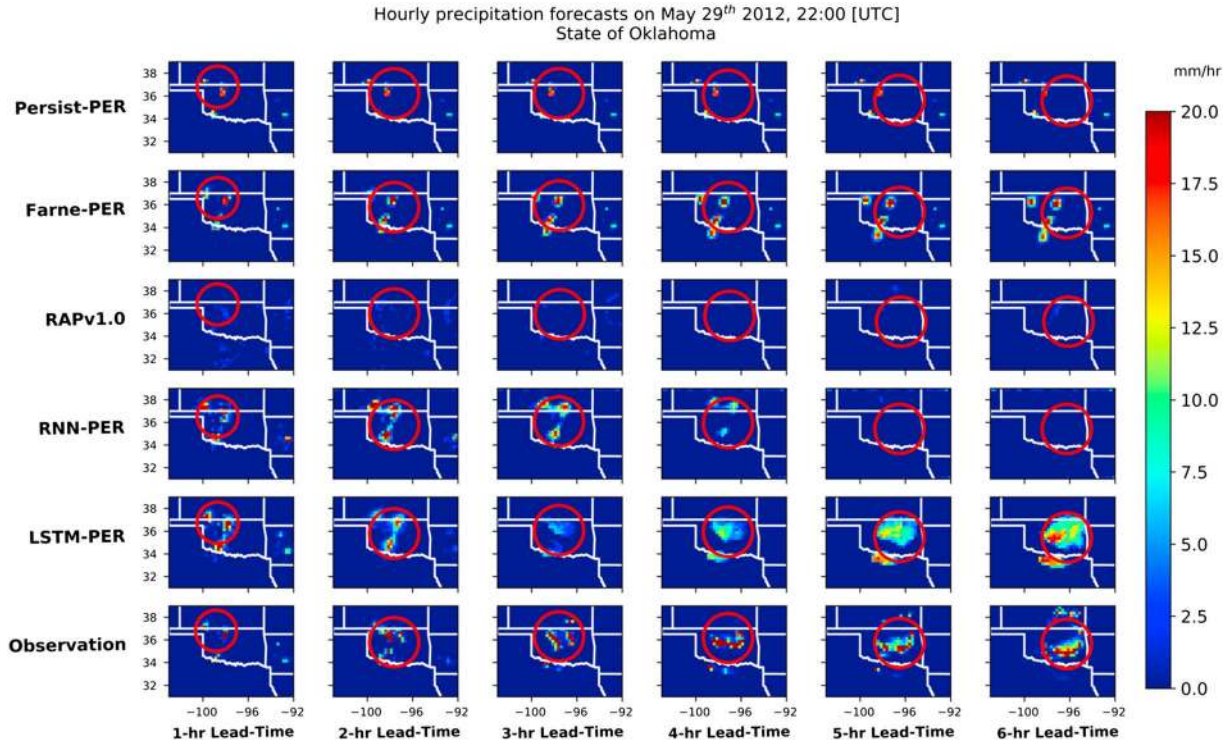

Figure 9. Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Oklahoma made on 29 May 2012 22:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

performance to for the rest of forecast lead time (Figure 6g). Comparing the error bars of the LSTM-PER and RNN-PER, the LSTM-PER tends to have smaller error bounds in all of the comparison metrics presented in Figures 6c–6g.

The statistics for the state of Florida is presented in Figure 7. Based on Figures 7a and 7b, the LSTM model demonstrates superior performances in terms of RMSE and CC for all lead times. The Farneback, RNN, and Persistency models have the second, third, and fourth lowest RMSE, respectively (Figure 7a). The second, third, and fourth highest CC also belong to the Farneback, RNN, and Persistency methods, respectively (Figure 7b). In Figure 7c, the LSTM-PER model yields lower RMSE values, where the RNN-PER and Farne-PER have similar RMSE. The RAPv1.0 model performs consistently close to 2.5 mm/hr with the increase in lead time (Figure 7c). Figure 7d demonstrates highest CC for the LSTM-PER for all of the lead times. The Farne-PER model shows second highest CC up to seventh time step, and after that, RAPv1.0 provides the second highest CC. The RNN-PER model shows lower CC in all lead times comparing to the Farne-PER model, and Persist-PER model performs the worst between all the investigated models in terms of CC (Figure 7d). In Figures 7e and 7f, the RNN-PER model has the highest POD and FAR values almost in all lead times. The LSTM-PER has the second highest POD values in all lead times (Figure 7e). The performances of LSTM-PER in terms of FAR show the second lowest FAR values up to 1 hr of lead time, and third lowest FAR from first- to sixth-forecast hours (Figure 7f). In comparison with the LSTM-PER, the Farne-PER shows lower POD values in Figure 7e, and lower FAR values in Figure 7f in all lead times. The Persist-PER has the lowest POD values in comparison with the RNN-PER, LSTM-PER, and Farne-PER in all lead times (Figure 7e). In Figure 7f, the Persist-PER has higher FAR values than LSTM-PER and Farne-PER in all lead times. In terms of CSI metric, the performances of the Farne-PER and LSTM-PER are similar up to 2 hr of lead time, and after that, the Farne-PER model deteriorate faster than the LSTM-PER (Figure 7g). Respectively, the Persist-PER and RNN-PER have the least and second least CSI values after second hour of lead time, and the RAPv1.0 model provides consistent CSI of 30% for all the lead times (Figure 7g). In the case study of Florida, the LSTM-PER error ranges indicate higher certainty of LSTM-PER model comparing to the error ranges of RNN-PER.

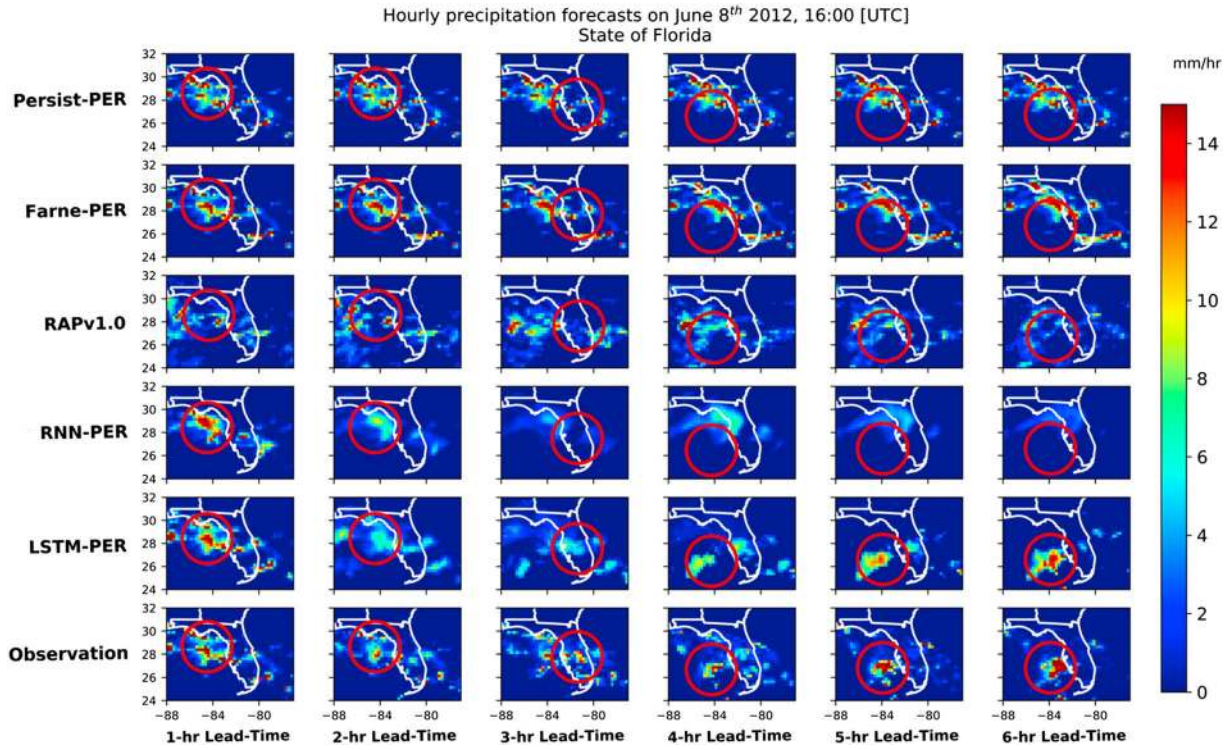

Figure 10. Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Florida made on 8 June 2012 16:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

Table 1 summarizes the performances of the investigated models by averaging the statistics of 12 forecast frames (6 hr) presented in Figures 5–7 for the states of Oregon, Oklahoma, and Florida, respectively. The values in bold in Table 1 present the best statistics for each metric among the investigated models.

In general, Table 1 shows better statistics for the LSTM/LSTM-PER in terms of CTBT RMSE, CTBT CC, rainfall RMSE, rainfall CC, FAR, and CSI compared to other models for the states of Oregon, Oklahoma, and Florida. The POD values in all the three case studies show higher values of the RNN-PER model. The LSTM/LSTM-PER model shows lower uncertainties comparing to RNN/RNN-PER in most metrics for all the case study regions.

4.2. Event-Based Visual Performances

To further investigate the forecasting capabilities of the models, an event-based visual comparison of the model forecasts over the state of Oregon, Oklahoma, and Florida is conducted and presented in Figures 8–10, respectively. It is worth mentioning that all the presented cases over the states of Oregon, Oklahoma, and Florida are selected based on the rainfall events larger than 6,250-km^2 area (Figures 8–10).

Figure 8 shows the visual comparison of the predicted precipitation for the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, LSTM-PER, and observation, which is the Q2 radar data, from top to bottom. From left to right, each column of Figure 8 represents hourly incrementing lead times (from 1 to 6 hr) starting from 17 July 2012 09:00 UTC time. Based on the results from Figure 8, the Persist-PER model has weak visual consistency compared to the observation, and Farne-PER model tends to shift the northwest precipitation patch (marked with red circle) toward the east and keep the northeast rainy patch (marked with green) still. The forecasts from RAPv1.0 model show movement of the northwest rainy patch (red circle) toward the east (Figure 8). The RAPv1.0 forecasts predict a strengthening of the northeast patch (green circle) for the first 2 hr and then the area of rainy patch shrinks. RNN-PER model shows low similarities of shape and intensities for the northwest patch; however, the northeast patch moves toward the west with the increase of lead time. The LSTM-PER forecasts show the most similarities in terms of location, shape, and intensity values of

Table 1
Average Statistical Performances of RAPv1.0, RNN/RNN-PER, LSTM/LSTM-PER, Persistency/Persist-PER, and Farneback/Farne-PER on Testing Data Set

Model	CTBT RMSE (K)	CTBT CC	Rainfall RMSE (mm/hr)	Rainfall CC	POD (%)	FAR (%)	CSI (%)
			Oregon				
RAPv1.0	—	—	2.36	0.31	35.25	43.56	27.70
RNN/RNN-PER	13.22 ± 5.24	0.45 ± 2e-4	2.65 ± 0.03	0.43 ± 0.06	**52.10 ± 7.34**	52.47 ± 6.38	33.71 ± 7.30
LSTM/LSTM-PER	**8.61 ± 2.64**	**0.62 ± 0.01**	**2.01 ± 0.02**	**0.51 ± 0.03**	47.64 ± 5.12	**41.06 ± 4.72**	**35.95 ± 5.10**
Persistency/persist-PER	14.13	0.39	3.4	0.31	37.16	64.75	22.34
Farneback/Farne-PER	10.86	0.52	2.17	0.44	39.29	45.44	29.54
			Oklahoma				
RAPv1.0	—	—	3.13	0.33	33.17	52.40	24.26
RNN/RNN-PER	14.56 ± 6.35	0.51 ± 0.05	2.90 ± 0.12	0.37 ± 0.03	**50.18 ± 7.24**	52.33 ± 4.31	32.94 ± 7.21
LSTM/LSTM-PER	**10.18 ± 2.23**	**0.59 ± 0.04**	**2.30 ± 0.10**	**0.45 ± 0.02**	47.05 ± 5.29	**46.97 ± 6.42**	**34.02 ± 5.27**
Persistency/Persist-PER	16.37	0.39	3.79	0.24	29.50	59.63	21.39
Farneback/Farne-PER	13.87	0.47	2.79	0.43	47.21	51.87	31.97
			Florida				
RAPv1.0	—	—	2.52	0.31	38.90	**50.68**	27.62
RNN/RNN-PER	15.90 ± 3.35	0.40 ± 0.05	2.93 ± 0.23	0.31 ± 0.06	**57.43 ± 4.67**	69.87 ± 3.45	24.79 ± 4.66
LSTM/LSTM-PER	**11.15 ± 1.60**	**0.56 ± 0.04**	**2.28 ± 0.11**	**0.48 ± 0.05**	48.71 ± 3.18	54.81 ± 3.38	**30.88 ± 3.17**
Persistency/Persist-PER	17.74	0.29	3.42	0.19	33.56	61.32	21.99
Farneback/Farne-PER	14.44	0.46	2.87	0.38	39.48	51.61	27.63

Note. Entries in bold are values that represent the best performances between the models. The RNN/RNN-PER and LSTM/LSTM-PER include the uncertainties obtained from the 30 independent runs of the RNN and LSTM models. CTBT = Cloud-Top Brightness Temperature; RMSE = root-mean-square error; RAPv1.0 = Rapid Refresh; RNN = Recurrent Neural Network; LSTM = Long Short-Term Memory; CC = correlation coefficient; POD = Probability of Detection; FAR = False Alarm Ratio; CSI = Critical Success Index.

rainfall as compared to Persist-PER, Farne-PER, RAPv1.0, and RNN-PER forecasts. The evolution of the northwest rainy patch is well captured by LSTM-PER, and the northeast rainy patch dissipates similar to the observation (Figure 8).

Figure 9 demonstrates the precipitation forecasts on 26 May 2012 22:00 UTC time over the state of Oklahoma. The results from Persist-PER indicate poor performances in terms of the location, shape, and intensity of the rainy patches. The Farne-PER forecasts show small movements of three rainy patches toward the center of the Oklahoma state (Figure 9) but fail to foresee the growth of the northwest and southern rainy patches. The RAPv1.0 model does not capture the high-intensity rainfall in the state of Oklahoma; however, small-intensity rainfall is forecasted. The locations of rainfall forecasted by RAPv1.0 model is close to the center of intensity of the observed precipitation (Figure 9). RNN-PER model shifts the three patches of rainfall toward midstate area up to the fourth hour of the forecast, and from there the precipitation patches dissipates. In Figure 9, LSTM-PER shows similar dynamics in terms of location of rainfall with an increase of lead time. The intensity of the LSTM-PER forecasts tends to underestimate especially in third and fourth hours of lead time. The shapes of rainy patches are similar to the observations; however, the LSTM-PER forecasts produce rainfall over a larger area compared to the observation (Figure 9).

Figure 10 presents the visual forecasts from the investigated models on 8 June 2012 16:00 UTC time over the state of Florida. Persist-PER model provides the last observed precipitation pattern, which has poor forecasting skills in terms of shape, location, and intensity of the rainy patches. The Farne-PER has insignificant changes with the increase in lead time; however, the small rainy patches tend to get closer and unite in the last hours of the forecast (Figure 10). The RAPv1.0 forecasts show a complex rainfall mass moving toward the southeast. The RAPv1.0 forecasts are in acceptable agreement with observations in terms of the evolving direction of the rainy patch; however, the shape and intensities of the rainy patches are not similar to the observations (Figure 10). The forecasts from RNN-PER demonstrate high similarities with observations up to 3 hr of lead time, and from third to sixth hour the rainy patch moves east toward the inland Florida state. The direction of rainy patch forecasted between third and sixth lead time hours does match the direction in the corresponding observations (Figure 10). The rainfall intensities are underestimated by RNN-PER model, especially in the fifth and sixth hours of the forecast. Based on the forecasts in Figure 10, LSTM-PER has good agreements with the observations in terms of the shape and the center of intensity. However, the LSTM-PER model slightly underestimates the rainfall intensities in all the lead times.

5. Discussion

5.1. Forecasting Performances

According to Figures 5–7, the statistical metrics for testing period show more accurate forecasting results from the LSTM/LSTM-PER model compared to the Persistency/Persist-PER, Farneback/Farne-PER, RNN/RNN-PER, and RAPv1.0. In general, the LSTM/LSTM-PER shows slower deterioration of performances compared to the other investigated methods (Figures 5–7). The LSTM/LSTM-PER method has better forecasting skills than the Farneback/Farne-PER model due to its capabilities in detecting the dynamics of the clouds by using a sophisticated spatiotemporal function. On the contrary, the Farneback/Farne-PER only uses a fine-tuned quadratic polynomial function to approximate the cloud movement of last two observations. The Farneback/Farne-PER model shows closer performances to the LSTM/LSTM-PER over the state of Oregon compared to the results from the other two states, due to advection-dominant precipitation occurring over the state of Oregon. The Farneback/Farne-PER forecasts have less similar performances compared to the LSTM/LSTM-PER due to complex evolving nature of the precipitation clouds over the states of Oklahoma and Florida, which cannot be captured properly by Optical flow techniques. Based on the statistics from Figures 5–7, the LSTM/LSTM-PER method is more robust and powerful than the RNN/RNN-PER model because of two primary reasons. First, the LSTM has a specific designed structure to fully utilize the training information within the data set. The gates in the LSTM memory block allow the model to learn and update the memory in an efficient way (Wu et al., 2015). It is noteworthy that the efficient structure of the LSTM enabled the model to learn complex dynamics of the convective-type precipitation over the states of Oklahoma and Florida. Second advantage of LSTMs over RNNs is the fact that the LSTM model has a higher capability of dealing with time-dependent data and maintaining the forecast skill of predictions for a longer lead time as compared to the baseline RNN model. Similar argument was also made by Hochreiter (1998) that RNNs cannot benefit from the further past information to increase the model performance due to the gradient vanishing/exploding. Gers et al. (2000) also concluded that the LSTM model has a characteristic of Constant Error Carousel, which is able to prevent the gradient vanishing problem by keeping the local error backflow constant in the absence of new input or error signal. In Figures 5–7, the RNN-PER model tends to predict significant false rainy pixels, which results in having higher POD values and higher FAR values at the same time. However, the performances of the RNN-PER in terms of CSI is low due to high false positive and missed values.

5.2. Forecasting Uncertainties

The uncertainty of precipitation forecasts from the presented experiments comes from two sources: (1) Errors from the forecast algorithms in predicting the CTBT images for future time-steps and (2) errors introduced by the PERSIANN algorithm when estimating rainfall intensity using the predicted CTBT images. The uncertainties in forecasting the CTBT data grow by the increase in lead time, due to aggregation of errors from previous forecasts (which is the input to predict the next time-step CTBT). The error aggregation is valid for all the investigated extrapolation-based models (Persistency, Farneback, RNN, and LSTM). However, due to stochasticity of Neural Networks, including the RNN and LSTM models, another source of error is the performances of models with the parameter sets obtained from stochastic optimization. The error bars in Figures 5–7 addresses the uncertainties associated with training of the RNN/RNN-PER and LSTM/LSTM-PER. The LSTM/LSTM-PER, in general, shows smaller ranges of error in almost all the case studies, which indicates easier convergence of the LSTM/LSTM-PER compared to the RNN/RNN-PER model because of easier training scheme. In addition, the RNN/RNN-PER errors grow exponentially with the increase in lead time; however, the LSTM/LSTM-PER model seems to aggregate errors slowly (Figures 5–7). The differences in error growth rate of LSTM/LSTM-PER and RNN/RNN-PER models indicate the higher capabilities of the LSTM model in its learning and recalling useful information.

The second source of uncertainties comes from the accuracy of rainfall estimations by the PERSIANN algorithm. The performance of PERSIANN algorithm lies in (1) the capabilities of the model in translating CTBT information into rainfall intensities and (2) the amount of useful information that the CTBT images provide for estimating the rainfall. The performances of the PERSIANN model with respect to the statistical metrics of precipitation is presented on the lead time of zero in Figures 5–7, panels c–g.

6. Summary and Conclusion

In this study, we proposed a framework to forecast precipitation in a short term (0–6 hr). The proposed framework consists of an advanced deep learning model (termed LSTM) to forecast continuous CTBT images, and a precipitation estimation algorithm (termed as PERSIANN algorithm) to obtain the forecasted rain rates. Three case studies are investigated over the CONUS, including the states of Florida, Oregon, and Oklahoma. In the first part of the evaluation of forecasting skills, the results from our proposed model (LSTM/LSTM-PER) were compared with a number of baseline models, including the Persistency/Persist-PER, Farneback/Farne-PER, RNN/RNN-PER, and RAPv1.0. Better statistics (CTBT RMSE, CTBT CC, rainfall RMSE, rainfall CC, POD, FAR, and CSI) are observed with the results from LSTM/LSTM-PER as comparing other extrapolation-based and numerical methods. In the second phase of evaluation, the visual comparison of event-based precipitation forecasts from Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER were demonstrated. The visual comparisons of LSTM-PER model showed higher similarities compared to the other investigated models. Specific conclusions are listed below:

- According to our experiments, we found the proposed LSTM combined with the PERSIANN system is able to generate accurate initial forecasts for severe and even low-rate precipitation events in up to 6 hr.
- ANNs, in specific advanced RNNs, are useful tools in support of forecasting complex precipitation in short range (0–6 hr), particularly for capturing the patterns of convective precipitation systems. In details, the proposed framework (i.e., the combination uses of LSTM method with PERSIANN algorithm) demonstrated higher forecasting capabilities comparing to some commonly used storm tracking and prediction methods, such as the RNN, and the advection flow method in our employed case studies.
- The advantages of our proposed forecasting model (i.e., LSTM) relies on the recurrent layer, which can learn the patterns of precipitation events better than the traditionally used RNN, the Farneback advection flow model, and Persistency method. And the prediction accuracy of LSTM algorithm is higher any of those in general based on the case studies in this work.
- As one type of tools to predict precipitation, extrapolation-based methods, in general, will give high prediction accuracy in short lead time, and the forecast skills will drop quickly as the lead time increases. Differs from the extrapolation-based methods, NWP models, which rely on the physical process of rainfall formation, tend to have lower prediction skills with a short prediction lead time. However, the forecast skills will remain consistent and stable comparing to extrapolation-based methods employed in this study.
- The uncertainties of our proposed framework come from either the forecast modeling itself as the LSTM is not end-to-end in this study or the process of estimate rain rates from CTBT images (the uncertainties associated with the PERSIANN system).
- Last, the proposed deep learning framework (i.e., the LSTM models), while acting alone, has some levels of redundancies to learn the spatiotemporal variabilities of an event. Future investigation is suggested to jointly use a spatial classification technique and the LSTM layers for data with high spatiotemporal variabilities. This combination can be developed by using autoencoders with LSTM layer at its most encoded layer as suggested in some other studies from the literature.

Acknowledgments

The financial support of this research is from U.S. Department of Energy (DOE Prime Award DE-IA0000018), California Energy Commission (CEC Award 300-15-005), NSF CyberSEES Project (Award CCF-1331915), NOAA/NESDIS/NCDC (Prime awards NA09NES4400006 and NCSU CICS and subaward 2009-1380-01), the U.S. Army Research Office (award W911NF-11-1-0422), and China's Key projects for international cooperation in science and technology innovation between governments (2016YFE0102400). Authors also thank Charless Fowlkes at the University of California Irvine, Department of Computer Sciences, for providing suggestions and help in designing the study. The GOES-IR data set was obtained from the repository of Climate Prediction Center (CPC; http://www.cpc.ncep.noaa.gov/products/global_precip/html/wpage.merged_IR.html). The Q2 radar data are listed in the references, and RAPv1.0 forecasts are available at NOAA's Rapid Refresh website (https://rapidrefresh.noaa.gov/).

References

AghaKouchak, A., Behrangi, A., Sorooshian, S., Hsu, K., & Amitai, E. (2011). Evaluation of satellite-retrieved extreme precipitation rates across the central United States. *Journal of Geophysical Research*, *116*, D02115. https://doi.org/10.1029/2010JD014741

Arkin, P. A., Joyce, R., & Janowiak, J. E. (1994). The estimation of global monthly mean rainfall using infrared satellite data: The GOES precipitation index (GPI). *Remote Sensing Reviews*, *11*(1–4), 107–124. https://doi.org/10.1080/02757259409532261

Ballard, S. P., Li, Z., Simonin, D., & Caron, J. F. (2016). Performance of 4D-Var NWP-based nowcasting of precipitation at the met Office for summer 2012. *Quarterly Journal of the Royal Meteorological Society*, *142*(694), 472–487. https://doi.org/10.1002/qj.2665

Behrangi, A., Hsu, K.-l., Imam, B., Sorooshian, S., Huffman, G. J., & Kuligowski, R. J. (2009). PERSIANN-MSA: A precipitation estimation method from satellite-based multispectral analysis. *Journal of Hydrometeorology*, *10*(6), 1414–1429. https://doi.org/10.1175/2009JHM1139.1

Benjamin, S. G., Weygandt, S. S., Brown, J. M., Hu, M., Alexander, C. R., Smirnova, T. G., et al. (2016). A North American hourly assimilation and model forecast cycle: The rapid refresh. *Monthly Weather Review*, *144*(4), 1669–1694. https://doi.org/10.1175/MWR-D-15-0242.1

Bližňák, V., Sokol, Z., & Zacharov, P. (2017). Nowcasting of deep convective clouds and heavy precipitation: Comparison study between NWP model simulation and extrapolation. *Atmospheric Research*, *184*, 24–34. https://doi.org/10.1016/j.atmosres.2016.10.003

Bright, D. R., & Mullen, S. L. (2002). Short-range ensemble forecasts of precipitation during the southwest monsoon. *Weather and Forecasting*, *17*(5), 1080–1100. https://doi.org/10.1175/1520-0434(2002)017<1080:SREFOP>2.0.CO;2

Byeon, W., Breuel, T. M., Raue, F., & Liwicki, M. (2015). Scene labeling with lstm recurrent neural networks (pp. 3547–3555).

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Connor, J. T., Martin, R. D., & Atlas, L. E. (1994). Recurrent neural networks and robust time series prediction. *IEEE Transactions on Neural Networks*, *5*(2), 240–254. https://doi.org/10.1109/72.279188

Eck, D., & Schmidhuber, J. (2002). A first look at music composition using LSTM recurrent neural networks. Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale 103.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1

Faridzad, M., Yang, T., Hsu, K., Sorooshian, S., & Xiao, C. (2018). Rainfall frequency analysis for ungauged regions using remotely sensed precipitation information. *Journal of Hydrology*, *563*, 123–142. https://doi.org/10.1016/j.jhydrol.2018.05.071

Farnebäck, G. (2003). Two-frame motion estimation based on polynomial expansion. *Image Analysis*, 363–370. https://doi.org/10.1007/3-540-45103-X_50

Foresti, L., Reyniers, M., Seed, A., & Delobbe, L. (2016). Development and verification of a real-time stochastic precipitation nowcasting system for urban hydrology in Belgium. *Hydrology and Earth System Sciences*, *20*(1), 505–527. https://doi.org/10.5194/hess-20-505-2016

Fovell, R. G. (1997). Consensus clustering of US temperature and precipitation data. *Journal of Climate*, *10*(6), 1405–1427. https://doi.org/10.1175/1520-0442(1997)010<1405:CCOUST>2.0.CO;2

French, M. N., Krajewski, W. F., & Cuykendall, R. R. (1992). Rainfall forecasting in space and time using a neural network. *Journal of Hydrology*, *137*(1–4), 1–31. https://doi.org/10.1016/0022-1694(92)90046-X

Ganguly, A. R., & Bras, R. L. (2003). Distributed quantitative precipitation forecasting using information from radar and numerical weather prediction models. *Journal of Hydrometeorology*, *4*(6), 1168–1180. https://doi.org/10.1175/1525-7541(2003)004<1168:DQPFUI>2.0.CO;2

Gers, F. A., & Schmidhuber, J. (2000). Recurrent nets that time and count (pp. 189–194). IEEE.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471. https://doi.org/10.1162/089976600300015015

Golding, B. (1998). Nimrod: A system for generating automated very short range forecasts. *Meteorological Applications*, *5*(1), 1–16. https://doi.org/10.1017/S1350482798000577

Graves, A. (2013) Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850.

Graves, A., Mohamed, A.-r., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks (pp. 6645–6649). IEEE.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, *18*(5-6), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924

Guo, J. (2013). Backpropagation through time.

Hall, T., Brooks, H. E., & Doswell, C. A. III (1999). Precipitation forecasting using a neural network. *Weather and Forecasting*, *14*(3), 338–345. https://doi.org/10.1175/1520-0434(1999)014<0338:PFUANN>2.0.CO;2

Heye, A., Venkatesan, K., & Cain, J. (2017). Precipitation nowcasting: Leveraging deep rcurrent convolutional neural networks.

Higgins, R., Mo, K., & Yao, Y. (1998). Interannual variability of the US summer precipitation regime with emphasis on the southwestern monsoon. *Journal of Climate*, *11*(10), 2582–2606. https://doi.org/10.1175/1520-0442(1998)011<2582:IVOTUS>2.0.CO;2

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *6*(2), 107–116. https://doi.org/10.1142/S0218488598000094

Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). *Gradient flow in recurrent nets: The difficulty of learning long-term dependencies. A field guide to dynamical recurrent neural networks*. Hoboken, NJ: IEEE Press.

Hochreiter, S., & Schmidhuber, J. (1997a). *LSTM can solve hard long time lag problems* (p. 473). Cambridge, MA: MIT Press.

Hochreiter, S., & Schmidhuber, J. (1997b). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hsu, K.-l., Gao, X., Sorooshian, S., & Gupta, H. V. (1997). Precipitation estimation from remotely sensed information using artificial neural networks. *Journal of Applied Meteorology*, *36*(9), 1176–1190. https://doi.org/10.1175/1520-0450(1997)036<1176:PEFRSI>2.0.CO;2

Hsu, K. l., Gupta, H. V., Gao, X., Sorooshian, S., & Imam, B. (2002). Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis. *Water Resources Research*, *38*(12), 1302. https://doi.org/10.1029/2001WR000795

Jordan, M. I. (1997). Serial order: A parallel distributed processing approach. *Advances in Psychology*, *121*, 471–495. https://doi.org/10.1016/S0166-4115(97)80111-2

Katiraie-Boroujerdy, P.-S., Nasrollahi, N., Hsu, K.-l., & Sorooshian, S. (2013). Evaluation of satellite-based precipitation estimation over Iran. *Journal of Arid Environments*, *97*, 205–219. https://doi.org/10.1016/j.jaridenv.2013.05.013

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*(1), 59–69. https://doi.org/10.1007/BF00337288

Kuligowski, R. J., & Barros, A. P. (1998b). Experiments in short-term precipitation forecasting using artificial neural networks. *Monthly Weather Review*, *126*(2), 470–482. https://doi.org/10.1175/1520-0493(1998)126<0470:EISTPF>2.0.CO;2

Kuligowski, R. J., & Barros, A. P. (1998a). Localized precipitation forecasts from a numerical weather prediction model using artificial neural networks. *Weather and Forecasting*, *13*(4), 1194–1204. https://doi.org/10.1175/1520-0434(1998)013<1194:LPFFAN>2.0.CO;2

Lakshmanan, V., Fritz, A., Smith, T., Hondl, K., & Stumpf, G. (2007). An automated technique to quality control radar reflectivity data. *Journal of Applied Meteorology and Climatology*, *46*(3), 288–305. https://doi.org/10.1175/JAM2460.1

Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019.

Liu, X., Yang, T., Hsu, K., Liu, C., & Sorooshian, S. (2017). Evaluating the streamflow simulation capability of PERSIANN-CDR daily rainfall products in two river basins on the Tibetan plateau. *Hydrology and Earth System Sciences*, *21*(1), 169–181. https://doi.org/10.5194/hess-21-169-2017

Mao, J., Xu, W., Yang, Y., Wang, J., & Yuille, A. L. (2014). Explain images with multimodal recurrent neural networks. arXiv preprint arXiv:1410.1090.

Moazami, S., Golian, S., Kavianpour, M. R., & Hong, Y. (2013). Comparison of PERSIANN and V7 TRMM Multi-satellite Precipitation Analysis (TMPA) products with rain gauge data over Iran. *International Journal of Remote Sensing*, *34*(22), 8156–8171. https://doi.org/10.1080/01431161.2013.833360

Nam, D. H., Mai, D. T., Udo, K., & Mano, A. (2014). Short-term flood inundation prediction using hydrologic-hydraulic models forced with downscaled rainfall from global NWP. *Hydrological Processes*, *28*(24), 5844–5859. https://doi.org/10.1002/hyp.10084

Ritter, B., & Geleyn, J.-F. (1992). A comprehensive radiation scheme for numerical weather prediction models with potential applications in climate simulations. *Monthly Weather Review*, *120*(2), 303–325. https://doi.org/10.1175/1520-0493(1992)120<0303:ACRSFN>2.0.CO;2

Robertson, D., Shrestha, D., & Wang, Q. (2013). Post-processing rainfall forecasts from numerical weather prediction models for short-term streamflow forecasting. *Hydrology and Earth System Sciences*, *17*(9), 3587–3603. https://doi.org/10.5194/hess-17-3587-2013

Romilly, T. G., & Gebremichael, M. (2011). Evaluation of satellite rainfall estimates over Ethiopian river basins. *Hydrology and Earth System Sciences*, *15*(5), 1505–1514. https://doi.org/10.5194/hess-15-1505-2011

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation, DTIC Document.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. *Cognitive Modeling*, *5*(3), 1.

Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2015) Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems 28*. Curran Associates, Inc.

Shi, X., Zhourong, C., Hao, W., Dit-Yan, Y., Wai-kin, W., & Wang-chun, W. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting (pp. 802–810).

Shrestha, D., Robertson, D., Wang, Q., Pagano, T., & Hapuarachchi, H. (2013). Evaluation of numerical weather prediction model precipitation forecasts for short-term streamflow forecasting purpose. *Hydrology and Earth System Sciences*, *17*(5), 1913–1931. https://doi.org/10.5194/hess-17-1913-2013

Srivastava, N., Mansimov, E., 7amp; Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms, pp. 843–852.

Sun, J., Xue, M., Wilson, J. W., Zawadzki, I., Ballard, S. P., Onvlee-Hooimeyer, J., et al. (2014). Use of NWP for nowcasting convective precipitation: Recent progress and challenges. *Bulletin of the American Meteorological Society*, *95*(3), 409–426. https://doi.org/10.1175/BAMS-D-11-00263.1

Sundermeyer, M., Schlüter, R., & Ney, H. (2012). LSTM Neural networks for language modeling (pp. 194–197) .

Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for. *Machine Learning*, *4*(2).

Vasiloff, S. V., Howard, K. W., Rabin, R. M., Brooks, H. E., Seo, D.-J., Zhang, J., et al. (2007). Improving QPE and very short term QPF: An initiative for a community-wide integrated approach. *Bulletin of the American Meteorological Society*, *88*(12), 1899–1911. https://doi.org/10.1175/BAMS-88-12-1899

Wallace, J. M. (1975). Diurnal variations in precipitation and thunderstorm frequency over the conterminous United States. *Monthly Weather Review*, *103*(5), 406–419.

Wang, G., Yang, J., Wang, D., & Liu, L. (2016). A quantitative comparison of precipitation forecasts between the storm-scale numerical weather prediction model and auto-nowcast system in Jiangsu, China. *Atmospheric Research*, *181*, 1–11. https://doi.org/10.1016/j.atmosres.2016.06.004

Warner, T. T., Peterson, R. A., & Treadon, R. E. (1997). A tutorial on lateral boundary conditions as a basic and potentially serious limitation to regional numerical weather prediction. *Bulletin of the American Meteorological Society*, *78*(11), 2599–2617. https://doi.org/10.1175/1520-0477(1997)078<2599:ATOLBC>2.0.CO;2

Werbos, P. J. (1988). Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, *1*(4), 339–356. https://doi.org/10.1016/0893-6080(88)90007-X

Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, *1*(2), 270–280. https://doi.org/10.1162/neco.1989.1.2.270

Wu, Z., Wang, X., Jiang, Y.-G., Ye, H., & Xue, X. (2015). Modeling spatial-temporal clues in a hybrid deep learning framework for video classification (pp. 461–470). ACM.

Xu, L., Gao, X., Sorooshian, S., Arkin, P. A., & Imam, B. (1999). A microwave infrared threshold technique to improve the GOES precipitation index. *Journal of Applied Meteorology*, *38*(5), 569–579. https://doi.org/10.1175/1520-0450(1999)038<0569:AMITTT>2.0.CO;2

Yang, T., Asanjan, A. A., Faridzad, M., Hayatbini, N., Gao, X., & Sorooshian, S. (2017). An enhanced artificial neural network with a shuffled complex evolutionary global optimization with principal component analysis. *Information Sciences*, *418*, 302–316.

Yang, T., Asanjan, A. A., Welles, E., Gao, X., Sorooshian, S., & Liu, X. (2017). Developing reservoir monthly inflow forecasts using artificial intelligence and climate phenomenon information. *Water Resources Research*, *53*, 2786–2812. https://doi.org/10.1002/2017WR020482

Yang, T., Tao, Y., Li, J., Zhu, Q., Su, L., He, X., & Zhang, X. (2017). Multi-criterion model ensemble of CMIP5 surface air temperature over China. *Theoretical and Applied Climatology*, 1–16.

Zahraei, A., Hsu, K.-l., Sorooshian, S., Gourley, J., Lakshmanan, V., Hong, Y., & Bellerby, T. (2012). Quantitative precipitation nowcasting: A Lagrangian pixel-based approach. *Atmospheric Research*, *118*, 418–434. https://doi.org/10.1016/j.atmosres.2012.07.001

Zahraei, A., Hsu, K. L., Sorooshian, S., Gourley, J. J., Hong, Y., & Behrangi, A. (2013). Short-term quantitative precipitation forecasting using an object-based approach. *Journal of Hydrology*, *483*, 1–15. https://doi.org/10.1016/j.jhydrol.2012.09.052

Zhang, J., Howard, K., Langston, C., Vasiloff, S., Kaney, B., Arthur, A., et al. (2011). National Mosaic and Multi-Sensor QPE (NMQ) system: Description, results, and future plans. *Bulletin of the American Meteorological Society*, *92*(10), 1321–1338. https://doi.org/10.1175/2011BAMS-D-11-00047.1

Zhu, Q., Hsu, K.-l., Xu, Y.-P., & Yang, T. (2017). Evaluation of a new satellite-based precipitation data set for climate studies in the Xiang River basin, southern China. *International Journal of Climatology*, *37*(13), 4561–4575. https://doi.org/10.1002/joc.5105