UNIVERSITY OF CALIFORNIA,
IRVINE

An Advanced Deep Learning Framework For Short-Term Precipitation Forecasting From Satellite Information

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Civil and Environmental Engineering

by

Ata Akbari Asanjan

Dissertation Committee:
Professor Soroosh Sorooshian, Co-Chair
Professor Kuolin Hsu, Co-Chair
Professor Charless Folkwes

2019

DEDICATION

To

My wife, Samin

My mother, Vajiheh

My uncle, Yakup

and all my loved ones in recognition of their worth, love and efforts which made me who I am today.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Undertaking this PhD has been a truly life-changing experience for me and it would not have been possible to do without the support and guidance that I received from many people.

First and foremost, I would like to sincerely thank my advisor Professor Soroosh Sorooshian, who provided me the opportunity to join the Center for Hydrometeorology and Remote Sensing (CHRS) at University of California, Irvine. He supported me throughout my Ph.D. years and taught me many important lessons among which are creative and out-of-the-box thinking, not only hard but effective working attitude.

I am also in debt to my co-advisors Professors Kuolin Hsu and Xiaogang Gao who helped me to opened my eyes to the extent of research in the field, and their advice accompanied me through different stages of my research.

I also want to thank my mentors Professor Tiantian Yang and Pari-Sima Katiraie-Boroujerdy for their continuous encouragement, patience and help.

CURRICULUM VITAE

Ata Akbari Asanjan

EDUCATION

University of California, Irvine

Ph.D. in Civil Engineering (Water Resources) June 2019

M.S. in Civil Engineering (Water Resources) March 2016

University of Tabriz, Iran

B.S. in Civil Engineering (Water Resources) August 2014

TECHNICAL SKILLS

Computer Languages	Python, MATLAB, SQL, R, Latex
Geospatial Tools	GDAL, Arc-GIS, ENVI, QGIS, NEX
Computer Vision Processing	TensorFlow, Theano, Keras, PyTorch, Scikit-Learn, OpenCV
Cloud-based Computing	AWS, Google Kubernetes Engine, HPC
Operating System	Unix, Windows, Macintosh
Tools	VNC, PuTTY, Cygwin, Vim, Emacs
Relevant Courses	Machine Learning, Image Understanding, Neural Networks and Deep Learning, Algorithm Design

EXPERIENCE

NASA Ames, Mountain View, CA June 2018 - Sept. 2018

Data Science Intern

Designed multiple deep neural networks to classify and segment coral reef categories using remote sensing data.

- Mosaiced and preprocessed coral reef datasets from Sentinel and WorldView satellites.
- Implemented a novel deep learning models combining a Laplacian Pyramid consist of

Generative Adversarial Networks and a Domain Adaptive Neural Network termed as LAPDANN in Python/TensorFlow.

- Significantly increased the high-resolution classification accuracy (by 11 percents) using state-of-the-art model and data augmentation techniques.

University of California, Irvine, Irvine, CA March 2016 - Present

Graduate Researcher

Built a deep learning framework for short-term precipitation forecasting.

- Preprocessed geophysical big data from GOES satellites and weather surveillance radar.
- Implemented advanced Neural Networks architecture including semi-conditional Generative Adversarial Networks (GAN) with Unet structure in Python.
- Built an advanced image-to-image translation GAN to estimate rainfall maps from long-wave infrared channel of GOES using Python.
- Installed Nvidia's Quadro P6000 GPU unit on local server machine to accelerate training process.

Implemented an evolutionary neural network.

- Developed a feedforward neural network using the Shuffle Complex evolutionary optimization algorithm to find optimal solutions.

Analyzed the precipitation uncertainty using statistical methods.

- Collaborated with fellow researchers to assess the uncertainties of three satellite precipitation product without observation using a statistical method termed Generalized Three-Cornered Hat.

University of California, Irvine, Irvine, CA Sept. 2015 - Present

Teaching Assistant

- Mentored more than 80 undergraduate and graduate students to ensure they learned fundamental concepts of hydrology.
- Coached students individually resulting in increased comprehension of materials and improved performance.
- Evaluated performance and provided feedback for over 80 students.

University of California, Irvine, Irvine, CA Sept. 2014 - March 2016

Master's Thesis

Feature Selection of PERSIANN, based on Multiple Regression Analysis with Principal Component Analysis

- Implemented Self-Organizing Linear Output (SOLO) map in MATLAB and Python.
- Designed an architecture using Multiple Linear Regression and Principal Component Analysis to solve the collinearity.
- Ranked inputs of the PERSIANN algorithm based on their importance and contribution to the outcome results.

SELECTED PROJECTS

Rainfall Sequence Prediction: Forecasted rainfall sequences using Self-Organized Feature Map structure to classify the spatiotemporal characteristics of precipitation.

Reservoir Time-series Forecast: Compared the performances of Support Vector Machines, Decision Trees and Artificial Neural Networks in forecasting monthly reservoir inflows.

Rainfall Estimation: Designed a deep learning model architecture consisting of Generative Adversarial networks and Convolutional autoencoder to estimate the rainfall intensity based on satellite infrared imagery.

PUBLICATIONS

- **Asanjan, A. A.**, Das K., Li A., Chirayath V., Torres-Perez J., Sorooshian S. (2019) *"Learning instrument invariant characteristics for generating high resolution global coral reef maps"* International Joint Conference on Artificial Intelligence (Under Review)

- Katiraie-Boroujerdy, P. S., **Asanjan, A. A.**, Chavishian A., Hsu, K. L., and Sorooshian, S. (2019). *"Assessment of Seven CMIP5 Model Precipitation Extremes Over Iran Based on a Satellite-Based Climate Dataset"* International Journal of Climatology

- **Asanjan, A. A.**, Yang T., Hsu, K. L., Sorooshian, S., Lin, J. Q. and Peng, Q. (2018). *"Short-term Precipitation Forecast based on the PERSIANN system and the Long Short-Term Memory (LSTM) Deep Learning Algorithm."* Journal of Geophysical Research - Atmospheres.

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

- Katiraie-Boroujerdy, P. S., **Asanjan, A. A.**, Hsu, K. L., and Sorooshian, S. (2017). *"Intercomparison of PERSIANN-CDR and TRMM-3B42V7 precipitation estimates at monthly and daily time scales."* Atmospheric Research, 193, 36-49.

- Yang, T., **Asanjan, A. A.**, Welles, E., Gao, X., Sorooshian, S., and Liu, X. (2017).

Developing reservoir monthly inflow forecasts using artificial intelligence and climate phenomenon information." Water Resources Research, 53(4), 2786-2812.

INVITED PRESENTATIONS

- **Asanjan, A. A.**, Das, K., Li, A., Rozenhaimer, M. S., Torres-Perez, J. L., Chirayath, V. and Sorooshian, S. (2018) *"Classifying Multimodal Coral Reef Images Using Domain Adaptation."* AGU Fall meeting abstracts.

- **Asanjan, A. A.**, Yang, T. Gao, X. Hsu, K. L. and Sorooshian, S. (2017) *"Short-range quantitative precipitation forecasting using Deep Learning approaches."* AGU Fall meeting abstracts.

- **Asanjan, A. A.**, Yang, T. Gao, X. Hsu, K. L. and Sorooshian, S. (2016) *"Precipitation Nowcast using Deep Recurrent Neural Network."* AGU Fall meeting abstracts.

- **Asanjan, A. A.**, Yang, T. Gao, X. Hsu, K. L. and Sorooshian, S. (2017) *"Short-term Precipitation Forecasting using Recurrent Neural Network"* Big Data and the Earth Sciences: Grand Challenges Workshop, University of California, San Diego.

- **Asanjan, A. A.**, Yang, T. Gao, X. Hsu, K. L. and Sorooshian, S. (2017) *"Applications of Deep Learning in Short-range precipitation forecasting"* Southern California Machine Learning Symposium.

HONORS AND AWARDS

Outstanding Engineering Student Award February 2019

Orange County Engineering Council, CA, USA

NVIDIA's academic GPU seeding Grant February 2018

University of California, Irvine, CA, USA

Graduate Student Research and Travel Grant March 2018

University of California, Irvine, CA, USA

Big Data and the Earth Sciences: Grand Challenges May 2017

Student Grant recipient, University of California, San Diego, CA, USA

Associated Graduate Students February 2016

University of California, Irvine, CA, USA February 2017

ABSTRACT OF THE DISSERTATION

An Advanced Deep Learning Framework For Short-Term Precipitation Forecasting From Satellite Information

By

Ata Akbari Asanjan

Doctor of Philosophy in Civil and Environmental Engineering

University of California, Irvine, 2019

Professor Soroosh Sorooshian, Co-Chair
Professor Kuolin Hsu, Co-Chair

Short-term Quantitative Precipitation Forecasting is important for aviation and navigation safety control, flood forecasting, early flood warning, and natural hazard management. Obtaining accurate and timely precipitation forecasts in short range of time (0-6 hours) is a challenging task. Addressing the challenges in forecasting accurate short-term rainfall is an open question in the field of hydrometeorology and is a major objective. This dissertation introduces a machine learning, in specific deep learning, framework to accurately forecast high- and low-intensity precipitation events. In details, this dissertation introduces an effective precipitation forecasting framework by (1) developing an infrared cloud-top brightness temperature forecasting model, and (2) proposing an effective infrared to rainfall intensity mapping model using satellite and radar data. The proposed framework is effective due to (1) solving a physical problem using a continuous infrared data in which evolution is dominated by the continuity law of heat transfer, (2) providing forecasts for various ranges of rainfall intensities, and (3) introducing a framework with potentials to become a quasi-global scale product. As the initial attempt to develop the precipitation forecasting model, a forecasting model was proposed by extrapolating infrared imageries using an advanced Deep Neural Network (DNN) and applying the forecasted infrared into an effective rainfall retrieval algorithm to obtain the short-term precipitation forecasts. 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 LSTM combined with the PERSIANN were 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. Furthermore, to improve the forecasting skills of the proposed method, a new infrared forecasting method was developed by improving the LSTM model (such as efficient use of neighborhood pixel information, resolving the loss of resolution problem and introducing more efficient objectives compared to maximum likelihood estimates). The new proposed infrared forecasting algorithm is a semi-conditional Generative Adversarial Network (GAN) consisting of convolutional, recurrent (LSTM) and convolutional-recurrent (ConvLSTM) layers in order to forecast temporally and spatially coherent infrared images. The results are compared with the non-adversarial version of the proposed model and demonstrate superior performances. In addition to this precipitation forecasting improvement step, a new precipitation estimation algorithm is introduced to replace the PERSIANN algorithm in order to increase the infrared-rainfall translation accuracy and enable the framework to become an end-to-end model. The new precipitation estimation model is a conditional GAN, termed as PERSIANN-GAN, which translates $0.25^\circ \times 0.25^\circ$ infrared data into same resolution precipitation estimates by defining a more flexible objective function. The PERSIANN-GAN results are compared with two Convolutional Neural Networks (CNNs) baseline models one without adversarial part and with bypass connections and the other one without adversarial part and without bypass connections. The model performances were also compared to the well-known operational PERSIANN product. The comparison results demonstrate higher visual and statistical performances of PERSIANN-GAN.

Chapter 1

Introduction

1.1 Importance of short-term precipitation forecast

Weather forecasting refers to the focus of science and engineering dealing with state of the atmosphere in a given time in the future. Weather state prediction, depending on its application, can range from minutes to years and decades, and each range is associated with its various challenges and uncertainties. Among all the atmosphere components, precipitation is an important weather phenomenon and one of the most vital elements of the hydrological cycle, which affects human lives in many aspects. Yet, accurate precipitation forecasting is often the most elusive tasks because of the variability range of precipitation in space and time. Despite the difficulty of precipitation forecasting, 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 06 hr of lead time, are acute for naval and aviation navigation, flash flood warning, flood forecasting, and other hydrological applications ([Ganguly and Bras, 2003, Kuligowski and Barros, 1998, Liu et al., 2017, Vasiloff et al., 2007, Zhu et al., 2017].

This dissertation focuses on the development and evaluation of a Short-term Quantitative Precipitation Forecasting (SQPF) framework, also referred to as nowcasting, which enables quasi-global precipitation forecasting for the range of 0 to 6 hours. This framework is not only intended for high-intensity rainfall event, but also can capture rainfall patterns with various intensity and areal coverage with high accuracy. The described capability of the proposed framework, makes it superior to other existing model which cannot resolve small scale rainfalls.

1.2 Overview of short-term precipitation forecasting approaches

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 06 hr of lead time, are acute for flash flood warning, flood forecasting, and other hydrological applications [Ganguly and Bras, 2003, Kuligowski and Barros, 1998, Liu et al., 2017, Vasiloff et al., 2007, Zhu et al., 2017]. Previous studies have shown that the Numerical Weather Prediction (NWP) models and the extrapolation-based methods are frequently used in the short-term precipitation forecast ([Bright and Mullen, 2002, Golding, 1998] Kuligowski Barros, 1998a; Nam et al., 2014; Robertson et al., 2013; Zahraei et al., 2013).

1.2.1 Numerical Weather Prediction based short-term precipitation forecast

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 [Golding, 1998, Ritter and Geleyn, 1992, Warner et al., 1997, Yang et al., 2018, 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 NOAAs 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.

1.2.2 Remote sensing based short-term precipitation forecast

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 and Barros [1998] implemented a Three Layer Feed Forward Neural Networks (TLFFNN) 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.

1.3 Overview of precipitation estimation approaches

Rainfall measurement/estimation is an important source of information for water management and extreme weather preparation purposes. Various methods are utilized to measure/estimate rainfall intensity and duration. As the most simple and direct measurement strategy, rain gauge are the most trusted and primary way of gathering rainfall information. However, due to (1) the sparsity of rain gauge networks; (2) unavailability in the remote areas and oceans and (3) inability to gather rainfall information with high temporal resolution, the rain gauges are not an impractical source for high spatial and temporal resolution monitoring [Smith et al., 1996].

A different category of rainfall observation system is radars. Radars observe rain intensity using electromagnetic radiation back-scatter from water droplets. This observation mechanism allows radars to cover higher temporal and spatial variability of rainfall events. However, the radar beam access is limited, thus, resulting in limited coverage over a certain area. In order to compensate limited spatial access of radars, a number of them should be utilized in an overlapping network in order to cover a wide area of interest which is not an economically efficient approach. Another shortcoming of the radar systems is their inability to penetrate mountains resulting in even more limited coverage in the mountainous regions. Furthermore, radars are not available over the oceans. The radar networks also suffer from attenuation, beam-filling, and beam overshoot [Sauvageot, 1994].

Due to the ineffectiveness of ground-based observations, satellite-based technologies were introduced as an alternative source of information. Various satellites with wide range of sensor spectrums have been developed and launched to monitor the state of the atmosphere. At the current stage of satellite-based precipitation estimation development, two sensor types are mainly utilized: (1) Infrared sensors on Geostationary Earth Orbit (GEO) and Low-Earth Orbiting (LEO) satellites and (2) Passive microwave (PMW) sensors and active instruments on LEO satellites. PMW data have robust physichal relation with hydrometeorological phenomena over the atmosphere, resulting in accurate estimation of precipitation. However, due to limited coverage of PMW sensors and their unavailability over the same location with fine temporal gaps, make them a less desirable choice compared to GEO satellites. GEO satellites orbit Earth in a geosynchronous orbit allowing the sensors to continuously monitor a fixed location from 36000 km above the Earth. Thus, making GEO satellites a valuable source of information with high temporal (30 minutes or finer in the new generations) and spatial resolution (0.04° or finer in the new generations). Several studies such as [Arkin, 1979, Arkin and Meisner, 1987] pointed out that infrared information of GEO satellites between 200 K to 260 K have high correlations with rainfall rates. [Arkin et al., 1994] stated that due to frequent visits, high spatial resolution and significant correlations with rainfall intensities of GEO satellites, there are the most dominant source of information for precipitation forecasts and estimations.

1.4 Research motivation and objectives

Machine learning research, especially deep learning, is very active and fast-paced area of research with wide range of applicability. Recent advances in deep learning, in particular, state-of-the-art video forecasting and image to image translation techniques can be leveraged in the field of Earth system sciences and hydrometeorological applications. This dissertation is dedicated to explore the applicability of state-of-the-art deep learning approaches to obtain timely precipitation forecasts and accurate rainfall estimation using infrared imageries from GEO satellites. Specifically, this dissertation addresses the following objectives:

- Investigating the effectiveness of deep learning approaches for extracting spatial and temporal features from image sequences of satellite data and obtaining accurate and timely infrared forecasts.

- Demonstrating the role of Recurrent Neural Networks and its recent variants (Long Short-Term Memory) in forecasting accurate infrared imageries.

- Evaluate the proposed methodology and the effectiveness of using new deep learning techniques compared to earlier generations.

- Introduce a new approach for image to image translation by reconsidering the use of maximum likelihood estimation and developing a more flexible approach in generative neural networks.

- Assess the improvements of new generative models in infrared forecasting task and explore the capability of sub-location learning in generalizing learnt features.

- Introduce a new framework to translate infrared imageries to rainfall intensity maps using a network learning from distribution and maximum likelihood of the data together.

1.5 Dissertation outline

This dissertation is organized as follows: Chapter 2 provides a review on the applications of Recurrent Neural Networks (RNNs) and their recent modified variant, termed as Long Short-Term Memory (LSTM) along with popular short-term forecasting methods such as Optical flow methods and Numerical Weather Prediction (NWP) models. Chapter 3 details the shortcomings of maximum likelihood estimation in the context of generative neural networks and introduces a new deep learning approach and discusses the details of the method and the progress of the state-of-the-art methods. In addition, the chapter proposes two main architecture of neural networks for efficient and effective image to image translations.

Chapter 4 aims to tackle the bottlenecks inherited by the introduced approach in chapter 2 and introduces a conditional generative model partially discussed in chapter 3. The method demonstrates the advantage of mixing convolutional layers with recurrent layers.

Chapter 5 introduces a conditional generative model adapted from the methods discussed in chapter 3 to estimate rainfall rates from InfraRed (IR) imageries obtained from Geostationary Operational Environmental Satellites (GOES). This chapter discusses the methodology and the improvements achieved from the proposed method and compared them to Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN).

Finally, chapter 6 presents the conclusions, summary and foreseened future directions.

Chapter 2

Short-Term precipitation forecasting based on the PERSIANN system and LSTM

2.1 Introduction

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 and 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 [1997b] 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 and Zipser, 1989, Yang et al., 2017a,b]. Despite the improvements in training procedure of RNNs, they cannot capture the temporal features especially the long-term dependencies [Hochreiter et al., 2001, Hochreiter and Schmidhuber, 1997b]. 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 and Schmidhuber [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 and 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 percent 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. Xingjian 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 (06 hr). Heye et al. [2017] used a similar approach to that used by Xingjian 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 homogeneous 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 (60N to 60S). 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.

2.2 Methodology

2.2.1 Elman-type Recurrent Neural Networks

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 2.1) 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 creating an inner loop. The hidden layer is also connected to the output layer to forecast the next time step (Figure 2.1).

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 $x = (x_1, \ldots, x_T)$ to represent the sequence of CTBT images from time step t = 1 to T, respectively. As shown in 2.1, the normalized input image x_t is fed into the model, and the RNN model computes the hidden vector sequence $h = (h_1, \ldots, h_T)$ by

$$h_t = \sigma(W_{ji}x_t + U_{jj}h_{t-1} + b_j) \tag{2.1}$$

$$\sigma(\alpha) = (1 + exp(-\alpha))^{-1} \tag{2.2}$$

$i = 1, ..., n_0$ $\quad$ $n_0 =$ number of input nodes

$j = 1, ..., n_1$ $\quad$ $n_1 =$ number of hidden nodes

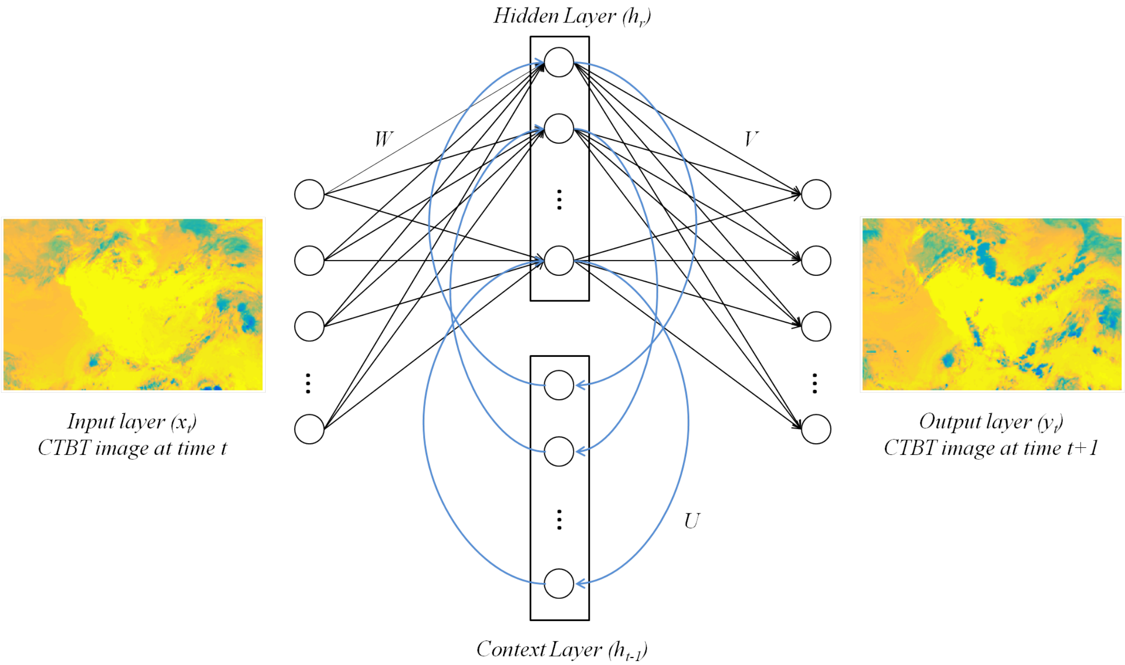


Figure 2.1: . The illustration of an Elman-type Three-Layer Recurrent Neural Network: x_t, h_t, h_{t-1}, and y_t represent the normalized input, hidden state, context state, and the output forecast at time t, respectively. W, U, and V are input-hidden, context-hidden, and hidden-output weights, respectively. CTBT = Cloud-Top Brightness Temperature.

where W is the input-hidden weight vector, U is the context unit weight vector, and b denotes the bias vector. The calculation of outputs is shown in the following 2.3:

$$y_t = V_{kj}h_t + b_k \tag{2.3}$$

where the vector $y = (y1, , yT)$ is the output of the model for time step $t = 1, 2, , T$; and V is the hidden-output weight vector.

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

$$E = \frac{\sum_{i=1}^{N}(y_i - o_i)^2}{N} \tag{2.4}$$

where E, o_i, and N are the output layer error term, target values of i^{th} 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.

2.2.2 Long Short-Term Memory (LSTM)

LSTM is a complex recurrent model developed by Hochreiter and Schmidhuber [1997a] 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., 1999], LSTM with peephole connections [Gers and 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. [1999], and a conceptual illustration of the implemented model is shown in the following Figure 2.2.

As it is shown in Figure 2.2, a LSTM block consists of an input gate, a forget gate, a memory cell, and an output gate. The following equations 2.5 to 2.9 represent the mathematics behind

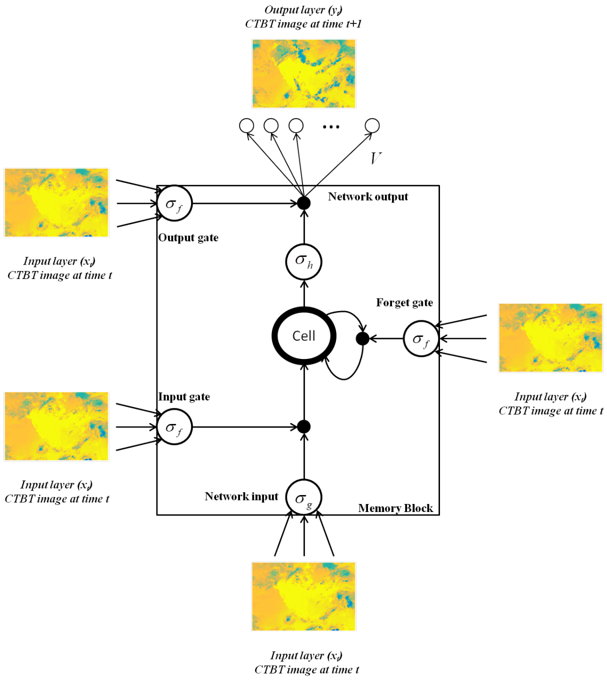


Figure 2.2: The conceptual illustration of employed LSTM: The x_t and y_t represent the normalized input and the output forecast at time t, respectively. The V represents the LSTM output weights. The Network, Input, Forget and Output gate and cell are the main components of the Memory Block. The σ_f, 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.

the LSTM Memory Block:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{2.5}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{2.6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_{hc}x_{t-1} + b_c) \tag{2.7}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}x_{t-1} + b_o) \tag{2.8}$$

$$h_t = o_t \odot tanh(c_t) \tag{2.9}$$

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 equations2.5-2.8 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 2.7 and 2.9 represents the inner product of matrices. Note that σ and $tanh$ in equations 2.5-2.9 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. [1999] 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.

2.2.3 Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN)

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 2.3). 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 and Gebremichael, 2011].

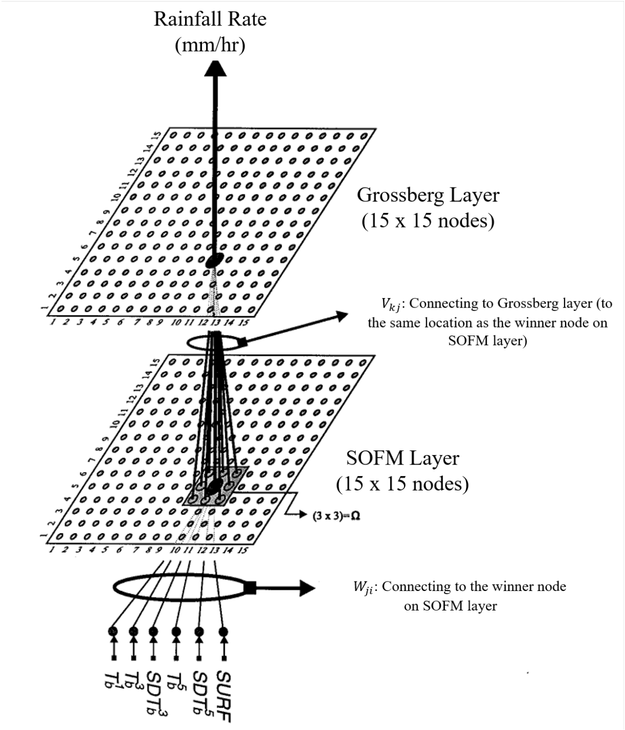


Figure 2.3: The simple structure of Precipitation Estimation from Remotely Sensed Information using Artificial Neural Network algorithm (Hsu et al. [1997]). The bottom layer is the Self-Organizing Map, and the upper layer is the regression layer. The 15 15 is the size of hidden layer, and the neighborhood of 33 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 33 CTBT pixel window, standard deviation of CTBT in $3x3$ pixel window, mean of 55 CTBT pixel window, standard deviation of CTBT in 55 pixel window, and the index of land, coast, or ocean of the pixel.

2.2.4 Baseline Models

Farneback Dense Optical Flow

The Farneback dense optical flow method is used in this chapter 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:

$$f_1(x) = x^T A_1 x + b_1^T x + c_1 \tag{2.10}$$

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:

$$f_2(x) = f_1(x - d) \tag{2.11}$$

$$f_2(x) = (x - d)^T A_1 (x - d) + b_1^T (x - d) + c_1 \tag{2.12}$$

$$f_2(x) = x^T A_1 x + (b_1 - 2A_a d)^T x + d^T A_1 d - b_1^T d + c_1 \tag{2.13}$$

$$f_2(x) = x^T A_2 x + b_2^T x + c_2 \tag{2.14}$$

The key outcome of the above equilibrium is

$$b_2 = b_1 - 2A_1 d \tag{2.15}$$

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

$$d = -\frac{(b_2 - b_1)}{2A_1} \tag{2.16}$$

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.

2.2.5 Proposed Short-Term Precipitation Forecasting System

2.2.6 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

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y_i} - y_i)^2}{N}}, \quad i = 1, \cdots, N \tag{2.17}$$

$$CC = \frac{\sum_{i=1}^{N}((\hat{y_i} - \hat{\mu})(y_i - \mu))}{\sqrt{(\sum_{i=1}^{N}(\hat{y_i} - \hat{\mu})^2)(\sum_{i=1}^{N}(y_i - \mu)^2)}} \tag{2.18}$$

where $\hat{y}$; y; $\hat{\mu}$; 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 2.192.21 describe the indices, respectively.

$$POD = \frac{TP}{TP + MS} \tag{2.19}$$

$$FAR = \frac{FP}{TP + FP} \tag{2.20}$$

$$CSI = \frac{TP}{TP + FP + MS} \tag{2.21}$$

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.

2.3 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 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 2.4).

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 $0.04° \times 0.04°$, 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 $0.25° \times 0.25°$. 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

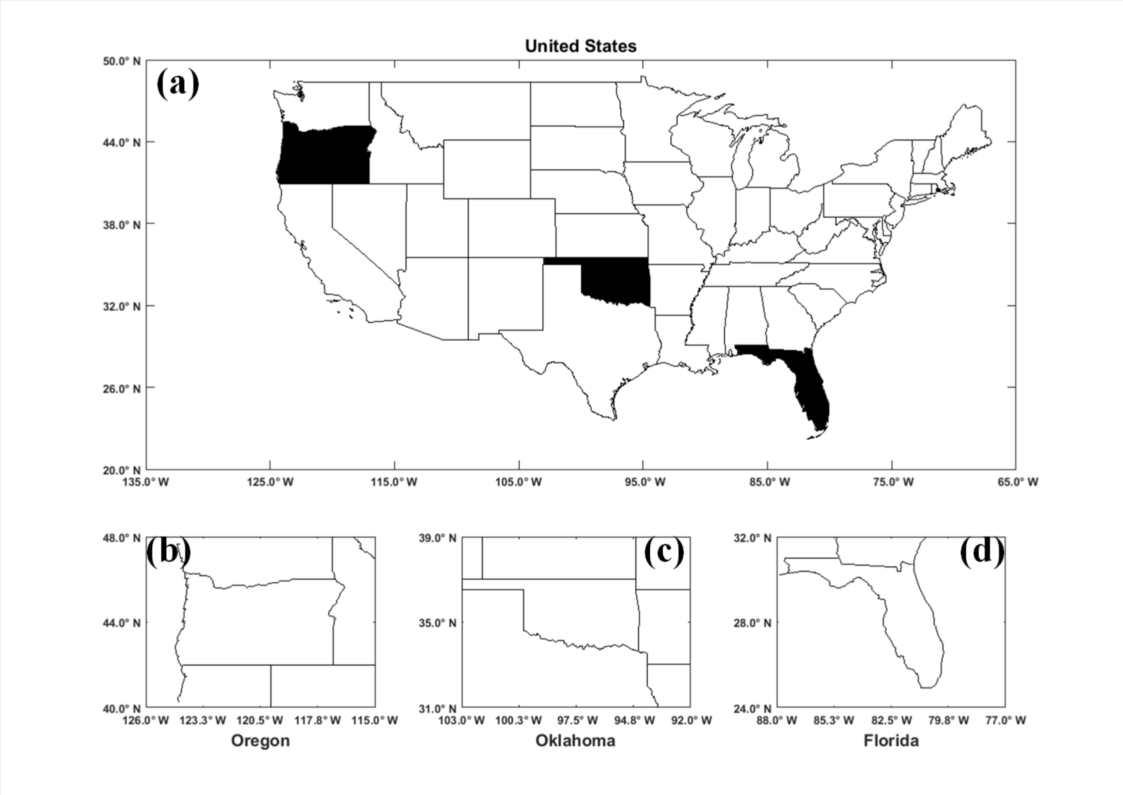


Figure 2.4: Contiguous United States map (a) and the selected study areas of the states of (b) Oregon, (c) Oklahoma, and (d) Florida.

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 $0.01° \times 0.01°$, respectively. In this study, the Q2 data were regridded to $0.25° \times 0.25°$ 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 beginning of this section (section 2.2.1). 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 and Hinton, 2012] optimizer with a learning rate of 0.002 and momentum of 0.9. Moreover, a 25 percents 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 2.2.2). 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 percents 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].

2.4 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 57) and (2) event-based visual comparison of precipitation forecasts over the case study regions (Figures 810).

2.4.1 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 2.52.7 for the states of Oregon, Oklahoma, and Florida, respectively. Figures 2.5a and 2.5b 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 2.5b, 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 2.5c-2.5g, 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 57, subplots cg, 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 percents 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 2.5c shows superior performance of the LSTM-PER model in terms of RMSE except for the first time step.

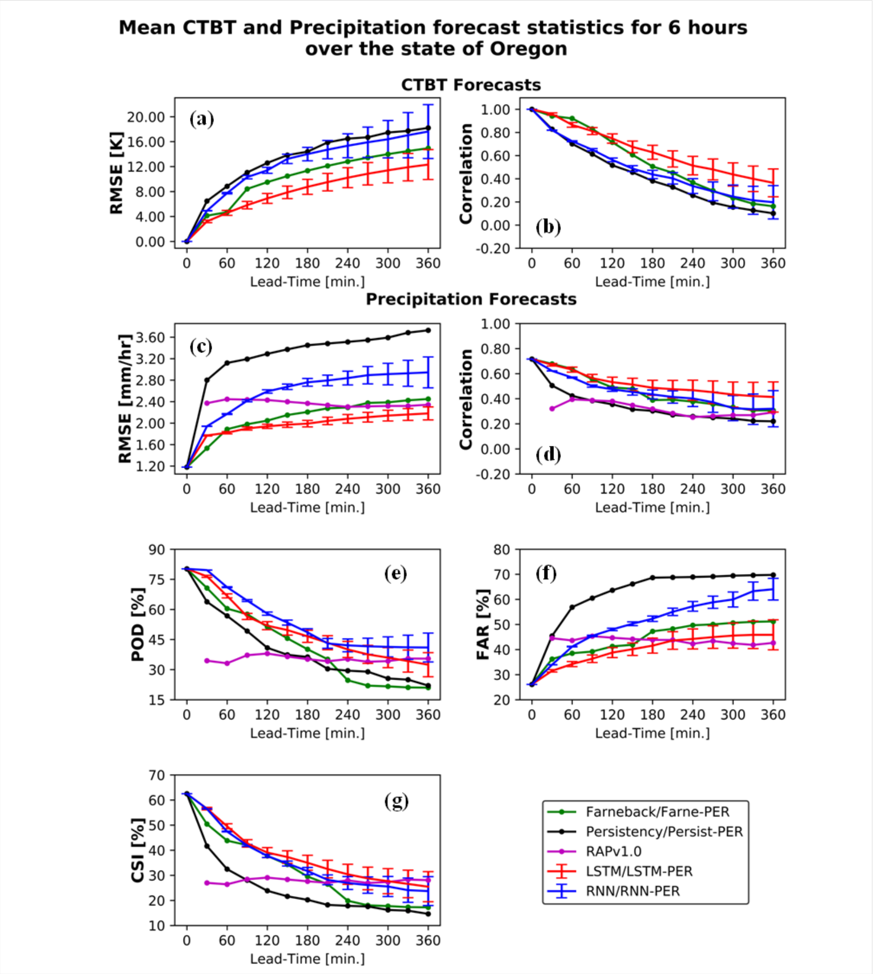


Figure 2.5: The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (cg) 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 2.5d, the LSTM-PER has slightly higher CC comparing to the Farne-PER and RNN-PER models. In Figure 2.5e, the RNN-PER model has the highest POD values with

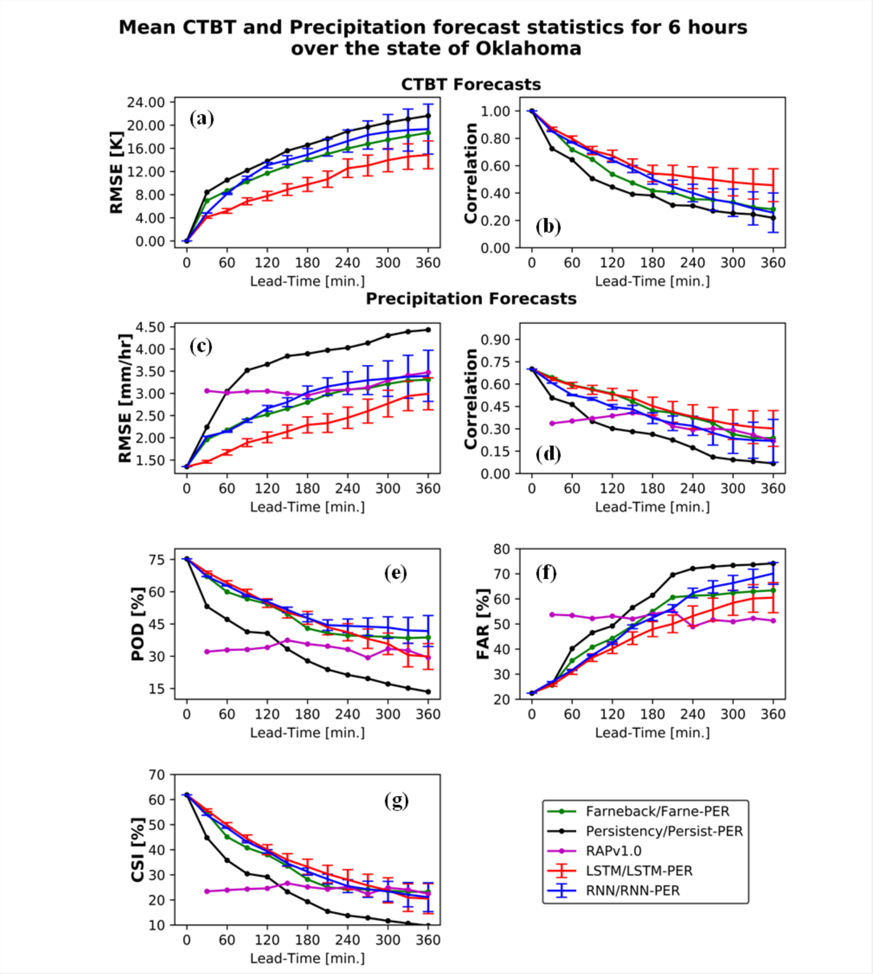


Figure 2.6: The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (cg) 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 percents throughout the forecast time. In Figure 2.5f, the LSTM-PER model has the lowest FAR values up to fourth hour of forecast and from there the RAPv1.0 model provides the lowest FAR values. The Farne-PER, RNN-PER, and Persist-PER models have the third, second, and first highest FAR values, respectively. Figure 2.5g 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 percents with the increase in lead time.

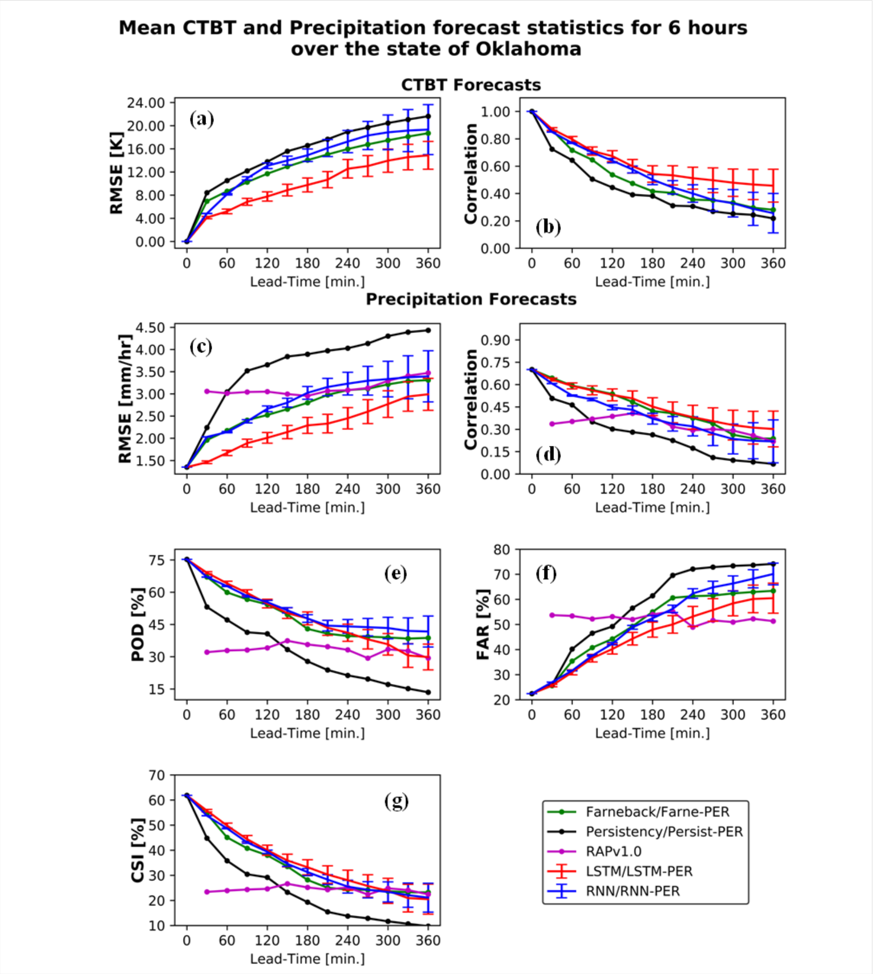


Figure 2.7: . The average of (a) root-mean-square error (RMSE) and (b) correlation coefficient (CC) between the observation and forecasted Cloud-Top Brightness Temperature (CTBT) from the Persistency, Farneback, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) over lead times from 30 to 360 min for whole testing period. (cg) 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.

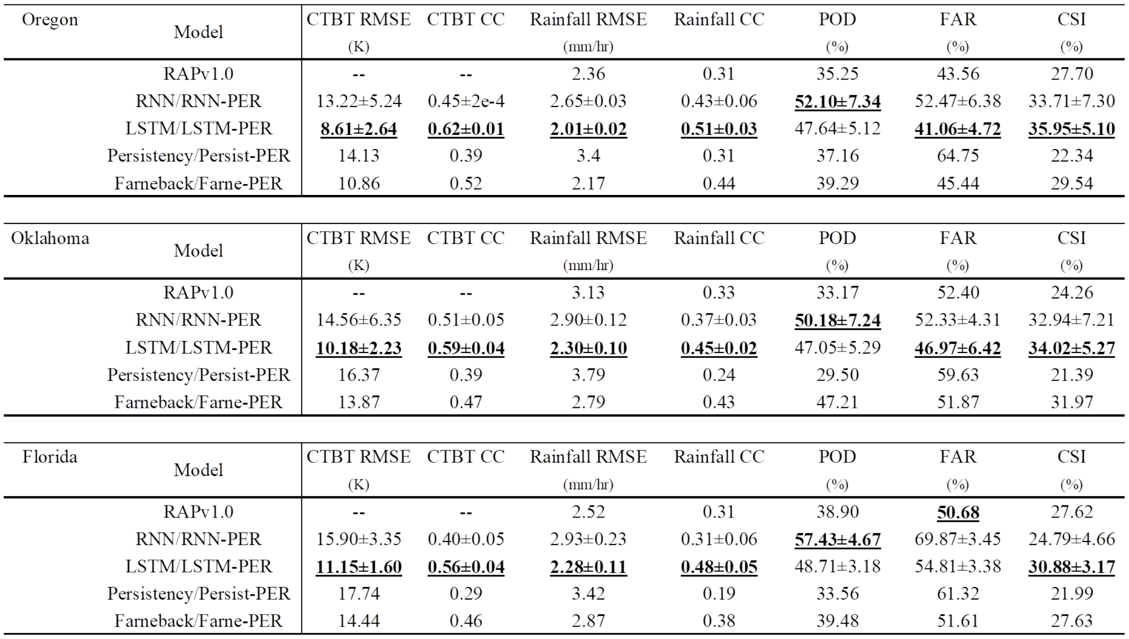
Comparing the 30 independent runs for the RNN-PER and LSTM-PER, the forecasts become more uncertain with the increase in lead time. However, the certainty of the RNN-PER model deteriorates faster than the LSTM-PER model in all comparison metrics (Figures 2.5c2.5g). Figures 2.6a and 2.6b presents the 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 2.5c-2.5g, Figures 2.6c-2.6g show the precipitation forecasting skills over the state of Oklahoma. Figures 2.6c 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 2.6c and 2.6d). In Figures 2.6e, 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 2.6e). In Figure 2.6f, 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 2.6e). Figure 2.6g 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 performance to for the rest of forecast lead time (Figure 2.6g). 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 2.6c-2.6g.

The statistics for the state of Florida is presented in Figure 2.7. Based on Figures 2.7a and 2.7b, 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 2.7a). The second, third, and fourth highest CC also belong to the Farneback, RNN, and Persistency methods, respectively (Figure 2.7b). In Figure 2.7c, 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 2.7c). Figure 2.7d 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 2.7d). In Figures 2.7e and 2.7f, 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 2.7f). In comparison with the LSTM-PER, the Farne-PER shows lower POD values in Figure 2.7e, and lower FAR values in Figure 2.7f 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 2.7e). In Figure 2.7f, 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 2.7g). 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 percents for all the lead times (Figure 2.7g). 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.

Table 1 summarizes the performances of the investigated models by averaging the statistics of 12 forecast frames (6 hr) presented in Figures 2.5-2.7 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.



Oregon	Model	CTBT RMSE (K)	CTBT CC	Rainfall RMSE (mm/hr)	Rainfall CC	POD (%)	FAR (%)	CSI (%)
	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	Model	CTBT RMSE (K)	CTBT CC	Rainfall RMSE (mm/hr)	Rainfall CC	POD (%)	FAR (%)	CSI (%)
	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	Model	CTBT RMSE (K)	CTBT CC	Rainfall RMSE (mm/hr)	Rainfall CC	POD (%)	FAR (%)	CSI (%)
	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

Table 2.1: Average Statistical Performances of RAPv1.0, RNN/RNN-PER, LSTM/LSTM-PER, Persistency/Persist-PER, and Farneback/Farne-PER on Testing Data Set.

2.4.2 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 2.82.10, 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-km2 area (Figures 2.8-2.10).

Figure 2.8 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 2.8 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 2.8, 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 2.8). 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 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 2.8).

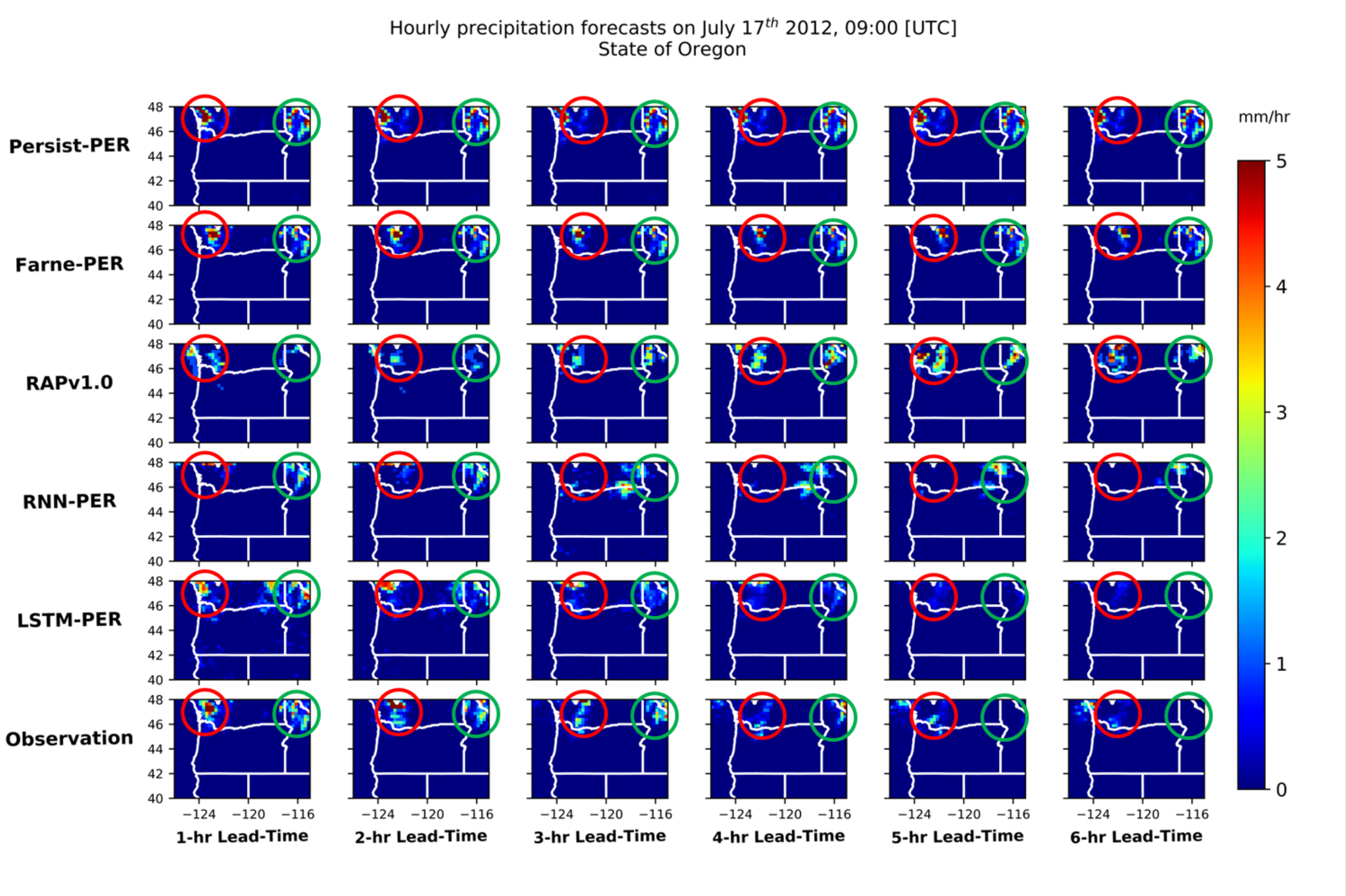


Figure 2.8: Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Oregon made on 17 July 2012 09:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

Figure 2.9 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 2.9) 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 2.9). 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 2.9, 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 2.9).

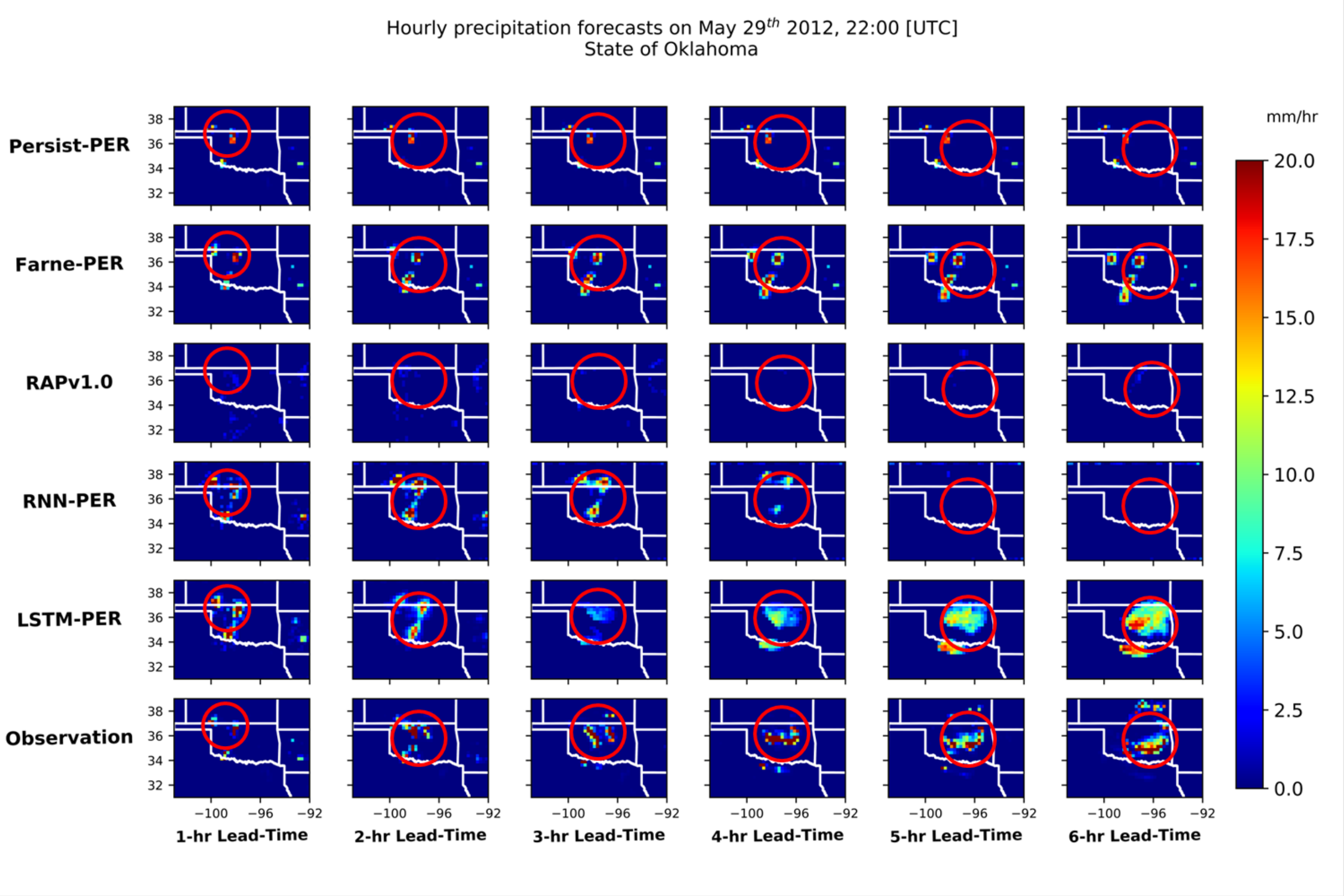


Figure 2.9: Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Oklahoma made on 29 May 2012 22:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

Figure 2.10 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 2.10). 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 2.10). 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 2.10, 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.

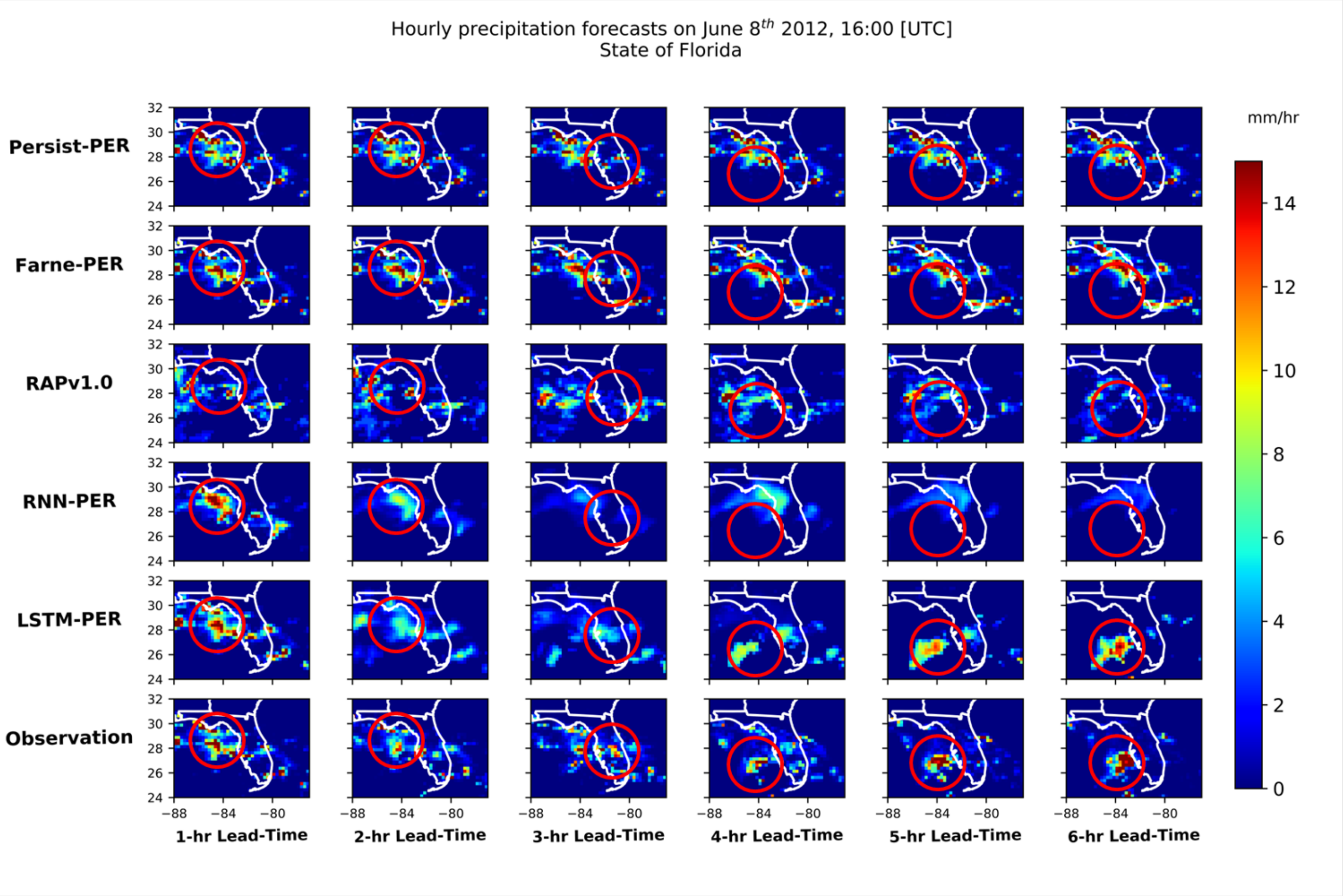


Figure 2.10: Hourly forecast of precipitation from the Persist-PER, Farne-PER, RAPv1.0, RNN-PER, and LSTM-PER models with lead times of 1, 2, 3, 4, 5, and 6 hr, and the observed precipitation on the corresponding time steps. The presented precipitation results are over the state of Florida made on 8 June 2012 16:00. LSTM = Long Short-Term Memory; RNN = Recurrent Neural Network.

2.5 Discussion

2.5.1 Forecasting Performances

According to Figures 2.5-2.7, 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 2.5-2.7). 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 2.5-2.7, 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. [1999] 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 2.5-2.7, 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.

2.5.2 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 2.5-2.7 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 2.5-2.7). 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 2.5-2.7, panels c-g.

2.6 Summary and Conclusion

In this study, I proposed a framework to forecast precipitation in a short term (06 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 (06 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.

Chapter 3

Applications of Conditional Generative Adversarial Networks (CGANs) in Monitoring and Forecasting Complex Earth System Components

3.1 Introduction

Machine learning algorithms can be generally classified into two groups: (1) Discriminative and (2) generative. Discriminative models attempt to link a complex and high-dimensional input to a low-dimensional target (Goodfellow et al. [2014]) On the contrary, generative models map low-dimensional inputs to high-dimensional rich content targets. Recently, generative models have been the main focus of many researchers due to their vital applications. However, generative models were barely investigated due to the hardship of estimating maximum likelihood and related approaches.

3.2 Insufficiency of Maximum Likelihood Loss Functions Such As Mean Square Error and Kullback-Leibler Divergence

Conventionally, many developed generative models utilize traditional loss function such as Mean Square Error (MSE) and Kullback-Leibler divergence (KL divergence), to optimize the model parameters. Such conventional loss functions, mainly benefiting from maximum likelihood, have proven to be effective, especially in discriminative approaches. However, due to the nature of problem in generative approaches, maximizing likelihood become less satisfactory due to its disagreement with human perception (Theis et al. [2015]).

Due to this artifact, generative models can be classified into two categories: (1) Explicit and (2) Implicit models. Explicit models follow the traditional way of parameter optimization by maximizing the likelihood, and assume a well-defined likelihood function exists (Kingma and Welling [2013]). On the contrary, implicit models relax the assumption of having a well-defined likelihood function and is capable of generating a likelihood function based on matching probability distributions (Goodfellow et al. [2014]).

Explicit models suffer from the difficulty of optimizing model parameters based on maximizing the well-defined likelihood function. This is due to the assumptions made by selecting a certain loss function, e.g. MSE and KL divergence. Specifically, looking into the maximizing the likelihood function as

$$\hat{\theta} = \underset{\theta \subset \Theta}{\operatorname{argmax}}\ \mathrm{E}_{x \in p}\left[f(x|\theta)\right] \tag{3.1}$$

$$\hat{\theta} = \underset{\theta \subset \Theta}{\operatorname{argmax}}\ \mathrm{E}_{x \in p}\left[\log f(x|\theta)\right] \tag{3.2}$$

In equation 3.1, the $\hat{\theta}$ and θ are the model parameters and true states of probability distribution which is a subset of Θ. $f(\cdot)$ denotes the density function of the data given the true state variables. The x is the data sample obtained from true distribution p. Equation 3.2 is the maximum logarithmic likelihood and is the most trivial form of likelihood estimation representation due to its easier solution.

In complex distributions, such as the temporal and spatial distribution of natural observations, learning the density functions can be quite tedious from the model perspective. For instance, if a sample from true distribution $(p(x))$ is associated with low probability in the model distribution, then the model is punished by have the $\log f(x|\theta)$ converging to negative infinity $(\log f(x|\theta) \to -\infty)$. Other way around, if a sample data is drawn from model distribution $f(x|\theta)$ with high probability in model distribution $(f(x|\theta))$ and low probability in the true distribution $(p(x))$, the maximum log likelihood penalty for this sample is not significant. Having the two above-mentioned cases allows model to spread out its distribution as much as possible to cover the entire true distribution. This behaviour of maximum likelihood function results in allocating a portion of probability density to the area of sample space where the true probability is zero. In other terms, the the model now expand to the sample space area that its drawn results is not acceptable in practice.

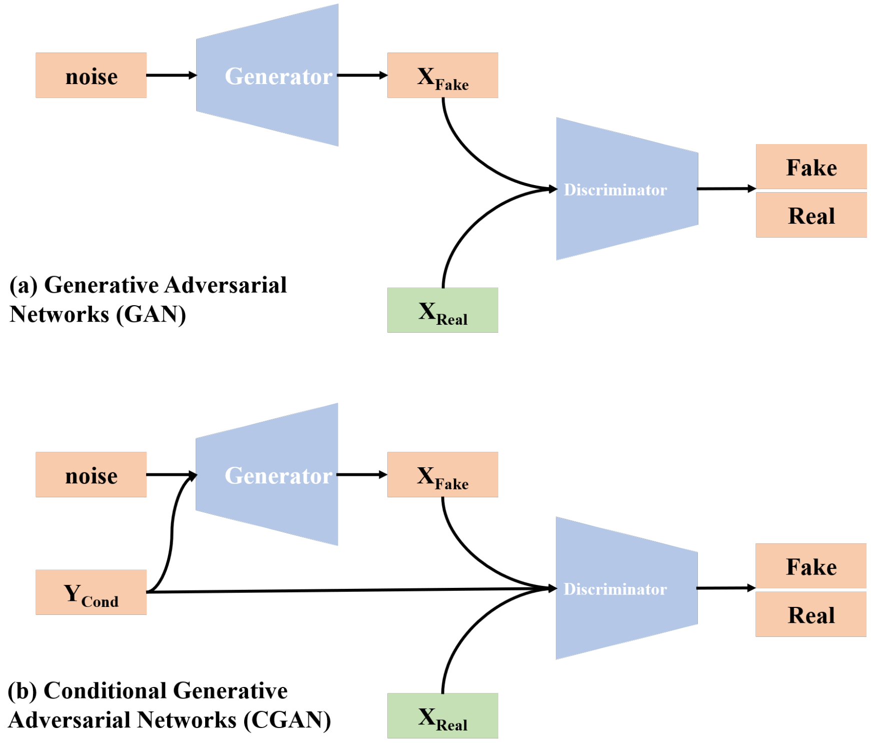
Recently, GAN models introduced a flexible scheme compared to maximum likelihood functions in which it is capable of learning a divergence as the objective function. Due to GANs' flexible evaluation metric, the generative model can be trained to minimize the Jenson-Shannon entropy which is symmetric divergence function and does not follow the flaws of KL divergence [Nowozin et al., 2016]. In other words, GANs are capable of expanding the probability mass as much as possible but in the same time, the model penalize the unrealistic samples, thus leading the model to learn the best distribution possible to describe the true distribution.

3.3 Generative Adversarial Networks

Generative Adversarial Networks (GANs), introduced by Goodfellow et al. [2014], is a new branch of generative models consisting from two adversary networks: (1) generator and (2) discriminator. The GAN architecture is an adaptation of minimax problem in game theory in which two players (i.e. minimizer and maximizer) compete. Maximizer attempts to win by maximizing the game score, and the minimizer tries to minimize the score by countering the move of maximizer. Minimax is a well-defined problem in game theory which has been proven to converge to a saddle point in which both players are satisfied with their strategies. In specific, the game ends where the score is maximum with respect to maximizer's strategy and minimum with respect to the minimizer's strategy.

3.4 Conditional GANs

In extension of GANs, Mirza and Osindero [2014] introduced a conditional GAN in which both generator and discriminator are conditioned on an input data rather than generating results from random noise. This extension of GAN enables the model to yield input-driven results.



(a) Generative Adversarial Networks (GAN)

(b) Conditional Generative Adversarial Networks (CGAN)

Figure 3.1: The general structure of the Generative Adversarial Networks and Conditional Generative Adversarial Networks. In part (a), the Generator takes noise to generate fake (X_{Fake}) data and discriminator decides the validity of the generated data by evaluating both real data (X_{Real}) and fake data (X_{Fake}). In part (b), the generator is conditioned on Y_{Cond} to generate fake data (X_{Fake}). Y_{Cond} is also fed into the discriminator in order to evaluate the fake (X_{Fake}) and real (X_{Real}) data based on their assigned condition.

3.5 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are, a class of feed-forward neural networks, designed to efficiently handle the spatial correlations within certain data types such as images. CNNs can learn shift and distortion invariant features using local receptive field and weight sharing. Traditional fully-connected neural networks are proven to be usefull in dealing with low-dimensional data such as point measurements, however, they are impractical while dealing with high-dimensional inputs data such as images. This inefficiency are due to the connection of neurons to all the neurons in the next layer, where not all the connections are required. In CNNs, instead of connecting all the neurons, the neurons are connected to a small group of neurons in the next layer, called receptive field. This modification in neuron connections results in focusing on spatially local features rather than a larger area with less spatial coherency.

The weight sharing property of CNNs has the advantages of (1) reducing the number of model parameters compared to traditional fully-connected neural networks, (2) decreasing the chance of over-fitting and (3) learning homogeneous features.

In the CNN architecture, the layers convolve a set of filters over each input sample (3.3 and 3.4) and tune the filters by calculating the gradient of the loss function with respect to each filter weight.

$$\begin{aligned}(f * g)(t) &= \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau \qquad (3.3)\\ &= \int_{-\infty}^{\infty} f(t-\tau)g(t)d\tau \qquad (3.4)\end{aligned}$$

The use of only CNN architecture, the location of the features is less important and the model focusing on only detecting the feature somewhere in the spatial domain. Thus, detecting all the features in an image with perseverance of position relative to other features will result in misclassification, i.e. any arbitrary order of face components will result in face detection. In order to solve this shortcoming, a subsampling scheme is introduced to reduce the sensitivity of the model outputs to shifts and distortions [LeCun et al., 1995]. Moreover, utilizing subsampling in the CNN architecture results in (1) learning translation invariant features and (2) reduction of the computational complexity.

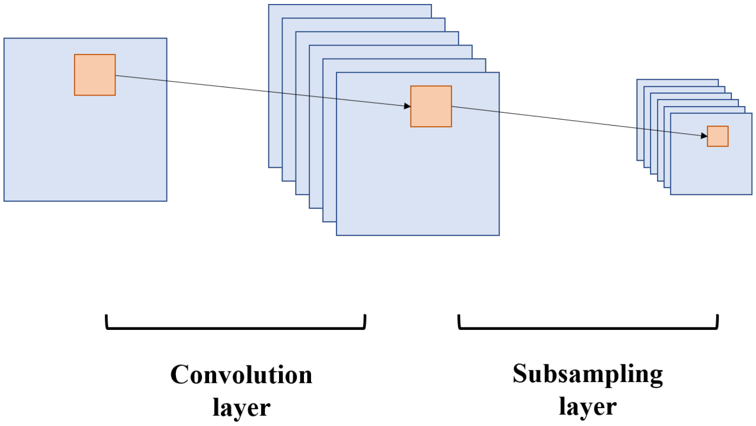


Figure 3.2: Schematic representation of Convolution and Subsampling layers: In the convolution part, a kernel (orange box) is convolving over the image and the response of this convolution is stored in the feature layer. The Subsampling layer regrids the feature layers into courser spatial and/or temporal resolutions. Max-Pooling and Average-Pooling are the two most popular subsampling techniques in which the maximum and the average of a sampling window is selected to represent that window in courser resolution, respectively.

3.6 U-net Structure

In the recent years, CNN models have outperform the state-of-the-art models in various tasks of visual recognition [Girshick et al., 2014, Krizhevsky et al., 2012, Karpathy et al., 2014]. The significance of CNN performance is mainly on the image classification tasks in which the model learns to classify images into a set of classes [Krizhevsky et al., 2012, Karpathy et al., 2014]. However, generative models require localization of outputs meaning that the value for each pixel should be determined by model [Ronneberger et al., 2015]. In order to address this issue, many attempts [Ciresan et al., 2012, Roth et al., 2015] used an sliding-window scheme in which the model uses a subset of image to predict the center value of that window, and by sliding this window all over the image, an output with certain values for each pixel is provided. The sliding-window scheme provides the localized results for each window and satisfy the requirements of generative models. However, despite the promising performance of sliding-window scheme, the method has two main issues: (1) the model is slow due to running over all the windows within each image and redundantly runs over overlapping patches and (2) the trade-off between using a smaller window and less subsampling which limits the context and using a larger window and more subsampling which reduces the localization accuracy. The U-net structure, introduced by Ronneberger et al. [2015], is a fully convolutional neural network with bypass connections from the features before each subsampling layer to the same-level features after upsamling (Figure 3.3). The bypass connection concatenates two same-level feature blocks in order to prevent ”loss of resolution” problem caused by subsampling units.

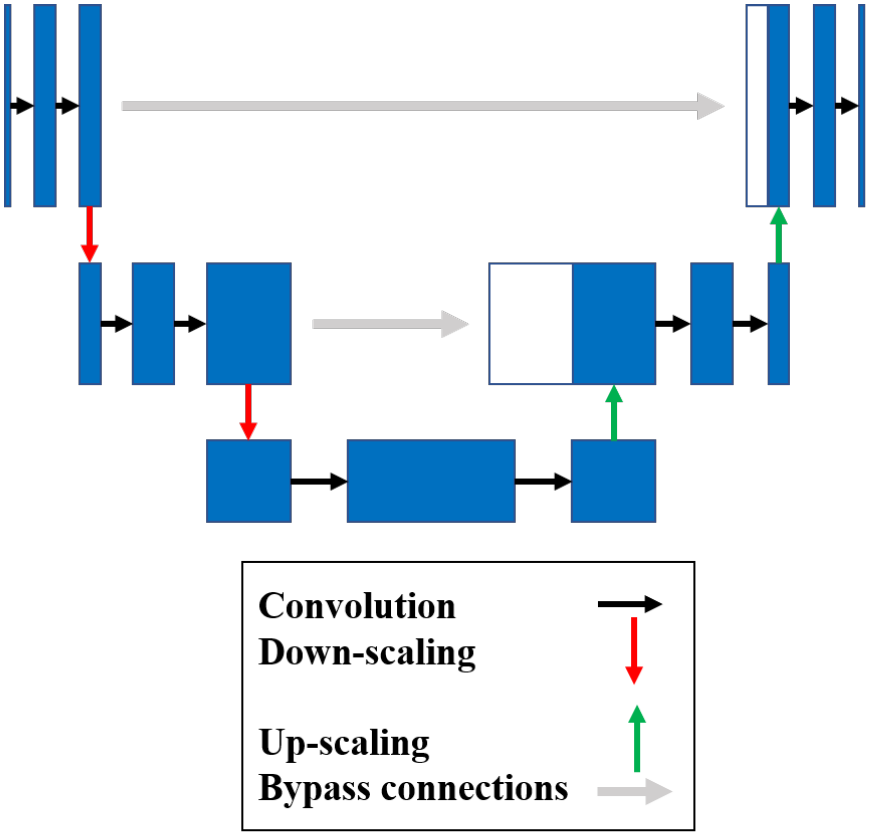


Figure 3.3: Schematic representation of U-net structure: The U-net strucuture is a fully convolutional network consisting of convolution layers and down-scaling (from fine to course resolution) and up-scaling (from course to fine resolution) and bypass connections (grey arrows). The bypass connections allow the model to copy and concatenate the encoding features to decoding features at each level and maintain the spatial information and down-scaling advantages at the same time.

3.7 SUM-net Structure

U-net algorithm is effective to solve the problem of ”loss of resolution”, however, due to crop and copy operation, the decoding part of the model requires roughly two times more computations compared to the encoding part. In order to make the U-net structure more efficient, instead of concatenation of same level encoding features with decoding features, a summation operation will be conducted (Figure **??**). This will forces the decoding part to learn more expressive features using the information for same-level encoder layer and previous layer.

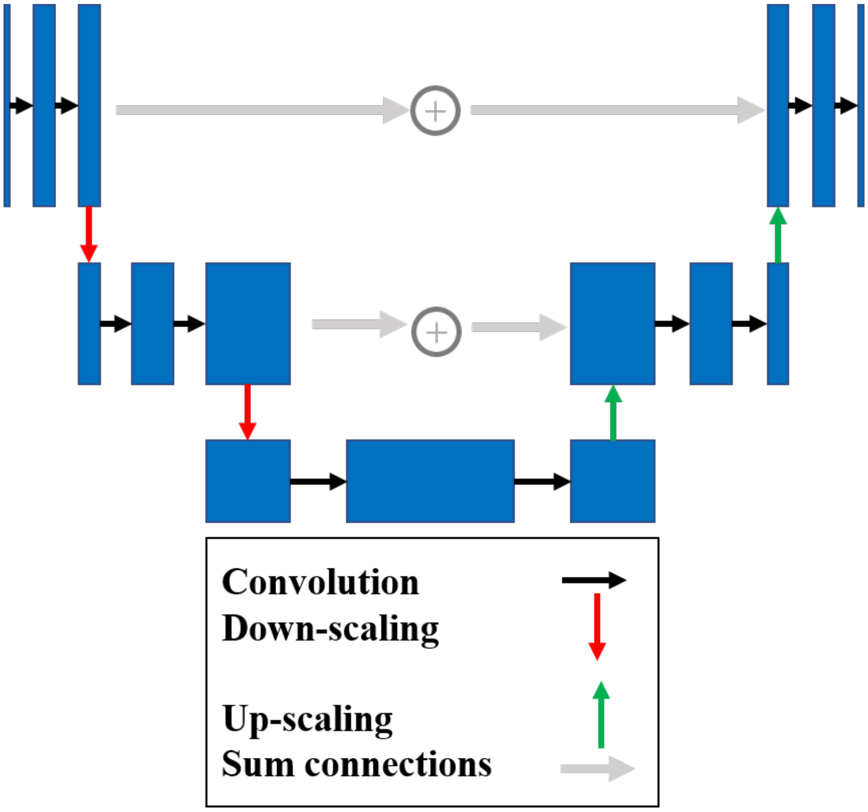


Figure 3.4: Schematic representation of SUM-net structure: The SUM-net structure is a fully convolutional network consisting of convolution layers and down-scaling (from fine to course resolution) and up-scaling (from course to fine resolution) and bypass connections (grey arrows). The sum connections allow the model to efficiently combine the encoding features to decoding features at each level and maintain the spatial information and down-scaling advantages at the same time.

Chapter 4

Short-Term Precipitation Forecast using CGANs constructed from Convolutional, Recurrent and ConvRecurrent layers

4.1 Introduction

In this chapter, the aim is to introduce a more accurate precipitation forecasting system compared to the one introduced in chapter 2 leveraging the recent developments in the field of machine learning and deep learning. In the previous study, the application of the Recurrent Neural Networks (RNNs) in forecasting infrared image sequences and their linked precipitation forecast, using PERSIANN algorithm, was investigated. The results presented in chapter 2 demonstrated higher forecasting skills of advanced RNNs termed as Long Short-Term Memory (LSTM) compared to traditional methods such as Optical Flow and Numerical Weather Prediction models. Despite the effectiveness of LSTM algorithm, there are several shortcomings that LSTM model cannot sufficiently address; (1) Efficient spatiotemporal learning, (2) ”loss of resolution” problem, and (3) difficulties of using Maximum Likelihood objective function.

4.1.1 Efficient Spatial Learning

Traditionally, fully-connected layers were introduced as the first type of neural networks. Fully-connected layers link all the inputs to all of the hidden nodes in the hidden layer. Similar for the more deeper representations, fully-connected layers connect all of the hidden nodes to the the hidden nodes in the next hidden layer. This architecture was found promising, especially for sparse input data. However, with the attempt to solve more complicated problems with neural networks, including image classification, image segmentation, image sequence forecast, etc., the conventional fully-connected neural networks, proven to be less useful. The incapability of the fully-connected networks, especially for learning spatial patterns emerge from the over-parameterization of fully-connected models in which most neurons do not fire to discriminate an image. In other words, fully-connected layers are inefficient to learn local features. LeCun et al. [1995] introduced Convolutional Neural Networks (CNNs) using convolutional structure in which the parameters are shared in space and each filter is forces to learn homogeneous local features throughout the dataset.

CNN algorithms have shown promising performances to efficiently solve various problems, especially in image classification tasks [Krizhevsky et al., 2012, Karpathy et al., 2014]. Most popular approach to solve a image classification problem is to use convolutional layers with subsampling to link high-dimensional image data to the low-dimensional target class [Krizhevsky et al., 2012, Girshick et al., 2014, Karpathy et al., 2014]. Researchers have extended this approach to image to image translation and image segmentation by changing the architecture into a compressive encoder-decoder CNN. In this version of CNN, the model links the high-dimensional image data to its corresponding results by reducing the spatial dimensionality, using subsampling techniques, to obtain a lower-dimensional feature representation in the encoder part. Later, the decoder links the low-dimensional features into high-dimensional image space via upsampling methods 4.1. This approach introduces blurriness to the model output due to the well-known problem of "loss of resolution". To address this problem, SUM-net structure described in 3 is introduced to overcome this problem.

Despite the promising performances of CNN algorithms in learning spatial patterns, they have limited capabilities to learn long range of temporal patterns depending on the size of their receptive fields. Thus, an efficient and effective combination of CNNs and RNNs are required for high-accuracy forecasting. Xingjian et al. [2015] has introduced a new neural network structure termed as ConvLSTM in which all the fully connected connections are replaced with convolutional operations. ConvLSTM layer have shown promising performances in terms of forecasting precipitation from radar echo data. In addition to the above-mentioned problems, traditional objective functions conventionally used for training neural networks, such as Mean Square Error (MSE) and Kullback-Leibler divergence, which categorize under Maximum Likelihood functions suffer from many intractible probabilistic approximations that makes the generative models accurate [Goodfellow et al., 2014].

In this study, a framework will be introduced to efficiently address the mentioned issues, and accurately forecast infrared imageries into the future.

4.2 Methodology

4.2.1 ConvLSTM: A Convolutional Layer With Long Short-Term Memory Feature

The ConvLSTM layer has been developed by Xingjian et al. [2015] with the idea of replacing the fully-connected connections with convolutional operations in LSTM model to benefit from local information in the data. This modification was conducted by replacing all dot products with convolution operation. The ConvLSTM model without peephole connections can be formulated as

$$i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + b_i) \tag{4.1}$$

$$f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + b_f) \tag{4.2}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_{hc} * x_{t-1} + b_c) \tag{4.3}$$

$$o_t = \sigma(W_{xo} * x_t + W_{ho} * x_{t-1} + b_o) \tag{4.4}$$

$$h_t = o_t \odot tanh(c_t) \tag{4.5}$$

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 equations4.1-4.4 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 $\odot$ and $*$ symbols in equations 4.1-4.5 represent the inner product of matrices and convolution operation. Note that σ and $tanh$ in equations 4.1-4.5 represent the Sigmoidal and Hyperbolic Tangent activation functions, respectively.

4.2.2 Proposed CGAN Framework For Short-Term Precipitation Forecast

The proposed framework, as shown in Figure 4.1, is a conditional GAN model consisting of a generator and a discriminator. The generator is a fully convolutional network, considering ConvLSTM as fully convolutional, with SUM-net architecture. The first and last two layers are convolutional-Leaky ReLU combination. In addition, third and fourth layers consist of convolutional-Leaky ReLU-MaxPooling and the most abstract features are obtained from ConvLSTM layers. The decoder part uses convolutional-Leaky ReLU-UpSampling for the two layers after ConvLSTMs and then only convolutional-Leaky ReLU. The input to the generator contains $t - n$ to t consecutive images and predict $t - n + 1$ to $t + 1$ samples. The discriminator is semi-conditioned using the $t - n + 1$ to $t + 1$ consecutive samples from the generator predictions and ground-truth samples. Using three convolutional-Leaky ReLU blocks shared in time for all the samples followed by a LSTM and a fully-connected layer, the discriminator discriminates whether if the input sample is from real or fake distribution. This architecture allows discriminator to consider the temporal and spatial coherency of the generated samples in order to fairly evaluate them and due to high overlaps between conditioning data (input of generator) and output of generator, inserting conditioning data is not required, thus making the model a semi-conditional GAN.

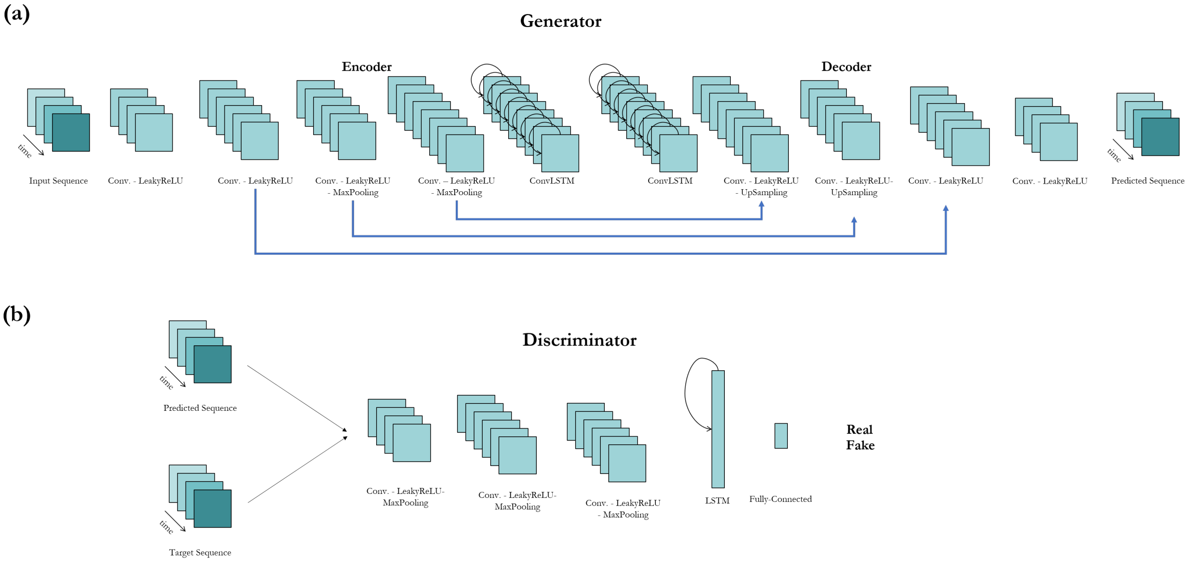


Figure 4.1: The details of the proposed Conditional GAN model; (a) The generator is a Fully Convolutional Networks (FCN) consisting of Convolutional and ConvLSTM layers. Input data is image sequences from $t - n$ to t and the output of the model is predicted image sequences from $t - n + 1$ to $t + 1$. The convolutional layers are shared in time in order to learn the spatial patterns. The blue arrays indicate sum operation to fulfill the SUM-net architecture and efficiently transfer temporal and spatial information from encoder to decoder. (b) The discriminator takes the image sequences from generator output and ground-truth target sequence and using the last state of the LSTM, decides whether the sample was from a real or a fake distribution.

4.3 Data and Case Study

4.3.1 Infrared Imageries

The input data used in this study is the long-wave IR channel with the wavelength of $10.8 \mu m$ from the Geostationary Operational Environmental Satellite(GOES) series. The IR channel provides information on the cloud-top temperature which is a useful feature for identifying the precipitable clouds. The utilized IR data is provided by the Climatic Prediction Center (`http://www.cpc.ncep.noaa.gov/products/global_precip/html/wpage.merged_IR.html`) and has a near-global coverage(60°N-60°S) with a $0.04^{\circ} \times 0.04^{\circ}$ spatial resolution. In this study, the IR data was regridded to $0.25^{\circ} \times 0.25^{\circ}$ resolution using the nearest neighbour algorithm to be consistent with the spatial resolution of PERSIANN algorithm.

4.3.2 Data Sampling

In this study, the focus is on learning spatial and temporal evolution patterns in the infrared (IR) data. Learning IR imageries are particularly selected due to its smooth transition in space and allows the model to mimic the heat transfer functions governing the changes of IR images in time. In order to obtain uniform samples from different evolution patterns of IR with different mechanisms, we randomly sampled 32×32 windows across the CONUS which had 20 percents or more pixels with less than 220 K compared to the number of pixels in the window (Figure 5.1). The criteria is set to ensure the model will observe samples with high moisture clouds. In the sampling process, with the fixed spatial location of the samples, we cropped a 20-sequence of that window in time to feed into our model as partially input and partially target values.

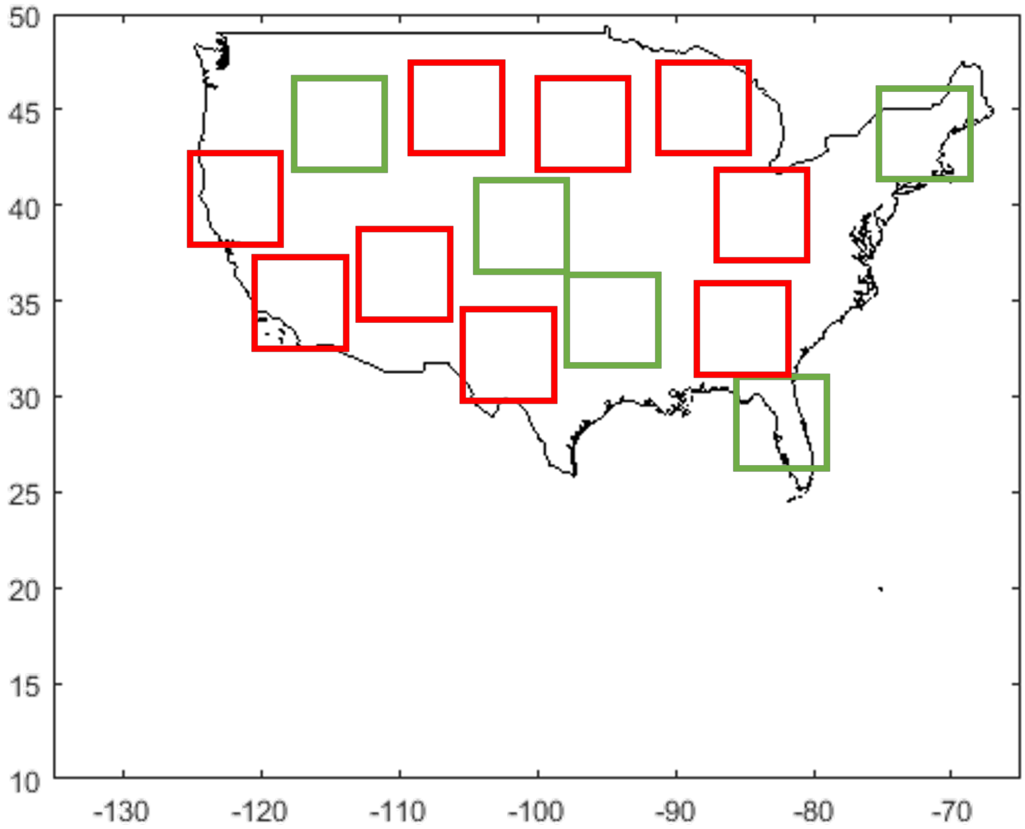


Figure 4.2: The random selection of sample patches for training process over the CONUS. The red rectangulars are the randomly selected samples which did not contain 25 percents or more rainy pixels. The green rectangulars are the samples which satisfied the rainfall criteria at each time step and are selected for training.

4.4 Results and Discussion

Visual Comparisons

Figure 4.3 demonstrates the evolution steps of infrared imageries from 0 to 5 hours ahead over the CONUS. Rows from top to bottom represent Target observations, ForeGAN (our proposed model) forecasts and Non-adversarial version of the ForeGAN (i.e. CNN with bypass connections) forecasts. Forecasts with increase in lead-time are shown from left to right columns ($t + 1$ to $t + 10$). The forecasts obtained from ForeGAN model are more accurately following the patterns of infrared evolution, especially the cold cloud regime over the northern part of the United States. On the contrary, CNN with bypass cannot follow the patterns as well as ForeGAN model and generates blurry results. The blurriness magnifies by the increase in lead-time and the forecasts obtained deteriorate from observed infrared imageries. It is noteworthy that spotted noise in ForeGAN, especially in further lead-time, are due to the artifacts of Fully Convolutional Networks and can be eliminated by further modification of hyper-parameters of the model.

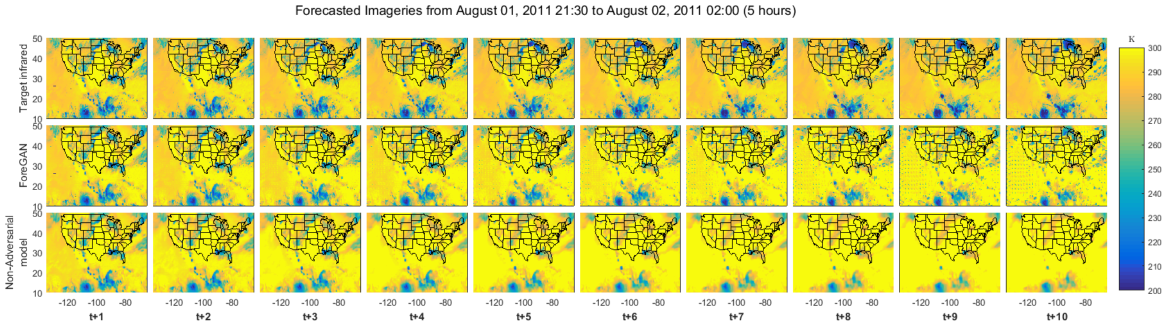


Figure 4.3: The result of ForeGAN and CNN with bypass connection over CONUS. The first, second and third rows represent Target infrared, ForeGAN forecasts and non-adversarial version of introduced model (CNN with bypass) forecasted imageries for 5 hours from August 01, 2011 21:00 [UTC] to August 02, 2011 02:00 [UTC]. Columns from left to right demonstrate half-hourly forecast time-steps.

Furthermore, Figure 4.4 is a subsection selected from Figure 4.3 in order to better demonstrate the evolution of cloud patterns. The ForeGAN model is able to detect the evolution patterns compared to infrared observations, however, the Non-adversarial model cannot follow the changes of infrared shape and location.

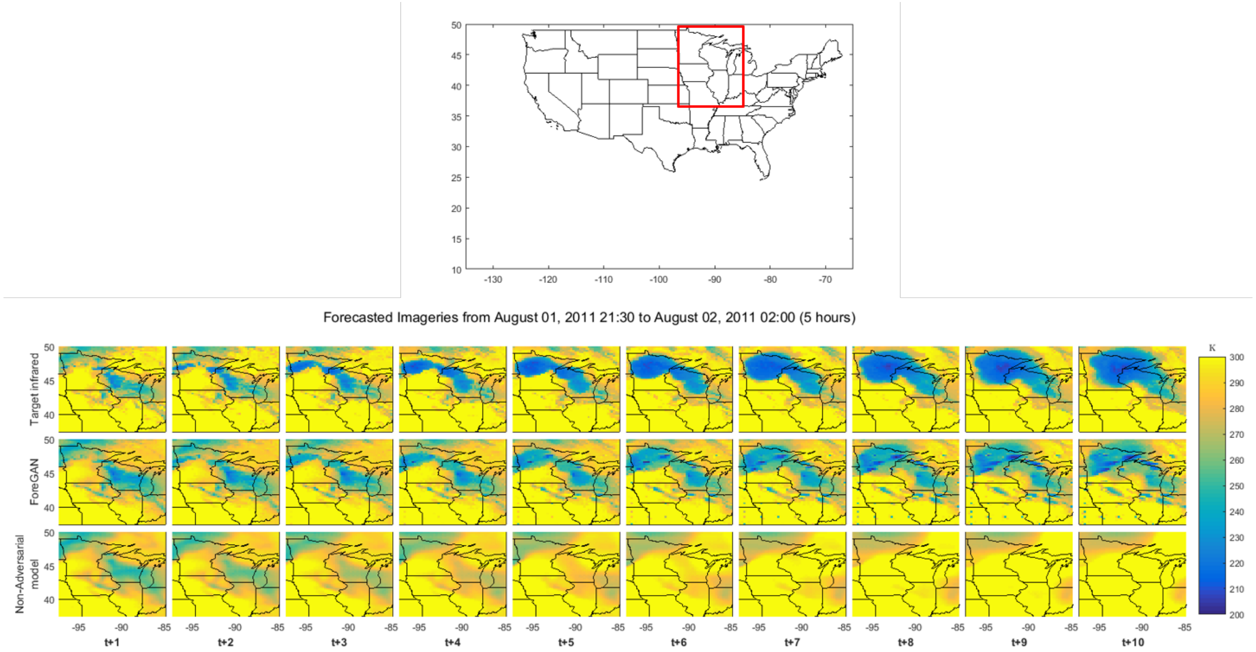


Figure 4.4: The result of ForeGAN and CNN with bypass connection for the sub-area within red box over CONUS. The first, second and third rows represent Target infrared, ForeGAN forecasts and non-adversarial version of introduced model (CNN with bypass) forecasted imageries for 5 hours from August 01, 2011 21:00 [UTC] to August 02, 2011 02:00 [UTC]. Columns from left to right demonstrate half-hourly forecast time-steps.

Furthermore, statistics such as RMSE, Correlation coefficient, SSIM and PSNR are calculate over the extended CONUS coverage to evaluate the predictability skills of ForeGAN for the same period of lead-time as presented in Figure 4.3. The statistics presented in Figure 4.5 evaluates ForeGAN and CNN with bypass connections with respect to target observation in terms of RMSE, correlation coefficient, SSIM and PSNR indices. As demonstrated in Figure 4.5, ForeGAN in all terms and for all lead-time steps performs better than its non-adversarial version significantly. It is noteworthy that in all metrics, with increase in lead-time the deterioration rate of ForeGAN is slower than CNN with bypass connections (Figure 4.5).

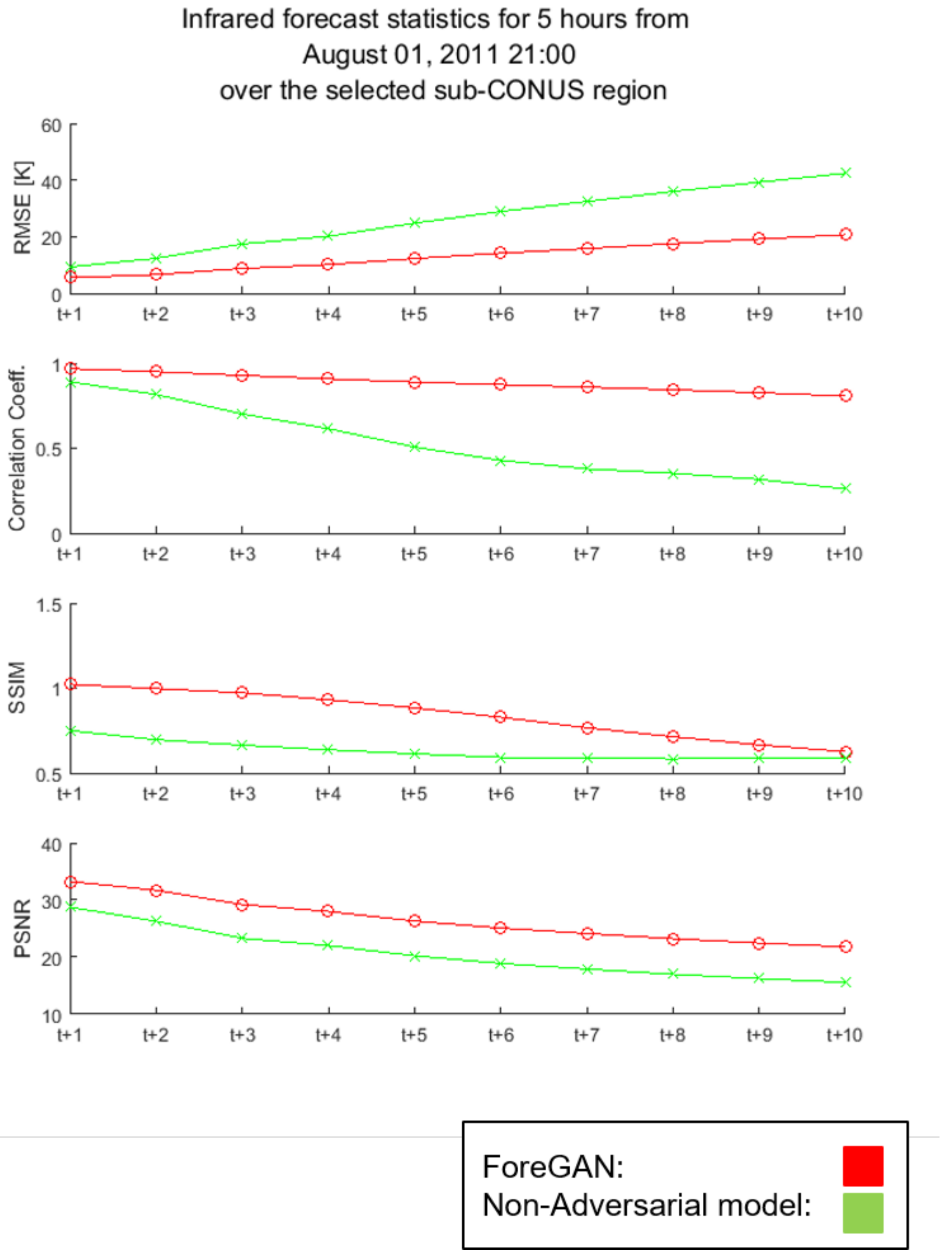


Figure 4.5: The predictability statistics for ForeGAN (red lines) and CNN with bypass connections (green lines) over the area marked with red rectangular in Figure 4.4. From top to bottom subplots, RMSE, correlation coefficient, SSIM and PSNR were calculated from August 01, 2011 21:00 [UTC] to August 02, 2011 02:00 [UTC], respectively. $t + 1$ to $t + 10$ on the x axis indicate the increase in lead-time.

4.5 Summary and Conclusion

This study presents a framework to forecast infrared imageries in short range of 0 to 6 hours. The proposed framework introduces a semi-conditional GAN architecture with flexible learning divergence to accurately forecast infrared images. The proposed approach is investigated over the whole CONUS and compared to its non-adversarial CNN and the method introduced in chapter 2. The results were evaluated based on visual infrared evolution and statistics such as RMSE, Correlation coefficient, SSIM and PSNR metrics. The main conclusions of this chapter are:

- Based on my experiments, the proposed GAN model accurately forecasts short-term infrared imageries, especially for low-temperatures which can yield precipitation.

- In particular, GAN approaches are an effective and useful techniques for accurate forecasting complex infrared structures and its evolution in time. The advantage of GAN-based forecasts is their sharpness quality even after few timestep forecast.

- GAN models are particularly more suitable for mimicking complex earth system processes such as cloud-top temperature forecasting due to their flexible objective function, and their capability in learning the true distribution via using generator-discriminator architecture.

- Defined structure of the forecasting model demonstrates higher capabilities in preserving details of cloud shapes and image structure compared to LSTM model and non-adversarial model in terms of visual comparison and statistical indices such as PSNR, SSIM and correlation coefficient.

Chapter 5

Precipitation Estimation from Remotely Sensed Information

5.1 Introduction

Precipitation is the driving force of the hydrological cycle and precipitation information is vitally important for reservoir operations, flood monitoring and water management purposes. The precipitation pattern governs much of the weather, climate, and ecological systems around the world. Timely and accurate precipitation information is of paramount importance for water resources management as well as for natural hazard prediction and management. The ground-based precipitation measurement systems such as rain gauges and weather radars typically suffer from the spatial and temporal gaps that tends to limit their application for remote and poorly instrumented regions. Satellite-based precipitation information has the advantage of near real-time estimation and global coverage, at high spatial and temporal resolutions which is particularly valuable for the parts of the world that do not have sufficient rainfall measurement networks. Additionally, satellite-based precipitation products provide rainfall measurement over the oceans and deserts which are not measured directly.

Satellite-based precipitation products typically rely on information from two types of satellites. Geosynchronous-Earth-Orbiting (GEO) and Low-Earth-Orbiting (LEO) satellites provide passive microwave (PMW) and Infrared (IR) data, respectively. Each of these data types provide advantages for their respective precipitation products. PMW data has the advantage of being directly measured from the hydrometeor content, whereas the IR data can only give information on the cloud top characteristics. However, given the longer record of GEO data together with its higher temporal resolution (compared to the PMW data), long archives of consistent precipitation information with high spatial and temporal resolutions can be achieved by utilizing the IR data. These advantages make the IR-based products suitable for climate studies and hydrological modeling where long records of data are required; as well as for extreme precipitation studies where information with high spatial and temporal resolution about the stages of the storm evolution are needed.

As an initial approach to link infrared cloud-top brightness temperature to rainfall rates, an empirical relation was introduced by Arkin [1979] and Arkin and Meisner [1987] called GOES Precipitation Index (GPI). The infrared-surface rainfall relation, using radar and satellite data, was established by differentiating rain/no-rain pixels with a fixed IR temperature (235 K). Then, $2.5° \times 2.5°$ grid points with temperatures less than 235 K are assigned to a fixed mean rain rate of 3 mm/hr. Finally, the instantaneous rain rates are accumulated to provide monthly rainfall data. Following Arkin [1979] work, Adler et al. [1993] introduced an adjusted version of GPI which adjusts the constant rainfall rates estimated by GPI using an adjustment ratio obtained from monthly microwave rainfall over monthly GPI estimated rainfall for each pixel. Later, rain rates for each pixel would be adjusted by its multiplication to the ratio. Furthermore, a more advanced technique developed by Hsu et al. [1997] termed as Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks (PERSIANN) is a widely-used operational precipitation product that uses IR data to estimate the rain-rate. The PERSIANN algorithm utilizes the cloud-top-temperature information in an unsupervised neural network framework called self organizing feature map to classify the pixels with rain or no-rain. Then, it uses an exponential regression function with a mean squared error(MSE) objective function to estimate the rain rates at a 0.25-degree spatial and hourly temporal resolutions. **?** improved the PERSIANN system in terms of method and introduced the PERSIANN-Cloud Classification system (CCS) which is also a near real-time operational product. PERSIANN-CCS algorithm extracts cloud features such as geometry and texture to estimate the precipitation at 0.04-degree spatial and hourly temporal resolutions. Similar to the PERSIANN algorithm, an exponential regression function with MSE objective function was used for the regression part of the PERSIANN-CCS method. **?** proposed a deep neural network algorithm that utilized bispectral satellite data from IR and water vapor (WV) channels to predict precipitation 0.04 degree spatial and hourly temporal resolutions. The proposed method took advantage from a commonly used deep learning algorithm called stacked denoising auto encoders(SDAE) in both detection and estimation stages. For the regression section, the two stage model benefited from a combination of MSE and Kullback-Leibler(KL) divergence loss functions. It was shown that the two stage model outperforms PERSIANN-CCS in both detection and estimation skills considerably.

5.2 Methodology

5.2.1 Proposed CGAN Framework For Precipitation Estimation

The proposed model consists of a generator and discriminator. The generator model is a fully convolutional network with copy and concatenate connections at each level to create a U-net structure. The generator is a Fully Convolutional Network (FCN) consisting of convolutional blocks in which a convolutional layer, a Batch Normalization (BN) layer and a leaky Rectified Linear Unit (leaky ReLU) activation layer are respectively utilized. The BN layer, in the above context, increases the stability and effectiveness of the generator. The BN layer reduces the variability of input data distributions which can be drastically different in each batch, and enables each block to learn independent features by pushing the distribution into a more consistent distribution [Ioffe and Szegedy, 2015] and also smooths the response surface for easier optimization via stochastic gradient descend and it sub-derived methods [Santurkar et al., 2018].

In this Network, the goal of the generator is to produce rainfall fields which are close to the real distribution of rainfall conditioned on the cloud-top brightness temperature images from infrared data. The discriminator is responsible to evaluate the generated data from generator by comparing the generated distribution to the real distribution drown from real data samples. The discriminator consists of two convolutional blocks each containing a convolutional layer, a ReLU activation layer and a dropout layer with 50 percent dropout rate. The last layer is a fully connected layer with Sigmoid activation function which yields the probability of the generated image being from a real distribution or a fake distribution.

Parameter Tuning

Conditional GAN model parameters are tuned by minimizing

$$L_{CGAN}(G,D) = \mathrm{E}_{x,y}[logD(x,y)] + \mathrm{E}_{x,z}[log(1 - D(x,G(x,z)))] \tag{5.1}$$

In the above equation, x, y and z represent the conditional data, target data and random noise, respectively. In Equation 5.1, the generator (G) tries to minimize the objective to gain more realistic results, and at the same time, discriminator (D) tries to maximize the objective to discriminate between the fake and real examples with highest margins.

Previous approaches ([Pathak et al., 2016, Isola et al., 2017]) have proposed combining traditional loss functions with the GAN loss function presented in 5.1 for the generator model. Combining a traditional loss function such as L2 objective function can improve the generated images compared to GAN loss only ([Isola et al., 2017]). In this study we implemented a combined loss of L2 loss function with a weight of 10 ($\lambda = 10$) with the GAN loss for the generator model (G)

$$G^* = \underset{G}{\operatorname{argmin}} \max_{D} L_{CGAN}(G, D) + \lambda \, \mathrm{E}_{x,y,z}[\|y - G(x, z)\|_2] \tag{5.2}$$

The above modification in the generative loss function forces the generator to not only try to misguide the discriminator, but also aims to reduce the L2 loss. This modification in the loss function would not alter the discrimination process.

5.2.2 Evaluation Metrics

In order to evaluate the performance of the studied models in detecting and estimating precipitation, various evaluation metric are used. The categorical and continuous evaluation metrics that are used in this study are explained in the following sections.

Continuous Indices

The models' performances in predicting the rainfall rate are inspected through the use of continuous indices. The continuous evaluation indices that we use in this study are root mean squared error(RMSE) and Structural Similarity (SSIM). RMSE is the commonly used measure of the differences between the model's predictions and observation and it is given by equation(5.3):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \left(\hat{y}_i - y_i\right)^2}{N}}, \tag{5.3}$$

$$i = 1, ..., N \tag{5.4}$$

where $\hat{y}$ and y are the model prediction and observation, respectively. N is number of pixels in the study area.

SSIM is a measure of similarity between two images and is considered as an improved version of the classic evaluation indices such as means squared error (MSE). SSIM can be regarded as a measure of quality of an image based on another image that is considered having the perfect quality. SSIM evaluates the visual impact of contrast and structure characteristics of an image and is given by equation (5.5)

$$SSIM(\hat{y}, y) = \frac{(2\mu_{\hat{y}}\mu_y + c_1)(2\sigma_{\hat{y}y} + c_2)}{\left(\mu_{\hat{y}}^2 + \mu_y^2 + c_1\right)\left(\sigma_{\hat{y}}^2 + \sigma_y^2 + c_2\right)} \tag{5.5}$$

where μ and σ are average and variance, respectively. $\sigma_{\hat{y}y}$ is the covariance of the model prediction and the observation and c_1 and c_2 are the regularization constants.

In addition to SSIM, Peak Signal to Noise Ratio (PSNR) is another metric to evaluate visual similarities between two images. PSNR compares the image qualities using the following equation;

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [\hat{y}(i,j) - y(i,j)]^2 \tag{5.6}$$

$$PSNR = 10 \cdot \log_{10}(\frac{R^2}{MSE}) \tag{5.7}$$

where MSE is the mean square error and m and n are the image dimensions. $\hat{y}$ and y are the estimated image and reference image, respectively. R in equation 5.7 represents the maximum value of a pixel that can get.

Categorical Indices

Categorical metric are used to assess the models' capabilities in rainfall detection. The categorical indices that we use in this study are probability of detection(POD), false alarm ratio(FAR), and the critical success index(CSI). These indices are given by equations 5.8-5.9, respectively.

$$POD = \frac{TP}{TP + MS} \tag{5.8}$$

$$FAR = \frac{FP}{TP + FP} \tag{5.9}$$

$$CSI = \frac{TP}{TP + FP + MS} \tag{5.10}$$

where TP, MS, and FP are the number of True Positives, number of Missed, and the number of False Positive pixels, respectively. These numbers are calculated for each pair of model predicted and observed images.

5.3 Data and Case Study

5.3.1 Infrared Imageries

The input data used in this study is the long-wave IR channel with the wavelength of $10.8\mu m$ from the Geostationary Operational Environmental Satellite(GOES) series. The IR channel provides information on the cloud-top temperature which is a useful feature for identifying the precipitable clouds. The utilized IR data is provided by the Climatic Prediction Center (`http://www.cpc.ncep.noaa.gov/products/global_precip/html/wpage.merged_IR.html`) and has a near-global coverage(60°N-60°S) with a 0.04° × 0.04° spatial resolution. In this study, the IR data was regridded to $0.25^\circ \times 0.25^\circ$ resolution using the nearest neighbour algorithm to be consistent with the spatial resolution of PERSIANN algorithm.

5.3.2 Stage IV observation data

The ground truth precipitation dataset that we employ in this study is the quality controlled multi-sensor(radar and gauge) stage IV data(`https://www.emc.ncep.noaa.gov/mmb/ylin/pcpanl/stage4/`). This dataset is generated by the National Center for Environmental Prediction(NCEP) by mosaicking the rainfall observations obtained from the National Weather Service(NWS) River Forecast Centers(RFC). The stage IV data has a spatial resolution of $0.04^\circ \times 0.04^\circ$ and an hourly temporal resolution. Similarly, the stage IV data was regridded to $0.25^\circ \times 0.25^\circ$ resolution using the nearest neighbor method.

5.3.3 Data Sampling

This study aims to learn the precipitation patterns and the link between infrared (IR) information and rainfall rates, homogeneously and regardless of the type of precipitation and event location. In addition to this criteria, learning IR-precipitation is quite challenging, especially in the right tail of rainfall distribution where high rainfall samples are rare. This creates an unbalancy in the training data and intuitively, forces any model to estimate higher probability for no rain and lower probability for high rain rates. To address these important issues, instead of a fix case study area, a random selection procedure was used to randomly choose samples across the CONUS and time. The selected samples then go under quality control and only the ones with 25 percent and more rainfall pixels compared to the total window pixels are selected for training (Figure 5.1). In addition, the selected windows should not have any no value data either in IR or in radar data.

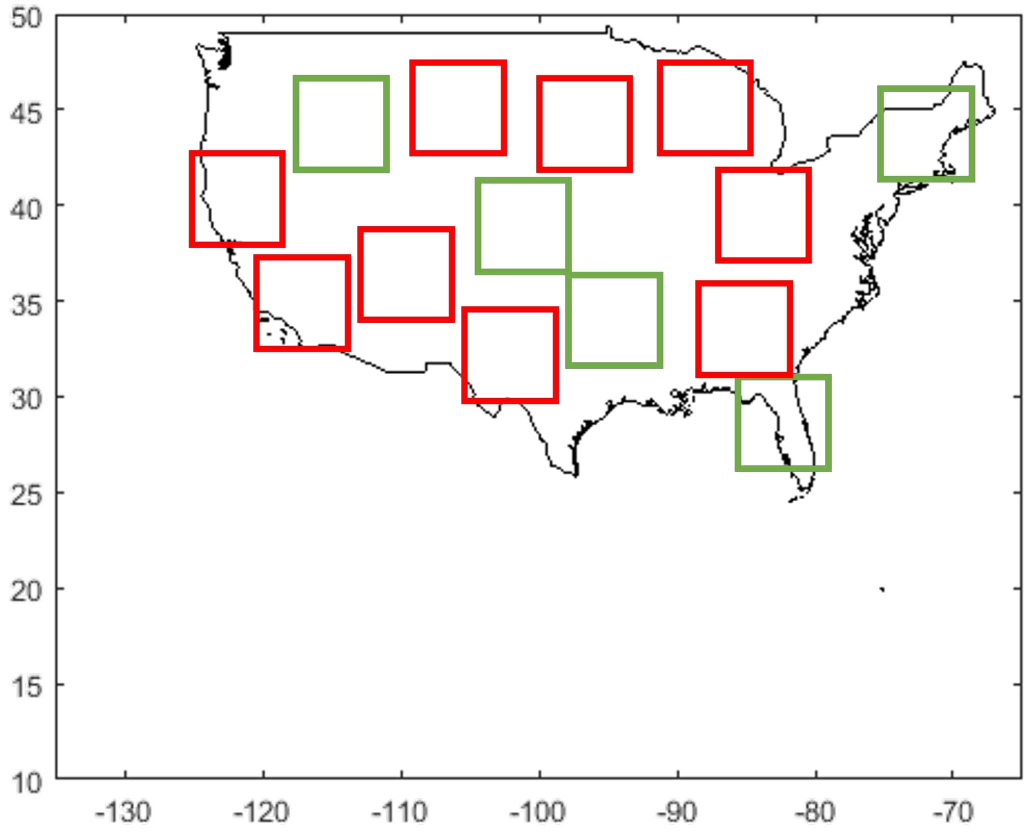


Figure 5.1: The random selection of sample patches for training process over the CONUS. The red rectangulars are the randomly selected samples which did not contain 25 percents or more rainy pixels. The green rectangulars are the samples which satisfied the rainfall criteria at each time step and are selected for training.

5.4 Results and Discussion

5.4.1 General Estimation skills

In Figure 5.2, the hourly, daily and monthly estimations (from left to right columns) are visualized for StageIV ground-truth observations, PERSIANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections. The results demonstrate higher similarities for PERISANN-GAN to StageIV in hourly, daily and monthly results. The results are selected from different events to visually evaluate the performance of each model compared to StageIV radar observations. The results, demonstrate higher similarities for PERSIANN-GAN and StageIV data in all hourly, daily and monthly scales. CNN with bypass connections also show consistent patterns in all scales, however, its intensities for the center of storms are not as close to StageIV observation as PERSIANN-GAN. Furthermore, PERSIANN overestimates the daily storm and shows larger high intensity patch than what is observed. In addition, low intensity rainfall over mid-south states (i.e. Colorado, New Mexico, and portion of Texas) for daily scale are well captured by PERSIANN-GAN, CNN with and without bypass connections, however, PERSIANN missed the rainfall especially over the state of Colorado. For the monthly scale, PERSIANN estimates are overestimating over mid-north states (i.e. Minnesota, Wisconsin, North Dakota, Michigan, Iowa, etc.). PERSIANN-GAN and CNN with bypass shows less overestimations over the mentioned area. Similar to PERSIANN, CNN without bypass connection also overestimates over the mid-north states.

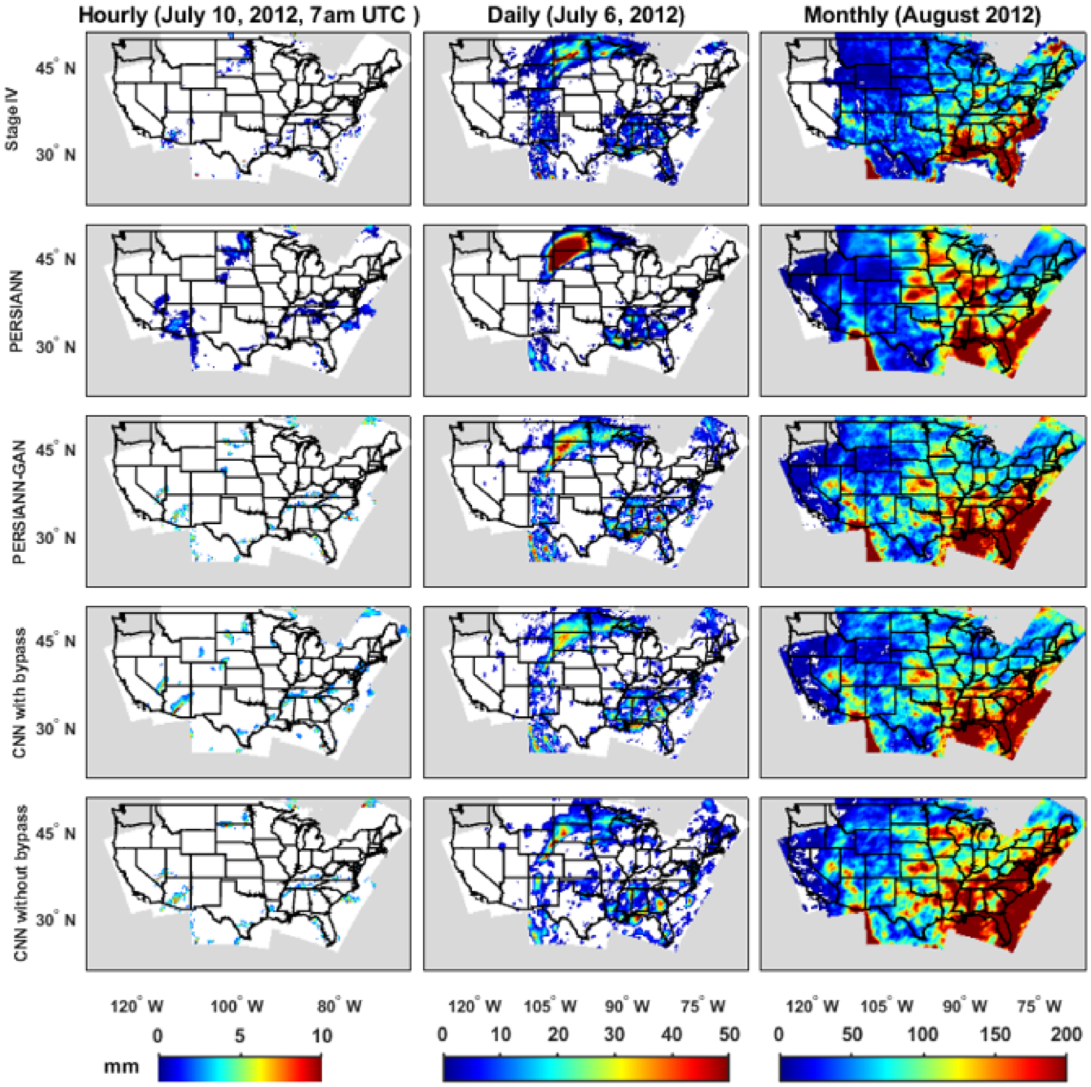


Figure 5.2: Visual performances of rainfall estimation in hourly, daily and monthly scales. Rows from top to bottom present StageIV, PERISANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections, respectively. The presented columns from left to right are for hourly, daily and monthly scales for July 10^{th} 2012, 07:00 AM UTC, July 6^{th} and August 2012, respectively.

Table 5.1 presents continuous and categorical metrics for evaluating daily precipitation estimates of PERISANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections over the CONUS for summer of 2012. The bold numbers indicate the best performance for each index in each row. Based on the evaluations in table 5.1, PERSIANN-GAN model has higher statistical performances compared to other baseline models including PERSIANN in terms of POD, FAR, CSI, Correlation coefficient, SSIM and PSNR. In terms of RMSE, CNN with bypass connections have better performances and this is due to optimizing CNN with bypass connections with RMSE, and PERSIANN-GAN is train on two different criteria. In addition, CNN with bypass has relatively better performances compared to PERSIANN and CNN without bypasses which for the case of CNN without bypasses, the results are expected.

Table 5.1: Statistical comparison of PERISANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections in daily scale over the CONUS. The metrics used to evaluate the models are Probability of Detection (POD), False Alarm Ratio (FAR), Critical Success Index (CSI), RMSEm Correlation Coefficient, Structure Similarity (SSIM) and Peak Signal to Noise (PSNR). The values in bold belong to best performing model in each row.

Metric	**PERSIANN-GAN**	**PERSIANN**	**CNN** **with bypass**	**CNN** **without bypass**
POD	**0.78**	0.74	0.77	0.66
FAR	**0.33**	0.34	0.37	0.41
CSI	**0.56**	0.53	0.53	0.45
RMSE	6.37	6.59	**6.06**	6.97
Correlation	**0.57**	0.55	0.54	0.44
SSIM	**37.82**	37.01	36.51	37.73
PSNR	**0.96**	0.95	0.93	0.94

5.4.2 Event-based comparison

To further evaluate the capabilities of PERSIANN-GAN in estimating accurate rainfall patterns and intensities, an extreme event, called hurricane Isaac, was studied during the study period. Hurricane Issac hit south-western US from August 28^{th} to September 4^{th} and resulted in significant amount of rainfall over several states including Louisiana, Mississippi, Alabama and etc. In this section, we compared the evolution of the heavy rainfall over the state of Alabama as an example to evaluate different model performances. Figure 5.3 demonstrates the visual patterns and intensities of rainfall over the state of Alabama. First, second, third, fourth and fifth rows belong to StageIV, PERSIANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections, respectively and columns from left to right, indicates the daily timesteps from August 28^{th} to September 4^{th}. The results show more consistency between the PERSIANN-GAN estimates and StageIV in terms of rainy patch shape and the intensities of rainfall, compared to other models. Specifically, the PERSIANN-GAN estimates capture better precipitation patterns, in comparison with StageIV observation, on August 29^{th} and 30^{th}, and September 3^{rd} and 4^{th}. Furthermore, PERSIANN-GAN captures the center of heavy rainfall correctly on August 30^{th} and September 4^{th}, where other models fail to locate the center of storm.

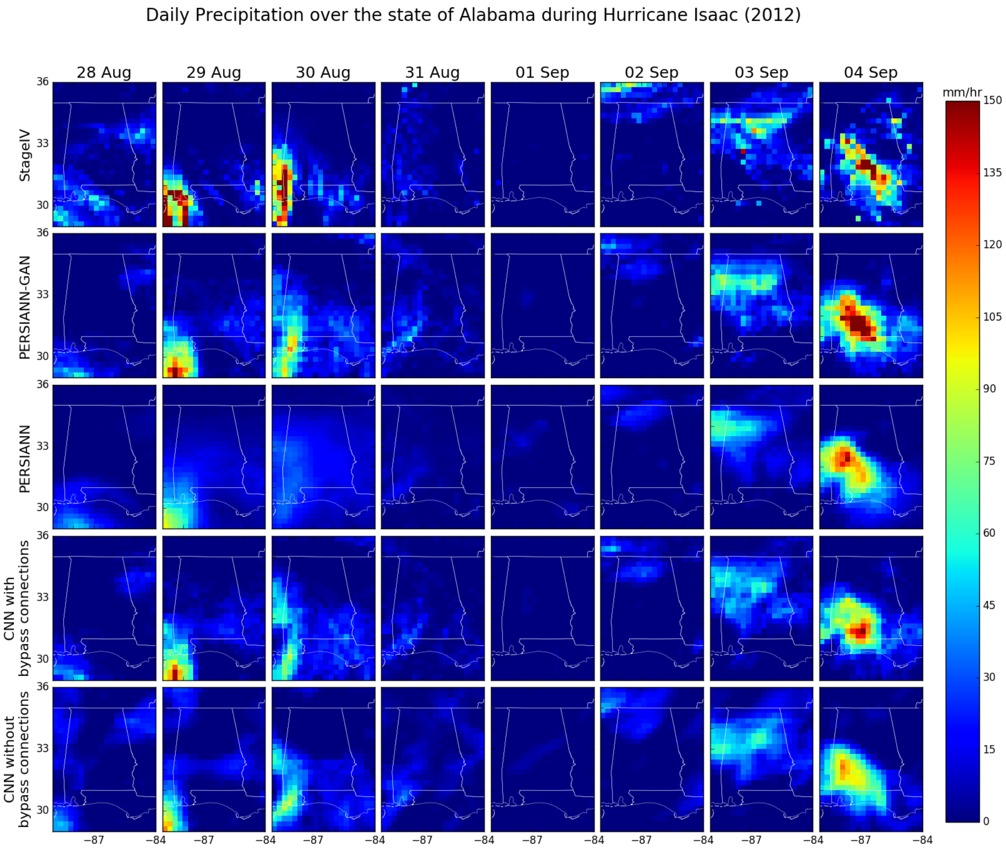

Daily Precipitation over the state of Alabama during Hurricane Isaac (2012)
28 Aug
29 Aug
30 Aug
31 Aug
01 Sep
02 Sep
03 Sep
04 Sep
StageIV
PERSIANN-GAN
PERSIANN
CNN with bypass connections
CNN without bypass connections
36
33
30
−87
−84
mm/hr
150
135
120
105
90
75
60
45
30
15
0

Figure 5.3: Evolution of extreme rainfall is demonstrated at each row for StageIV observation, PERSIANN-GAN, PERSIANN, CNN with bypass connections and CNN without bypass connections, respectively. The rainfall evolution is presented in daily scale starting from Aug. 28th to Sept. 4th.

To further elaborate on the capabilities of each model in capturing hurricane Isaac, the mean of 6-hourly precipitation for StageIV, PERSIANN and PERSIANN-GAN are calculated for the period of storm (Figure 5.4a). The results show consistency of PERSIANN-GAN with StageIV and the 6-hourly intensities match the StageIV results. Furthermore, Figure 5.4b shows the Structure Similarity (SSIM) index, which is a criteria for visual similarities, for PERSIANN-GAN (red) and PERSIANN (green), both compared to StageIV data. The results indicate equal or higher visual similarities for PERSIANN-GAN in comparison with PERSIANN for 6-hourly rainfall over state of Alabama. Figure 5.4c presents the RMSE of PERSIANN-GAN (red) and PERSIANN (green) compared to StageIV. The results show similar performances for both PERISANN-GAN and PERSIANN during the whole period of hurricane Isaac. In general, the results presented in Figure 5.4 directly indicates higher visual consistency of PERSIANN-GAN to StageIV in no cost for RMSE metric.

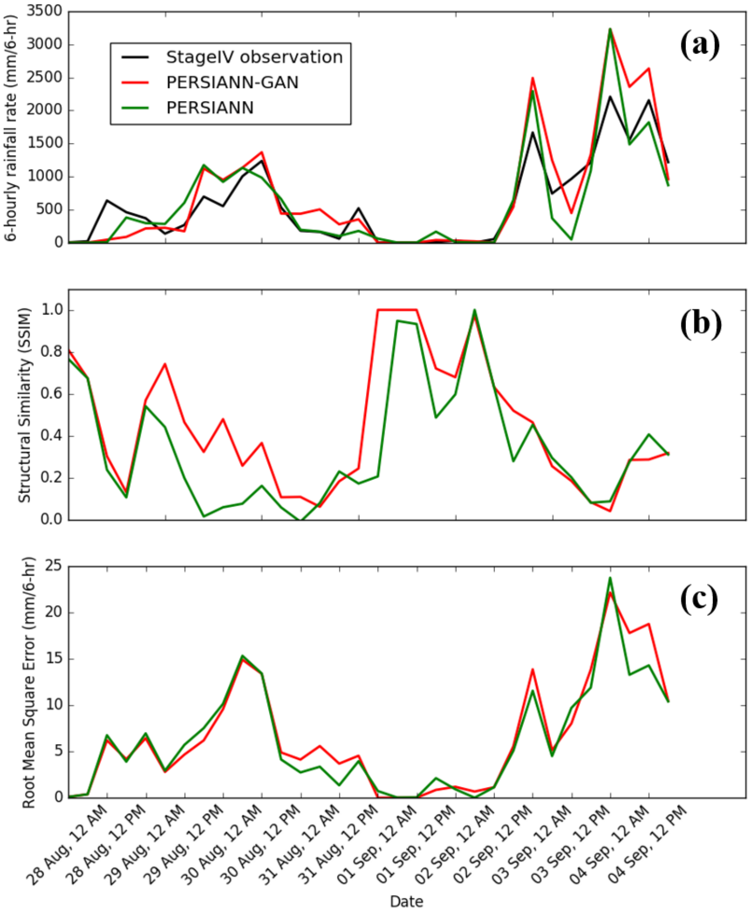


Figure 5.4: Statistical comparison of hurricane Isaac over the state of Alabama: (a) 6-hourly mean of precipitation for StageIV (black), PERSIANN (green) and PERSIANN-GAN (red), (b) SSIM metric of 6-hourly results for PERSIANN (green) and PERSIANN-GAN (red), and (c) RMSE results for 6-hourly precipitations for PERSIANN (green) and PERSIANN-GAN (red).

5.5 Summary and Conclusion

In this chapter of the dissertation, I proposed a framework to estimate precipitation intensity rates from infrared cloud-top brightness temperature imageries of GOES satellites using a newly introduced framework termed Generative Adversarial Networks. In this framework, I conditioned the model on infrared images in order to construct an image to image translation scheme. A U-net structure was implemented within the generator to ensure the model does not throw out spatial information due to the subsampling process. The proposed model, named PERSIANN-GAN, performances were investigated over the CONUS for the summer months of 2012 (July-August-September) by comparing to two baseline CNNs similar to the generator model, with and without bypass connections, and the PERSIANN estimates. The results are compared in hourly, daily and monthly scales and PERSIANN-GAN demonstrated higher visual similarities to stageIV radar. Furthermore, I investigated the performance of PERSIANN-GAN in detecting extreme event, Isaac, occurring over the state of Alabama. The visual and statistical performances such as RMSE and SSIM indicated closer detection patterns of PERSIANN-GAN to stageIV radar observations. Finally, an overall statistical comparison was conducted using POD, FAR, CSI, SSIM, PSNR, RMSE and correlation coefficient for PERSIANN-GAN, PERSIANN, CNN with bypass and CNN without bypass. The results indicated better detection capabilities for PERSIANN-GAN. The high level conclusion highlights of this study are

- Based on the conducted experiments, the proposed framework to estimate precipitation is enable to accurately detect wide intensity range of precipitation events using infrared imageries.

- GAN architecture efficiently and effectively translates infrared to precipitation and demonstrate accurate rainfall detection, especially by determining correct rainfall intensities.

- The advantage of the proposed estimation model (i.e., PERSIANN-GAN) is to adopt a mixture of a flexible learning divergence and maximum likelihood instead of only estimating maximum likelihood. The flexibility of the objective function allows the model to overcome maximum likelihood challenges and generate more realistic precipitation estimations.

- Acquiring U-net type of structure for generator model allowed increase the accuracy of estimations, as also seen in CNN with bypass. Higher accuracy is a result of preserving spatial information in the model after applying subsampling operations.

Chapter 6

Conclusions and future directions

6.1 Summary of findings

Main contributions of this dissertation are as follows:

- Proposing a state-of-the-art deep learning algorithm to forecast infrared imageries from 0 to 6 hours, with potentials of becoming an operational product over quasi-global scale.
- Introduced a novel generative framework to accurately translate infrared information to rainfall rates using a flexible approach in defining the objective function.
- Developed a short-term precipitation forecasting framework by effectively breaking the tasks of forecasting and estimation to reduce the problem complexity and increase the model understanding of the problem.
- Proposed and investigated a new probabilistic approach in computer science to overcome the complex earth system science problems such as precipitation forecasts.

In details, the objectives of this dissertation outlined in Chapter 1 are addressed in this dissertation:

- ***Investigating the effectiveness of deep learning approaches for extracting***

spatial and temporal features from image sequences of satellite data and obtaining accurate and timely infrared forecasts.

This objective was first explored in chapter 2 by introducing an infrared forecasting scheme using LSTM and further advanced in chapter 4 by proposing a novel semi-conditional GAN model for image sequence forecasting. Both study, demonstrated that advanced and recently invented deep learning techniques can efficiently and effectively address complex spatial, temporal and spatiotemporal processes within earth system components.

- ***Demonstrating the role of Recurrent Neural Networks and its recent variants (Long Short-Term Memory) in forecasting accurate infrared imageries.***

  In chapters 2 and 4, the role of recurrent layers, especially advanced and sophisticated layers such as LSTM, can improve the predictability skills of precipitation compared to traditional methods such as Optical Flow techniques, NWP model outputs and conventional methods such as Persistency. Chapter 2 thoroughly discusses the analogy of the superiority of LSTM model to conventional RNNs and chapter 4 discusses a combination of LSTM structure with convolutional scheme to join the advantages of two techniques.

- ***Evaluate the proposed methodology and the effectiveness of using new deep learning techniques compared to earlier generations.***

  This objective was thoroughly discussed in chapters 2, 4 and 5 by evaluating advanced deep learning techniques over states of Oregon, Oklahoma and Florida and CONUS. The evaluations included visual and statistical comparisons to comprehensively examine the capability of the proposed advanced deep learning models in accurately generalizing features to other unseen study areas.

- ***Introduce a new approach for image to image translation by reconsidering***

the use of maximum likelihood estimation and developing a more flexible approach in generative neural networks.

Chapter 3 comprehensively discussed the challenges associated with using maximum likelihood estimation for generative models and discussed the deficiencies of traditional objective functions such as Mean Squared Error and Kullback-Leibler divergence which aims to satisfy the maximum likelihood estimation criteria suffer from blurred and inaccurate output image generation. In the same chapter, we discussed the new approach vastly used in the field of computer science named Generative Adversarial Networks which tackles this shortcoming by defining learned divergence to evaluate generative model behavior.

- ***Assess the improvements of new generative models in infrared forecasting task and explore the capability of sub-location learning in generalizing learnt features.***

  This objective was addressed in chapters 4 and 5 by assessing the generative approaches for infrared forecasting and precipitation estimation comparing to same-design non-adversarial models over the CONUS extend. In both studies, the new generative approach demonstrated superior capabilities to its non-adversarial type approaches.

- ***Introduce a new framework to translate infrared imageries to rainfall intensity maps using a network learning from distribution and maximum likelihood of the data together.***

  Chapter 5 discussed the new approach in which the model learns both from maximum likelihood estimates and distribution matching techniques simultaneously to estimate accurate intensities along with correct shape of the rainfall patches. The combinatorial objective function allows the model to fix the underestimations and overestimations frequently observed in various satellite datasets.

6.2 Future extensions

The developed forecasts models in chapters 2, 4 and the precipitation estimation model introduced in chapter 5 have offers many potential extensions for future research. In particular, the possible extensions of this research area can be explored by: (1) including additional data sources, (2) improving model structure, and (3) putting the models in operational use.

6.2.1 Additional Data Sources

Additional information sources can increase the reasoning capabilities of the forecast and estimation models by having a better understanding of the physical process. Particularly for the forecasting algorithm, more information can be gathered and used in the framework from satellites such as wind sub-products of imager and sounder sensors of GOES satellites, High Density Water Vapor Cloud Drift Winds from MODIS satellites and from NWP models for their certain components such as Geo-Potential Heights, Specific Humidity, Total Column Liquid Water, Total Column Ice Water, Total Column Water Vapor, 10-meter vertical velocity and etc. In addition, surface characteristics, such as digital elevation and land/ocean discrimination can be very helpful for the training and learning processes of rainfall estimation and short-term forecast [Hsu et al., 1997, Gesch et al., 1999].

Furthermore, sudden shifts in cloud temperature often yield high intensity precipitation and using consecutive infrared time frames can improve rainfall detection accuracy.

Additionally, leveraging high temporal and spatial resolution is another improvement strategy in which the physical processes can be best captured. This solution against the attempt to learning estimation and/or forecasting from half-hourly, 0.25° or coarser resolutions where physical processes have been dampened significantly.

In addition to satellite information, ground observations are another important source of information to obtain more accurate estimates and forecasts [Katiraie-Boroujerdy et al., 2017], and in-situ observations can be used for bias-correction of satellite product [Yang et al., 2016].

6.2.2 Model Development

High-paced advancements of deep learning opens room for new innovations in the field of hydrometeorology and remote sensing, especially precipitation estimation and forecasting. In infrared-rainfall image translation models, developing a model attention mechanism can significantly improve the accuracy of the model, since the direct relation of coldness of clouds and rainfall rates have been proven [Adler et al., 1993, Arkin, 1979]. Visual attention mechanism can force the model to focus on the portions of infrared images that yield the highest error during training phase, and resulting in an improved and more robust estimation algorithm.

Another improvement strategy is to equip the estimation model with recurrent layers in order to capture sudden shift rainfalls already described in previous section.

6.2.3 Operational Applications

The models introduced in chapters 4 and 5 are Fully Convolutional Networks with the flexibility of taking any shape of images as input. This advantage of FCNs allow the models to easily become operational products with minimal effort. The presented results over the CONUS as a proof of robustness demonstrate that these models can generalize from local and CONUS scale into quasi-global scale.

Moreover, in order to further investigate the effectiveness of the introduced models, hydrological applications of the model outputs should be tested, especially for precipitation forecast products, in order to verify the reliability of the products in user-end applications. Lastly, the model performances should be tested for extreme precipitation events and hurricanes, which are mandatory for disaster monitoring and preparation.

Bibliography

R. F. Adler, A. J. Negri, P. R. Keehn, and I. M. Hakkarinen. Estimation of monthly rainfall over japan and surrounding waters from a combination of low-orbit microwave and geosynchronous ir data. *Journal of Applied Meteorology*, 32(2):335–356, 1993.

A. AghaKouchak, A. Behrangi, S. Sorooshian, K. Hsu, and E. Amitai. Evaluation of satellite-retrieved extreme precipitation rates across the central united states. *Journal of Geophysical Research: Atmospheres*, 116(D2), 2011.

P. A. Arkin. The relationship between fractional coverage of high cloud and rainfall accumulations during gate over the b-scale array. *Monthly Weather Review*, 107(10):1382–1387, 1979.

P. A. Arkin and B. N. Meisner. The relationship between large-scale convective rainfall and cold cloud over the western hemisphere during 1982-84. *Monthly Weather Review*, 115(1): 51–74, 1987.

P. A. Arkin, R. Joyce, and J. E. Janowiak. The estimation of global monthly mean rainfall using infrared satellite data: The goes precipitation index (gpi). *Remote Sensing Reviews*, 11(1-4):107–124, 1994.

S. P. Ballard, Z. Li, D. Simonin, and J.-F. Caron. 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, 2016.

A. Behrangi, K.-l. Hsu, B. Imam, S. Sorooshian, G. J. Huffman, and R. J. Kuligowski. Persiann-msa: A precipitation estimation method from satellite-based multispectral analysis. *Journal of Hydrometeorology*, 10(6):1414–1429, 2009.

S. G. Benjamin, S. S. Weygandt, J. M. Brown, M. Hu, C. R. Alexander, T. G. Smirnova, J. B. Olson, E. P. James, D. C. Dowell, G. A. Grell, et al. A north american hourly assimilation and model forecast cycle: The rapid refresh. *Monthly Weather Review*, 144 (4):1669–1694, 2016.

V. Bližňák, Z. Sokol, and P. Zacharov. Nowcasting of deep convective clouds and heavy precipitation: comparison study between nwp model simulation and extrapolation. *Atmospheric Research*, 184:24–34, 2017.

D. R. Bright and S. L. Mullen. Short-range ensemble forecasts of precipitation during the southwest monsoon. *Weather and Forecasting*, 17(5):1080–1100, 2002.

W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki. Scene labeling with lstm recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3547–3555, 2015.

K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in neural information processing systems*, pages 2843–2851, 2012.

J. T. Connor, R. D. Martin, and L. E. Atlas. Recurrent neural networks and robust time series prediction. *IEEE Transactions on Neural Networks*, 5(2):240–254, March 1994. ISSN 1045-9227. doi: 10.1109/72.279188.

D. Eck and J. Schmidhuber. A first look at music composition using lstm recurrent neural networks. *Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale*, 103, 2002.

J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer, 2003.

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. *Hydrology and Earth System Sciences*, 20(1):505, 2016.

R. G. Fovell. Consensus clustering of us temperature and precipitation data. *Journal of Climate*, 10(6):1405–1427, 1997.

M. N. French, W. F. Krajewski, and R. R. Cuykendall. Rainfall forecasting in space and time using a neural network. *Journal of hydrology*, 137(1-4):1–31, 1992.

A. R. Ganguly and R. L. Bras. Distributed quantitative precipitation forecasting using information from radar and numerical weather prediction models. *Journal of Hydrometeorology*, 4(6):1168–1180, 2003.

F. A. Gers and J. Schmidhuber. Recurrent nets that time and count. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, volume 3, pages 189–194. IEEE, 2000.

F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with lstm. 1999.

D. B. Gesch, K. L. Verdin, and S. K. Greenlee. New land surface digital elevation model covers the earth. *EOS, Transactions American Geophysical Union*, 80(6):69–70, 1999.

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

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

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

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

A. Graves and J. Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5-6):602–610, 2005.

K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10): 2222–2232, 2017.

J. Guo. Backpropagation through time. *Unpubl. ms., Harbin Institute of Technology*, 2013.

T. Hall, H. E. Brooks, and C. A. Doswell III. Precipitation forecasting using a neural network. *Weather and forecasting*, 14(3):338–345, 1999.

A. Heye, K. Venkatesan, and J. Cain. Precipitation nowcasting: Leveraging deep recurrent convolutional neural networks. 2017.

R. Higgins, K. Mo, and Y. Yao. Interannual variability of the us summer precipitation regime with emphasis on the southwestern monsoon. *Journal of Climate*, 11(10):2582–2606, 1998.

S. Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116, 1998.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997a.

S. Hochreiter and J. Schmidhuber. Lstm can solve hard long time lag problems. In *Advances in neural information processing systems*, pages 473–479, 1997b.

S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber, et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001.

K.-l. Hsu, X. Gao, S. Sorooshian, and H. V. Gupta. Precipitation estimation from remotely sensed information using artificial neural networks. *Journal of Applied Meteorology*, 36(9): 1176–1190, 1997.

K.-l. Hsu, H. V. Gupta, X. Gao, S. Sorooshian, and B. Imam. Self-organizing linear output map (solo): An artificial neural network suitable for hydrologic modeling and analysis. *Water Resources Research*, 38(12):38–1, 2002.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

P.-S. Katiraie-Boroujerdy, N. Nasrollahi, K.-l. Hsu, and S. Sorooshian. Evaluation of satellite-based precipitation estimation over iran. *Journal of arid environments*, 97:205–219, 2013.

P.-S. Katiraie-Boroujerdy, A. A. Asanjan, K.-l. Hsu, and S. Sorooshian. Intercomparison of persiann-cdr and trmm-3b42v7 precipitation estimates at monthly and daily time scales. *Atmospheric research*, 193:36–49, 2017.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69, 1982.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

R. J. Kuligowski and A. P. Barros. Localized precipitation forecasts from a numerical weather prediction model using artificial neural networks. *Weather and forecasting*, 13(4):1194–1204, 1998.

V. Lakshmanan, A. Fritz, T. Smith, K. Hondl, and G. Stumpf. An automated technique to quality control radar reflectivity data. *Journal of applied meteorology and climatology*, 46 (3):288–305, 2007.

Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

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

X. Liu, T. Yang, K. Hsu, C. Liu, and S. Sorooshian. Evaluating the streamflow simulation capability of persiann-cdr daily rainfall products in two river basins on the tibetan plateau. *Hydrology and Earth System Sciences (Online)*, 21(1), 2017.

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

M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

S. Moazami, S. Golian, M. R. Kavianpour, and Y. Hong. 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, 2013.

S. Nowozin, B. Cseke, and R. Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279, 2016.

D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

B. Ritter and J.-F. Geleyn. A comprehensive radiation scheme for numerical weather prediction models with potential applications in climate simulations. *Monthly Weather Review*, 120(2):303–325, 1992.

T. G. Romilly and M. Gebremichael. Evaluation of satellite rainfall estimates over ethiopian river basins. *Hydrology and Earth System Sciences*, 15(5):1505–1514, 2011.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers. Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 556–564. Springer, 2015.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

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

S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? In *Advances in Neural Information Processing Systems*, pages 2483–2493, 2018.

H. Sauvageot. Rainfall measurement by radar: A review. *Atmospheric Research*, 35(1): 27–54, 1994.

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

L. C. Smith, B. L. Isacks, A. L. Bloom, and A. B. Murray. Estimation of discharge from three braided rivers using synthetic aperture radar satellite imagery: Potential application to ungaged basins. *Water Resources Research*, 32(7):2021–2034, 1996.

N. Srivastava, E. Mansimov, and R. Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015.

J. Sun, M. Xue, J. W. Wilson, I. Zawadzki, S. P. Ballard, J. Onvlee-Hooimeyer, P. Joe, D. M. Barker, P.-W. Li, B. Golding, et al. Use of nwp for nowcasting convective precipitation: Recent progress and challenges. *Bulletin of the American Meteorological Society*, 95(3): 409–426, 2014.

M. Sundermeyer, R. Schlüter, and H. Ney. Lstm neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*, 2012.

L. Theis, A. v. d. Oord, and M. Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

S. V. Vasiloff, D.-J. Seo, K. W. Howard, J. Zhang, D. H. Kitzmiller, M. G. Mullusky, W. F. Krajewski, E. A. Brandes, R. M. Rabin, D. S. Berkowitz, et al. 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, 2007.

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

G. Wang, J. Yang, D. Wang, and L. Liu. 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, 2016.

T. T. Warner, R. A. Peterson, and R. E. Treadon. 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):2599–2618, 1997.

P. J. Werbos. Generalization of backpropagation with application to a recurrent gas market model. *Neural networks*, 1(4):339–356, 1988.

R. J. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 461–470. ACM, 2015.

S. Xingjian, 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 *Advances in neural information processing systems*, pages 802–810, 2015.

L. Xu, X. Gao, S. Sorooshian, P. A. Arkin, and B. Imam. A microwave infrared threshold technique to improve the goes precipitation index. *Journal of Applied Meteorology*, 38(5): 569–579, 1999.

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

T. Yang, A. A. Asanjan, E. Welles, X. Gao, S. Sorooshian, and X. Liu. Developing reservoir monthly inflow forecasts using artificial intelligence and climate phenomenon information. *Water Resources Research*, 53(4):2786–2812, 2017b.

T. Yang, Y. Tao, J. Li, Q. Zhu, L. Su, X. He, and X. Zhang. Multi-criterion model ensemble of cmip5 surface air temperature over china. *Theoretical and applied climatology*, 132(3-4): 1057–1072, 2018.

Z. Yang, K. Hsu, S. Sorooshian, X. Xu, D. Braithwaite, and K. M. Verbist. Bias adjustment of satellite-based precipitation estimation using gauge observations: A case study in chile. *Journal of Geophysical Research: Atmospheres*, 121(8):3790–3806, 2016.

A. Zahraei, K.-l. Hsu, S. Sorooshian, J. Gourley, V. Lakshmanan, Y. Hong, and T. Bellerby. Quantitative precipitation nowcasting: a lagrangian pixel-based approach. *Atmospheric Research*, 118:418–434, 2012.

A. Zahraei, K.-l. Hsu, S. Sorooshian, J. J. Gourley, Y. Hong, and A. Behrangi. Short-term quantitative precipitation forecasting using an object-based approach. *Journal of hydrology*, 483:1–15, 2013.

J. Zhang, K. Howard, C. Langston, S. Vasiloff, B. Kaney, A. Arthur, S. Van Cooten, K. Kelleher, D. Kitzmiller, F. Ding, et al. National mosaic and multi-sensor qpe (nmq) system: Description, results, and future plans. *Bulletin of the American Meteorological Society*, 92 (10):1321–1338, 2011.

Q. Zhu, K.-l. Hsu, Y.-P. Xu, and T. Yang. 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, 2017.