DISSERTATION

GREMLIN: GOES RADAR ESTIMATION VIA MACHINE LEARNING TO INFORM NWP

Submitted by

Kyle Aaron Hilburn

Department of Atmospheric Science

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2023

Doctoral Committee:

Advisor: Steven D. Miller

Christian D. Kummerow
Elizabeth A. Barnes
Imme Ebert-Uphoff
Curtis R. Alexander

Copyright by Kyle Aaron Hilburn 2023

All Rights Reserved

ABSTRACT

GREMLIN: GOES RADAR ESTIMATION VIA MACHINE LEARNING TO INFORM NWP

Imagery from the Geostationary Operational Environmental Satellite (GOES) has been a key element of U.S. operational weather forecasting since 1975. The latest generation, the GOES-R Series, offers new capabilities to support the need for high-resolution rapidly refreshing imagery for situational awareness. Despite the well demonstrated value to human forecasters, usage of GOES imagery in data assimilation (DA) for initializing numerical weather prediction (NWP) has been limited, particularly in cloudy and precipitating scenes. By providing a rich and powerful library of nonlinear statistical tools, artificial intelligence (AI) / machine learning (ML) enables new approaches for connecting models and observations. The objective of this research is to develop techniques for assimilating GOES-R Series observations in precipitating scenes for the purpose of improving short-term convective-scale forecasts of high-impact weather hazards. The hypothesis of this dissertation is that by harnessing the power of ML, the new GOES-R capabilities can be used to create equivalent radar reflectivity suitable for initializing convection in high-resolution NWP models.

Chapter 1 will present a proof-of-concept that ML can be used as an observation operator for GOES-R to simulate Multi-Radar Multi-Sensor (MRMS) composite reflectivity data and therby initialize convection in NOAA's Rapid Refresh and High-Resolution Rapid Refresh (RAP/HRRR). Development of the GREMLIN (GOES Radar Estimation via Machine Learning to Inform NWP) convolutional neural network (CNN) will be described. This includes the creation of a hierarchy of open source datasets, and will emphasize the importance of the neural

network loss function in focusing the attention of the network on the most important meteorological features. Explainable AI (XAI) tools are applied to GREMLIN to discover three primary strategies employed by the network in making predictions, highlighting the unique ability of CNNs to utilize spatial context in satellite imagery. The results of retrospective Rapid Refresh Forecast System (RRFS) forecasts will be described, which show that GREMLIN can produce more accurate short-term forecasts than using real radar data over areas of the U.S. with poor radar coverage.

In Chapter 2, the Interpretable GREMLIN model is developed to elucidate the nature of the spatial context utilized by CNNs to make accurate predictions. This clarity is accomplished by moving the inner workings of the CNN out into a feature engineering step and replacing the neural network with a linear regression model. This exposes the effective input space of the CNN and establishes well defined relationships between inputs and outputs, which provides guarantees on how the model will respond to novel inputs. Despite a 24x reduction in the number of trainable parameters, the interpretable model has similar accuracy as the original CNN. Using the interpretable model, five additional physical strategies missed by XAI are discovered. The pros and cons of interpretable model development and implications for generalizability, consistency, and trustworthy AI will be discussed.

Finally, Chapter 3 will extend this research for the development of Global GREMLIN, discussing the challenges and opportunities. GREMLIN is validated for regimes outside of the training dataset, and regime dependence is quantified in terms of temperature and moisture. The impacts of additional predictors and advanced ML architectures, and the derivation of uncertainty estimates that will be needed for new DA approaches in RRFS, will be discussed.

Current efforts to implement GREMLIN on NOAA's GeoCloud, which will make GREMLIN available to a broader base of users, will be described.

ACKNOWLEDGEMENTS

I would first like to thank Chelle Gentemann and Chris Kummerow for the conversation they had in 2015 that ultimately led me to CIRA and completing a Ph.D. at CSU. Thank you to my advisor Steve Miller for always bringing things back to the physical science of remote sensing, which helped me go beyond a black-box machine learning model and gain new insights. Support of the GOES-R Program under grant NA19OAR4320073, and of the NOAA RDHPCS for access to the Fine Grain Architecture System on Hera, are gratefully acknowledged.

PREFACE

It is hard not to think that we are living in a golden age in satellite meteorology. The GOES-R Series has advanced the state-of-the-art for Earth observation from geostationary orbit to a spatial resolution of 500 m and a temporal resolution of 10 minutes for the Full Disk and 30 seconds for the highest impact events. The Advanced Baseline Imager (ABI) on the GOES-R Series provides observations at new wavelengths that address limitations of previous sensors. Moreover, similar sensors on geostationary satellites operated by other nations (e.g., Japan's Himawari, Korea's GEO-KOMPSAT, and EU's Meteosat, among others) are now providing a "geo-ring" of high spatial and temporal resolution observations over much of the Earth. The GOES-R Series has also led the way to demonstrate lightning observations from geostationary orbit. Thus, meteorologists today are blessed with a wealth of new information that is highly relevant to situational awareness and short-term forecasting of high impact weather hazards.

Complementing these observational gains are advances in computing power and Artificial Intelligence (AI) algorithms. Traditionally, satellite retrieval algorithms have been limited to operating on a pixel-wise basis using physical or statistical methods, and that space/time information was only used crudely. The power of AI is that it gives retrieval algorithms the ability to extract the information content in the spatial and temporal patterns in satellite imagery. Humans have been making subjective use of this information for decades, but it wasn't until the development of convolutional neural networks (CNNs) that it became possible to derive objective algorithms that can use this information without a human-in-the-loop. This capability opens a whole new set of possibilities for machines to help humans get more value from the wealth of satellite observations, and to reduce the burden on human forecasters coming from the

"firehose" of data, thus providing more time for humans to focus on decision support services. The underlying philosophy of this dissertation is that transformative advancements to weather operations can be driven by the combination of AI, big data, and cloud computing (Figure P.1). By following this approach, new applications, products, and services can be developed in support of a Weather Ready Nation.

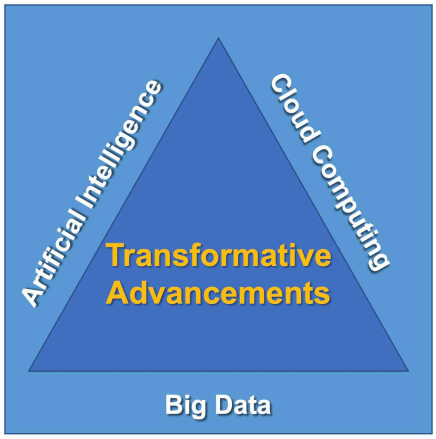

Figure P.1. The transformative advancements triangle, adapted from NOAA's Strategic Plan (https://sciencecouncil.noaa.gov/NOAA-Science-Technology-Focus-Areas).

However, the power of AI also brings risks that are only beginning to be understood. AI algorithms can tend to be "brittle," meaning that they can fail in unexpected ways when encountering new (or, "novel") data. AI algorithms can also tend to be "twitchy," meaning that while the algorithms accurately reproduce the training data in a mean sense, they may not exhibit realistic temporal variability (e.g., continuity). Finally, the more advanced AI algorithms lack transparency, meaning it can be difficult to impossible to understand why an AI algorithm makes the prediction that it does. This obscurity is a major obstacle for development of the trustworthy AI needed to support human decision making.

This dissertation addresses both the opportunities and the challenges presented by the application of AI to environmental remote sensing of Earth. Chapter 1 highlights the power of CNNs to extract spatial patterns from imagery and perform data fusion of radiances and

lightning. Explainable AI (XAI) techniques are used to uncover some of the strategies learned by AI. However, XAI leaves many questions unanswered, and so in Chapter 2, the explainability is built into the AI right from the start – resulting in an interpretable model. This yields new insights into the elusive nature of the "spatial context" used by CNNs and humans when interpreting satellite imagery. Chapter 3 lays the groundwork for extending the AI model to operate globally and demonstrates that it is possible to interpret AI to gain physical insights. Finally, the future of GREMLIN is discussed in Chapter 4.

TABLE OF CONTENTS

CHAPTER 1: GREMLIN PROOF-OF-CONCEPT[1]

1.1. Introduction

Geostationary Operational Environmental Satellite (GOES) imagery is a key element of U.S. operational weather forecasting, supporting the need for high-resolution, rapidly refreshing imagery for situational awareness (*Line et al. 2016*). While used extensively by human forecasters, its usage in data assimilation (DA) for Numerical Weather Prediction (NWP) models is limited. Instead, DA makes greater usage of microwave and infrared sounder data on low Earth orbiting satellites (*Lin et al. 2017*). Sounders provide more vertically resolved information than imagers, which is advantageous for characterizing the three-dimensional model state, but are carried almost exclusively on low-earth orbiting satellites—providing global coverage but at the expense of coarse temporal resolution and latency from sensor to NWP center that can reach 1.5 hr or more. Geostationary imagers provide much faster temporal refresh (now 10 minutes for Full Disk and 5 minutes over CONUS) and very low latency over a limited field of regard. Thus, there is an opportunity for operational DA to benefit from the high volume of low-latency, complementary data coming from the global constellation of geostationary imagers.

Operational DA for convective-scale NWP has made steady scientific advances (*Gustafsson et al. 2018*), but all-sky assimilation of infrared radiances has yet to be operationally demonstrated (*Geer et al. 2018*). This means that the most dynamic areas from the standpoint of precipitation, having significant impacts on human activities, are also the areas that have the least

---

[1] Chapter 1 includes content from:
Hilburn, K. A., I. Ebert-Uphoff, and S. D. Miller, 2021: Development and interpretation of a neural network-based synthetic radar reflectivity estimator using GOES-R satellite observations. *J. Appl. Meteor. Climatol.*, **60**, 3-21, https://doi.org/10.1175/JAMC-D-20-0084.1.
© American Meteorological Society. Used with permission.

amount of data to constrain estimates of the current atmospheric state. One approach is radiance assimilation (RA), which has the advantage of being physically based, making it simpler to interpret. *Okamoto et al. (2018)*, *Honda et al. (2018a,b)* and *Sawada et al. (2019)* tested assimilation of Himawari-8 water vapor absorption bands, finding improvements for heavy rain cases. *Otkin and Potthast (2019)* assimilate a water vapor band on SEVIRI, finding that the all-sky radiance bias correction is critical to making a positive impact on analyses. Demonstration of GOES-16 Advanced Baseline Imager (ABI) RA was provided by *Zhang et al. (2018, 2019)* and *Jones et al. (2020)*. These studies make different assumptions about how to inflate observation and background errors and how to weight information in the vertical. Errors in model microphysics and radiative transfer will be inherited by RA, and the land surface will come into play for window channels. *Jones et al. (2020)* find improved convective initiation forecasts with all-sky RA, but their best results come from using clear-sky radiances and cloud property retrievals. RA cannot be used for assimilating lightning observations, so an observation operator is required to convert those observables into control variable increments. *Kong et al. (2020)* demonstrate improvements from assimilating GOES Lightning Mapper (GLM), using an observation operator that takes advantage of the strong physical relationship between lightning and graupel mass and volume.

A limitation of infrared RA in cloudy and precipitating pixels is saturation of information content. For GOES ABI, the information content of individual pixels saturates around optical depths of 160 (8) during day (night), which are the maximum values reported by the retrieval algorithm (*Walther et al. 2013*). For warm season convection over CONUS, we find these values roughly correspond to composite reflectivity (REFC, the vertical maximum radar reflectivity in the column) of 20-25 (0-5) dBZ during day (night) (*Rutledge et al. 2020*). This truncated

sensitivity means, in turn, that infrared RA holds only limited information about precipitating scenes. This limitation is also present with physically based cloud property retrievals (*Jones et al. 2015*). The machine learning (ML) technique of convolutional neural networks (CNNs) has the advantage of using the information content present in image gradients, which we will show provides reliable information content up to REFC of about 50 dBZ. Moreover, ML provides an effective framework for using lightning information together with radiance information. So, in this work ML serves as an observation operator for DA, but there are many other potential applications of ML to DA, for example quality-control, bias correction, observation thinning, and post-processing to name a few. This unique ability of CNNs to capture spatial information – together with the large quantity, high quality, high resolution, and low latency of GOES-R data – is justification for exploring the capabilities of ML to enhance DA. Moreover, human forecasters are bombarded with an increasing quantity of information and have limited bandwidth. Here, exploration of ML methods can help meteorologists to extract maximum value from the firehose of GOES-R observations.

The objective of this research is to ingest GOES-R Series observations from the Advanced Baseline Imager (ABI; *Schmit et al. 2017*) and GOES Lightning Mapper (GLM; *Goodman et al. 2013*) in precipitating scenes for the purpose of improving short-term convective-scale forecasts of high impact weather hazards. The Rapid Refresh and High-Resolution Rapid Refresh (RAP/HRRR) has long used radar reflectivity to estimate latent heating to spin-up convection in the models. (*Benjamin et al. 2016, Dowell et al. 2022, James et al. 2022, Weygandt et al. 2022*). Using this pathway for GOES information would require producing 3D fields of radar reflectivity. We will treat this problem as vertically separable, first estimating the horizontal spatial distribution of REFC, and then estimating the vertical profile in

a second step. This chapter will tackle the REFC part of the problem, focus on convective-scale applications, and consider warm season convection over the eastern CONUS where radar coverage is best. We describe the development of a convolutional neural network (CNN) for that purpose, including architecture selection and a novel approach to design a loss function to deal with class imbalances of REFC values (i.e., strong echoes are relatively infrequent). Performance is evaluated using metrics including the mean-square-error (MSE), coefficient of determination (R^2), categorical metrics (probability of detection, false alarm rate, critical success index, and categorical bias) at various output threshold levels, and evaluation of the root-mean-square-difference (RMSD) binned over the range of true output values.

A potential disadvantage of ML is that it is statistically based, making it harder to interpret. So, besides producing a trained and evaluated model, part of the focus of this chapter is on developing tools for the interpretation and explanation of the strategies for how CNNs make predictions. This chapter is concerned specifically with tools for the GOES-radar translation problem, but a more general review for image-to-image translation problems is provided by *Ebert-Uphoff and Hilburn (2020)*. Using ML to transform satellite data inputs has the potential to introduce errors arising from uncertainties related to the connection between observed cloud top features and lightning with estimates of latent heating vertical profiles, so it is very important that we understand how the ML makes its predictions and characterizes the errors. *McGovern et al. (2019)* provide a thorough review of many approaches for understanding ML predictions. However, the focus of that paper is on methods for analyzing networks for image classification tasks, i.e., networks that take images as inputs and produce one scalar value as the output. In this study, the network is performing image-to-image translation, taking images as inputs and producing images as outputs. Thus, some techniques in *McGovern et al. (2019)* are not directly

applicable to image-to-image translation problems, and we explored several other methods. For interpreting image-to-image translation CNNs, Layer-wise Relevance Propagation (LRP, *Montavon et al. 2018*, *Lapuschkin et al. 2019*) was found to provide very useful information (Section 1.3.4). This chapter uses a novel analysis methodology combining LRP (Section 1.3.4) together with target architecture experiments (Section 1.3.2) and synthetic inputs (Section 1.3.5) to gain insights on strategies learned by the ML model that produce good skill.

The ML model developed in this chapter is designed for DA applications, but there are other related research efforts with aviation and nowcasting applications. *Veillette et al. (2018)* derived a CNN to predict radar vertically integrated liquid (VIL) from satellite data for aviation applications (Federal Aviation Administration Offshore Precipitation Capability FAA OPC). This is a similar problem to the one tackled by this chapter; however, they use a somewhat unconventional architecture where features are extracted from each input variable separately and then combined in fusion layers. An interesting question about that architecture is whether it allows the network to learn to use lightning data to focus its attention on specific IR features as in this work (Section 1.3.4). Another major difference with *Veillette et al. (2018)* is in how they handle the class imbalance issue. Herein, we use a weighted loss function approach, while *Veillette et al. (2018)* deliberately sample data to create a balanced training dataset with roughly equal portions of zero, low, and high-intensity VIL. *Ayzel et al. (2020)*, *Agrawal et al. (2019)*, and *Samsi et al. (2019)* trained CNNs with a similar U-Net architecture as in this chapter for the problem of nowcasting using radar data. *Su et al. (2020)* approached the nowcasting problem using a recurrent architecture, which should better capture temporally evolving features than a standard feedforward architecture. There are several commercial entities seeking to provide proxy global radar datasets. The Earth Networks company has developed PulseRad, which uses

their ground-based lightning detection network to create global proxy radar maps. The ClimaCell company has merged data from several low-Earth orbiting and geostationary satellites to create a Global Precipitation Layer product. The interpretation methods developed in this chapter could be applied to other CNN models for global radar or nowcasting to potentially improve upon the models and make them more explainable.

We begin with short descriptions of the "source" observations from the GOES-R ABI (Section 1.2.1) and GLM (Section 1.2.2), followed by our "target" observations from the Multi-Radar Multi-Sensor MRMS (Section 1.2.3). The approach for constructing the ML training and validation datasets is described in Section 1.2.4. The CNN architecture is described in Section 1.2.5, and the approach for constructing a weighted loss function is given in Section 1.2.6. The resulting CNN prototype has been dubbed "GOES Radar Estimation via Machine Learning to Inform NWP" (GREMLIN). In Section 1.3.1, we begin with an overall characterization of the performance of GREMLIN, finding remarkably good performance, even at higher REFC values. To explain how GREMLIN makes such predictions, in Section 1.3.2 we selectively disable specific abilities of this model, resulting in a progression of simpler models, and analyze their results. By examining the predictions from various models (withholding certain channels and/or withholding spatial information), many insights can be gleaned. To examine the use of spatial information, we discuss and visualize the Effective Receptive Field of GREMLIN (Section 1.3.3). To understand how the network is making its predictions, and how it uses radiance information and lightning together, we apply the attribution method Layer-wise Relevance Propagation (Section 1.3.4). Finally, we construct synthetic inputs representing different meteorological scenarios to probe the network's response and gain further insights into the use of spatial information by the network and to characterize its sensitivity (Section 1.3.5). Section 1.4

presents summary and conclusions.

1.2. Data and Methodology

1.2.1. Advanced Baseline Imager

This study makes use of level L1b radiances from the GOES-R ABI (*Schmit et al. 2017*) on GOES-16. We take advantage of the higher spatial resolution (2 km) and faster temporal refresh (5-min over CONUS) compared to the previous generation of GOES imagers. To produce a unified Day-Night algorithm, we focus on just infrared channels, and for maximum portability and compatibility to legacy observing systems, use the "heritage" channels:

- Channel 7, 3.9-micron, shortwave infrared window
- Channel 9, 6.9-micron, mid-level water vapor (~442 mb)
- Channel 13, 10.3-micron, clean longwave infrared window

The conversion and calibration of observed radiances *Rad* to brightness temperatures T_B for GOES ABI follows *Schmit et al. (2012*),

$$T_B = \frac{c_2}{\ln\left(\frac{c1}{Rad}+1\right)} \tag{1a}$$

$$T_{B,C} = \frac{T_B - b_1}{b_2} \tag{1b}$$

where c_1 and c_2 are the wavenumber-dependent coefficients used to compute the monochromatic T_B, and b_1 and b_2 are spectral bandpass correction offset and scale for calculating the calibrated brightness temperature $T_{B,C}$. These coefficients are provided in the GOES L1b NetCDF data files. We note that during the daytime, use of the optical depth information from the red band (ABI Band 2; 0.64 μm) reflectance and the cloud particle size and phase information from ABI Band 6 (near-infrared; 2.2 μm) reflectance provide additional skill, however, use of these bands is beyond the scope of this chapter. Also this information is daytime-only, making it more difficult

to leverage in a unified 24-hr way.

Two angular quantities are especially relevant to the interpretation of ABI imagery: satellite viewing zenith angle and solar zenith angle. This study makes use of GOES-16 data from 2019, positioned in its operational East position (75.2ºW). In this slot, the satellite viewing zenith angle increases from 35º in northern Florida to 60º in North Dakota. Since we are focusing over just CONUS, we can ignore viewing zenith angle dependence because the limb cooling effect (*Elmer et al. 2016*) is small in the atmospheric window bands we are considering. We will also consider an example of storms over Colorado in 2017 when GOES-16 was in its initial check-out position (89.5ºW), which had satellite viewing zenith angles around 45º, compared to 50º in the operational East position. ABI Band 7 (3.9 μm) has a daytime solar reflective component and we tested adding solar zenith angle as an input, but found it only made small improvements, so we left it out of the Version 1 GREMLIN model. In Section 1.3.1 we consider the skill of the model as a function of the solar zenith angle, which was calculated following *NOAA NESDIS (1998)*.

In a traditional pixel-based retrieval, correcting the effect of parallax (*Vicente et al. 2002* and Appendix A in *Miller et al. 2018)* is essential for matching up satellite data with radar data on these scales. The main uncertainty with parallax correction is estimating the height of the cloud. One can assume a fixed height, such as 10 km, to substantially reduce the error, at least for the deep clouds that are most relevant; or one can use a cloud top height product, but this can introduce blank spots in the parallax corrected imagery when low and high clouds are next to each other. To remove parallax offsets to first order, we assumed a height of 10 km. Besides residual parallax errors, there are other reasons for spatial displacements, namely vertical wind shear, and the CNN seems to learn to apply additional spatial displacements on its own based on

what it sees in the training data.

1.2.2. Geostationary Lightning Mapper

The other major advancement provided by the GOES-R Series is real-time lightning observation from the GLM (*Goodman et al. 2012, Goodman et al. 2013*). Lightning is incredibly useful in constructing synthetic radar fields because of its association with the locations of strong updrafts within an embedded convective complex. The physical basis for this association is the strong spatial relationship between lightning flash rates, updraft vertical velocity (W), and latent heat release. If the terminal velocity of a raindrop goes as the square-root of the diameter, then it can be shown that mass (and latent heating) goes as W^6 and (linear) radar reflectivity factor goes as W^{12}. Meanwhile, using simple electrostatic arguments (*Price and Rind 1992, Boccippio 2002*), one can derive that lightning flash rate goes as W^5 for continental thunderstorms.

Much of the research on using GLM for severe weather has focused on the temporal variability, in particular lightning jumps (*Schultz et al. 2009, Schultz et al. 2015*). However, temporal variability of optically sensed lightning can provide misleading signals. This seems to be due to time varying detection efficiency effects related to the production of cloud ice (*Rutledge et al. 2020*), and possibly to the unsteady nature of updrafts. Instead, spatial variability contains more reliable information content, supplementing missing information at very high optical depths, and is especially useful at night. While there is spatial variability in GLM detection efficiency (*Marchand et al. 2019*), our CNN is more sensitive to the presence of lightning rather than the magnitude of lightning activity, which makes it less sensitive to GLM detection efficiency issues.

GLM maps total lightning with a spatial resolution of 8 km at nadir to 14 km at the limb. The basic unit of data, called an “event”, is a gridded quantity, integrating all lightning pulses

within the grid box over a 2 ms time window. The Lightning Cluster and Filter Algorithm (LCFA) combines adjacent lightning pixels into "groups", which are then clustered into "flashes" using a 330 ms temporal window and a 16.5 km spatial window. Thus, groups and flashes are represented as point observations consisting of a latitude, longitude, time, and area. The LCFA also performs filtering to reduce false alarms. The locations provided in the Level-2 GLM data files have a parallax correction applied. Examination of a few sample storms found the best results (in terms of correlation with REFC) occur when using GLM groups, because they provide more "filled in" maps than using flashes. For this work we create group-extent density maps using the group area, assuming it is circular, and accumulating data over 15-minute intervals. We tested 5-minute accumulation periods but found this finer temporal granularity produced stratiform areas that flicker on and off from frame to frame. The lighting data units are given as: groups 5-min^{-1} km^{-2}.

1.2.3. Multi-Radar Multi-Sensor Dataset

The target dataset to which we are training is the quality-controlled composite reflectivity from the Multi-Radar Multi-Sensor (MRMS) product (*Smith et al. 2016*). The vertical coverage of MRMS as a function of location is given in Figure 1.1, which was created using the 3D reflectivity MRMS fields. Our region of interest for this study is the Continental United States (CONUS), east of the Rocky Mountains, over which radar beam blockage issues are minimal. As the radar beam propagates away from the transmitter it is progressively higher above Earth's surface due to both the curvature of the earth and the non-zero elevation angle of the beam itself (minimum of 0.5° for the operational Next-Generation Radar; NEXRAD). A comparison of REFC for Hurricane Dorian off the Florida coast with GOES observations indicated that when the vertical coverage falls below 70%, implying that only echoes above 3 km can be measured,

the estimate of REFC becomes questionable. When only 50% of the vertical levels are present, this implies that only echoes above 6 km can be measured, and it appears that REFC provides very little reliable information. Over the Great Plains, where dew point depressions are large and cloud bases are higher than in the tropical environments of hurricanes, the reliability of REFC might fall-off with distance more slowly. To use the best quality radar data, we are restricting our domain of interest to east of 105ºW, for which nearly all locations have 70% coverage, and most areas (by virtue of their population) have 90% coverage, or a minimum height of 1.25 km.

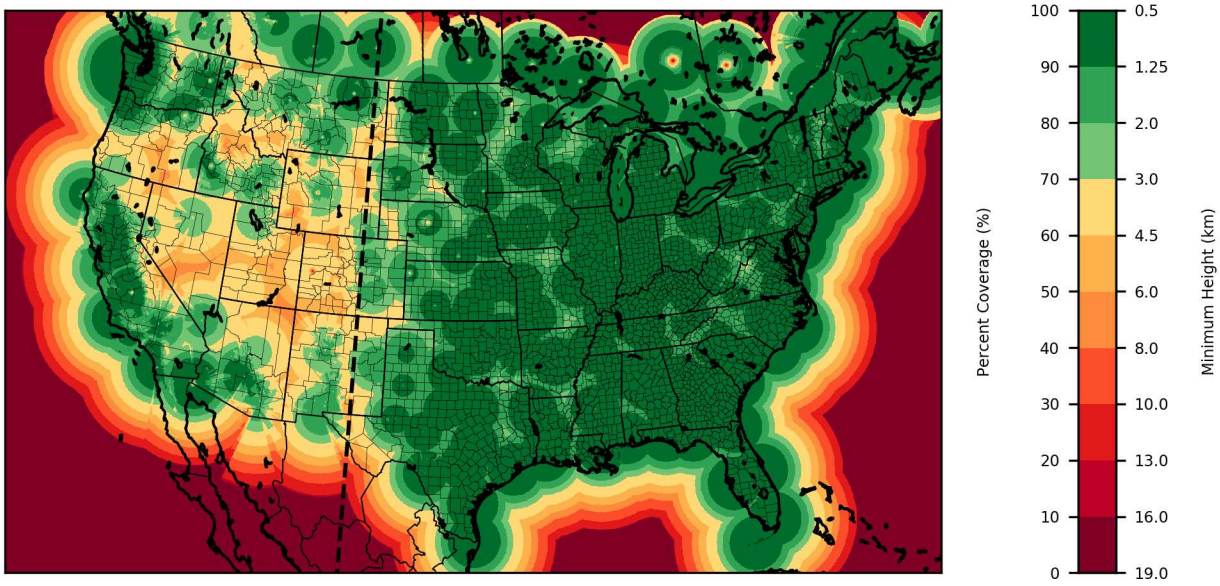

Figure 1.1. MRMS radar coverage in terms of the percent of MRMS levels available at each location and the minimum height at that location, 105ºW is indicated by the dashed black line.

1.2.4. Dataset Construction

The first step in constructing a dataset for training ML is to resample all the inputs and outputs to a common grid. Since the goal of this work is to use the results for data assimilation, we have chosen the 3 km High-Resolution Rapid Refresh (HRRR) mass grid as the target grid. The projection and grid parameters are provided in Table 1.1, the formulae used for constructing the Lambert Conformal Conic and Cylindrical grids are given by *Synder (1987)*, and the

formulae for the geostationary projection are provided by *Harris Corporation (2016*). The MRMS grid is nominally 0.01º or roughly 1 km, and the GOES grid for the infrared bands used in this study is 2 km, so resampling to 3 km has minimal distortion. GOES and MRMS pixels were averaged into their corresponding HRRR grid cell. We note that due to averaging, after resampling MRMS to 3 km REFC values above 60 dBZ are very rare. The second step in preparing the data for training a CNN is to scale the inputs and outputs to the range 0-1. The scaling parameters for each variable are given in Table 1.2, which were based on histograms of the variables. We found the training results were not very sensitive to the exact values of the scaling parameters, however the channel importance coming from LRP (Section 1.3.4) was sensitive.

Table 1.1. Projection and grid parameters for each dataset.

GOES		**MRMS**		**HRRR**	
Parameter	**Value**	**Parameter**	**Value**	**Parameter**	**Value**
Projection	Geostationary	Projection	Cylindrical	Projection	Lambert Conformal Conic
Altitude	35786023.0 m	Lower left longitude	-130ºE	Reference longitude	262.5ºE
Equatorial radius	6378137.0 m	Lower left latitude	20ºN	Reference latitude	38.5ºN
Polar radius	6356752.31414 m	Longitude scale	0.01	Standard parallel	38.5ºN
Center longitude	-75.0ºE	Longitude dimension	7000	X scale	3.0 km
X scale	5.6e-05	Latitude scale	0.01	X dimension	1799
X offset	-0.101332	Latitude dimension	3500	Y scale	3.0 km
X dimension	2500			Y dimension	1059
Y scale	-5.6e-05			Earth radius	6370 km
Y offset	0.128212				
Y dimension	1500				

Table 1.2. Scaling parameters for each variable. Each scaling is linear, and inverted scaling maps maximum values to 0 and minimum values to 1.

Channel	Minimum	Maximum	Inverted
C07 (3.9 μm)	200 K	300 K	True
C09 (6.9 μm)	200 K	250 K	True
C13 (10.3 μm)	200 K	300 K	True
GLM	0.1 groups 5-min^{-1} km^{-2}	50 groups 5-min^{-1} km^{-2}	False
MRMS	0 dBZ	60 dBZ	False

To reduce data volume and have the CNN focus on scenes of interest, Storm Prediction Center (SPC) filtered storm reports are used to automatically define regions and times of interest to maximize the number of storm reports (tornado, hail, wind). We selected samples from the 92-day period 4/17/2019 to 7/17/2019 during which there was abundant severe weather. The samples consisted of 256x256-pixel images on 3-km HRRR grid (768x768 km) and 6-hour periods with 15-minute refresh. A histogram of the number of storm reports per day has a mode between 20-50 reports per case. Each case represents a 6-hour period on each day, which may span 0Z. Figure 1.2a shows that this construction approach results in a geographic preference for the Upland South and Southern Great Plains. Figure 1.2b shows a temporal preference for mid to late afternoon into the early evening. We split the data using a chronological 80% - 20% split for training-validation. Based on this split, the July cases were used for validation, while April-June was used for training. We have a total of 1798 samples for training and 448 samples for validation. In this chapter we are restricting the focus to warm-season convection to benchmark ML performance for this particular phenomenon and identify the strategies learned by ML. Testing on wintertime precipitation, it is clear that extending the model to synoptic scale systems will require a deeper model with larger receptive field (Section 1.3.3). Future work training on a larger dataset will use the results of this chapter to ensure that the model can be extended without losing performance for warm-season convection.

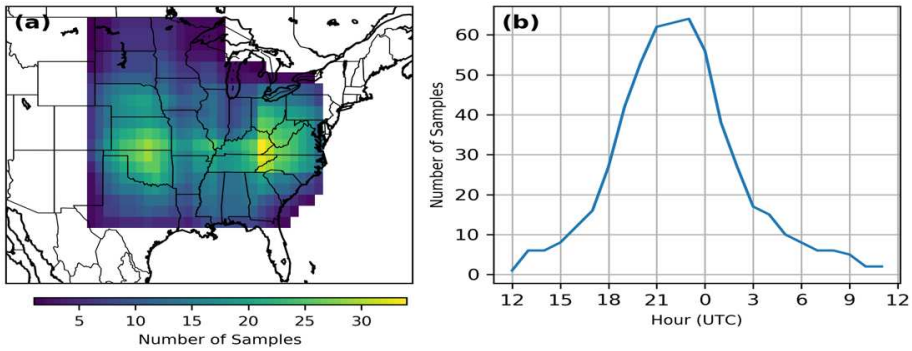

Figure 1.2. (a) Spatial distribution of samples. (b) Temporal distribution of samples.

1.2.5. Selection of Convolutional Neural Network Architecture

This ML problem takes images as inputs and returns images as outputs, making this an image-to-image translation problem. The U-Net architecture is ideally suited (*Ronneberger et al. 2015*) to this problem type, and Figure 1.3 shows the model architecture we used. The model is drawn with optional skip connections, which concatenate information from the encoder side to reduce the loss of high-resolution spatial information. However, for the results we will present we turned those connections off because they only provided small improvements and complicate the visualization (Section 1.3.4). For this application, the GOES data provides mostly cloud-top information, while the radar provides information from deeper inside the cloud, thus the high-resolution spatial information that skip connections provide is not necessarily helpful.

The CNN depicted in Figure 1.3 has three encoding and three decoding blocks. The encoding portion maps the inputs (images) to a feature space, and the decoder maps the representation in feature space back into images. Each of the three encoding blocks consists of a convolution layer followed by a pooling layer. A pooling layer reduces resolution and allows the subsequent layers to detect patterns of larger spatial extent. Each decoder block consists of a convolution layer followed by an upsampling layer. Upsampling layers can be thought of as the (imperfect) inverse of a pooling layer, namely increasing resolution and using interpolation to

generate an approximation. The convolutional filters are 3x3 kernels that the network learns during training. While U-Nets often double the number of filters per convolution layer going down the encoding branch, and likewise halve the filters going up the decoding branch, we found this produced very small improvements. Instead, we used a constant number of filters, namely 32 filters/convolution layer. Using more than 32 filters/layer was unnecessary and leaves many filters inactivated. Using fewer filters/layer, such as 16, gave similar overall statistics as 32, but the outputs were noticeably blurrier. The final layer of the network is a convolutional layer that does a pixel-wise (1x1 filter) linear combination of the 32 filters into one output field. We note that the combination we use of an upsampling layer with nearest-neighbor interpolation followed by a 1x1 convolution produces identical pixel values in 2x2 blocks in the output field. As a small future improvement, we will include additional 3x3 convolutional layers to obtain an interpolated result within the 2x2 blocks.

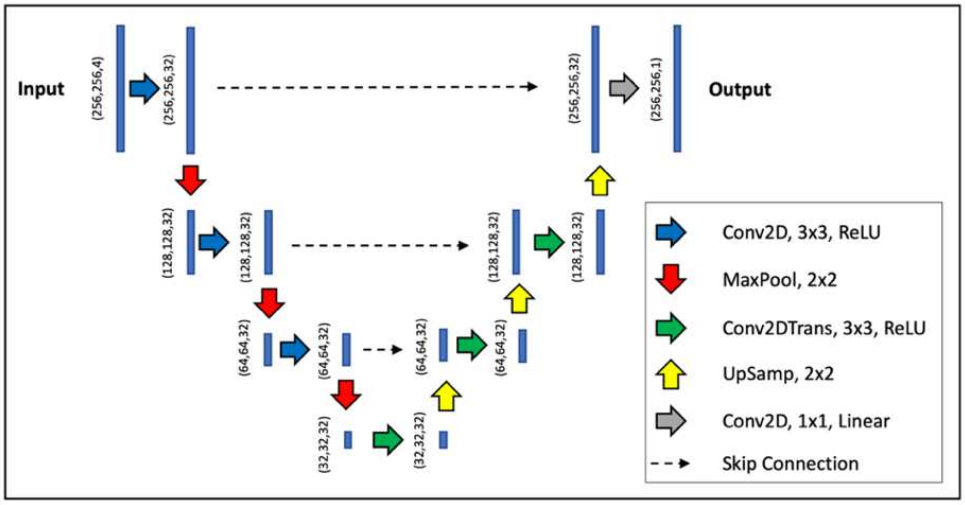

Figure 1.3. U-Net architecture for a model with 47,457 trainable parameters. The images are 256x256 pixels with four input channels (ABI C07, C09, C13, and GLM group extent density) and one output channel (MRMS composite reflectivity). The convolutional layers (blue and green arrows) each have 32 filters of size 3x3 and use a rectified linear unit activation function. The final convolutional layer (gray arrow) combines results from all filters into one output channel using one 1x1 filter and linear activation. The encoding branch (left side) uses 2x2 maximum pooling to reduce the image dimensions, while the decoding branch (right side) uses 2x2 upsampling to increase the image dimensions. The skip connections (dashed black arrows), which concatenate channels across the network, are turned off.

As noted above, there are three encoder and three corresponding decoder layers. Based on an analysis of training and validation losses, we found that going deeper resulted in overfitting. Also note our choice of using only one convolution layer per encoder/decoder block, while U-Nets often use two convolution layers per block. Using two convolution layers per block doubles the number of trainable parameters, also making the chance of overfitting more likely. We are concerned with warm season convection, a phenomenon that is inherently small scale (e.g., meso-γ to the smaller end of meso-α), and a network of this depth and architecture performs well. However, for larger spatial phenomena, such as hurricanes and synoptic-scale frontal precipitation, a deeper network will be required. In such cases, more samples would be needed for training. When additional real samples are unavailable, data augmentation is the next best approach. As a side note, we found we could obtain similar results as those shown in this chapter with a training dataset of 1/10 original sample by doing 10x augmentation, done by adding random noise to the real samples. However, the results shown herein used no data augmentation.

The model was trained on a single NVIDIA Tesla P100 GPU for 100 epochs, which took 15 minutes of wall-clock time. Using a batch size of 18, the model had a memory footprint of 0.5 GiB, and the data required 8 GiB of memory. The final model stored in HDF5 is only 625 KB. The training history is shown in Figure 1.4. Training beyond 100 epochs we observed the validation loss flattened while the training loss continued decreasing, indicating that further training would produce overfitting. The loss function is described in the next subsection. The final Version 1 GREMLIN model has validation statistics against MRMS observations: RMSD = 5.53 dBZ and $R^2 = 0.740$.

1.2.6. Design of Loss Function to Address Class Imbalance

In ML, the loss (or cost) function quantifies the difference between the model predicted values and the actual true value. The process of training a model involves changing the NN's weights to minimize the loss function. An important consideration in training the NN is the choice of loss function since radar reflectivity fields suffer from a class imbalance issue (i.e., non-uniform population of classes in the truth data) with an exponentially decreasing distribution for high values. In this section we discuss a new way to design a loss function to balance good performance for the rare (but important) high values with good performance for small values. Accompanying TensorFlow code for this custom loss function is available in *Ebert-Uphoff et al.* (2021).

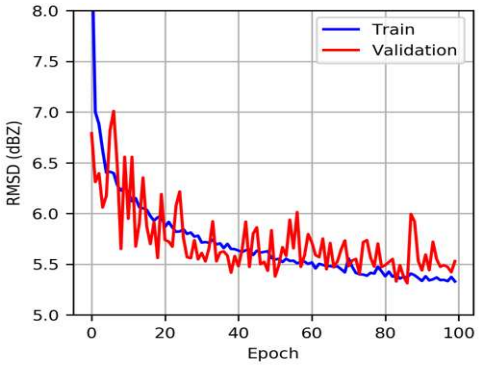

Figure 1.4. Training history for GREMLIN in terms of the root-mean-square difference with MRMS.

Training the NN using the standard unweighted pixel-wise mean-square-error MSE loss function results in sub-optimal performance at high REFC (Figure 1.5). High radar reflectivity values are relatively less common: if y represents the scaled radar reflectivity (scaling 0-60 dBZ linearly into the range 0-1), then the probability density function is closely approximated by $P(y) \propto e^{-5y}$ with an R^2=0.80. We use a performance diagram (Figure 1.5) to select loss function weights that produce the minimum categorical bias. Categorical statistics and contingency tables are discussed in *Wilks (2006)* and performance diagrams are discussed in *Roebber (2009)*. The binary categories are created by evaluating whether the true and predicted REFC are greater than

a threshold. While minimizing the categorical bias does not guarantee that the results will also have maximal critical success index, we found that in practice this was the case.

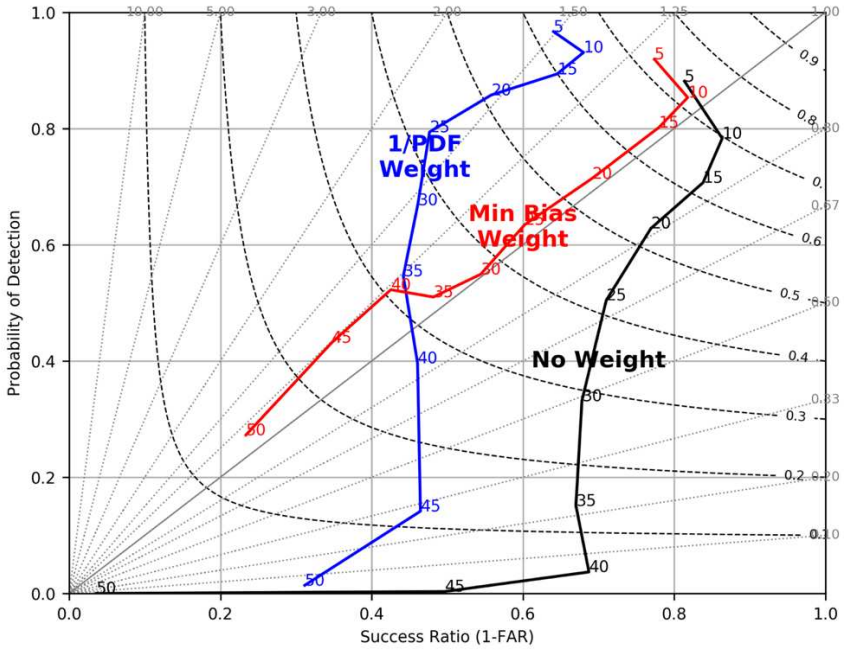

Figure 1.5. Performance diagram for REFC categories 5, 10, …, 50 dBZ. Dashed black contours are critical success index, and grey dotted lines are categorical bias. The solid black line is performance using unweighted MSE loss function, solid blue line uses 1/PDF weighted MSE loss function, and the solid red line uses weights that produce the minimum categorical bias (GREMLIN).

Our approach is related to using an Area Under the Receiver Operating Characteristic Curve as a loss function but avoids the problem of derivatives not existing for a discontinuous function. The approach also acts as a global constraint on the realism of the resulting fields by balancing overprediction and underprediction of reflectivity at all values. We define weights (*W*) for the MSE loss function (*L*) according to a generalized exponential:

$$L(y_{true}, y_{pred}) = \frac{1}{N}\sum_{j=1}^{N} W(y_{true})(y_{pred} - y_{true})^2 \quad (2a)$$

$$W(y_{true}) = e^{by_{true}^{c}} \quad (2b)$$

where y_{true} and y_{pred} are the true and predicted values of y (scaled 0-1) and N is the number of training samples. We then vary b and c in a grid search, training a NN model for each

combination, to find the optimal model producing the minimum categorical bias. Values of the categorical bias are calculated at each REFC threshold *i* from 5 to 50 dBZ in steps of 5 dBZ, and best matching model is found taking the parameter combination *k* with:

$$\min_{k}\left(\operatorname{mean}_{i}\left(\left|1-Bias_{i,k}\right|\right)\right) \tag{3}$$

To get reliable results, we also train several versions of the model (20 versions) that differ only in their random seeds and then select the model minimizing (3). During training we observed that errors for low REFC values settled down first, as evidenced by a categorical bias near 1, and that errors for high REFC values settled down last. We used only the training samples to perform model selection, to keep the validation samples independent. While the intuitive 1/PDF weights would give *b*=5 and *c*=1, we found the minimum categorical bias weights were *b*=5 and *c*=4 for the MSE (mean-square-error) loss and *b*=5 and *c*=3 for the MAE (mean-absolute-error) loss. The disparity suggests there might be a way to choose coefficients from first principals based on the PDF, but we note that the best results require a much heavier weighting of the high values than would be implied by direct usage of the inverse of the PDF.

1.3. Results and Discussion

1.3.1. Baseline Network Performance

The overall performance of our final neural network, GREMLIN, is shown as the red line in Fig. 1.5. To understand the abilities of GREMLIN to produce synthetic radar reflectivity, it is helpful to consider a specific example. Here, we considered a severe weather event that occurred along the Colorado Front Range on 8 May 2017. Figure 1.6 compares MRMS REFC (a,c,e) with GREMLIN REFC (b,d,f) at three times during the event (21Z, 23Z, 01Z), noting that the first large hail reports were at 20:50Z and lasted until 21:30Z. This case is notable because of its severe impact on the Denver Metropolitan Area; the storms produced up to baseball sized hail

(2.75 inches) and was the costliest weather catastrophe in Colorado – producing $1.4 billion in insured losses (*Svaldi, 2017*). In addition to its human impact, this case poses challenges for both infrared imagers and optically sensed lightning. It is an example of Great Plains thunderstorms with abundant cloud water concentrations (e.g, *Williams et al. 2005*) that produce large anvils that obscure the convective cores in infrared imagery, making them particularly challenging for estimating radar reflectivity from passive visible/infrared satellite observations. Here, lightning information from GLM can in principle assist in homing in on these convective cores. While these conditions also lead to very high lightning rates, *Rutledge et al. (2020)* show these conditions also produce storms for which the lighting flash height is relatively low, making for large optical paths between the lightning source and the upper cloud boundary along the GLM sensor line of sight (both in general and for this case). This regionally common "inverted" charge structure causes a relative minimum in lightning detection efficiency over the Great Plains (*Marchand et al. 2019, Fuchs et al. 2018*) compounding the challenges presented by this case.

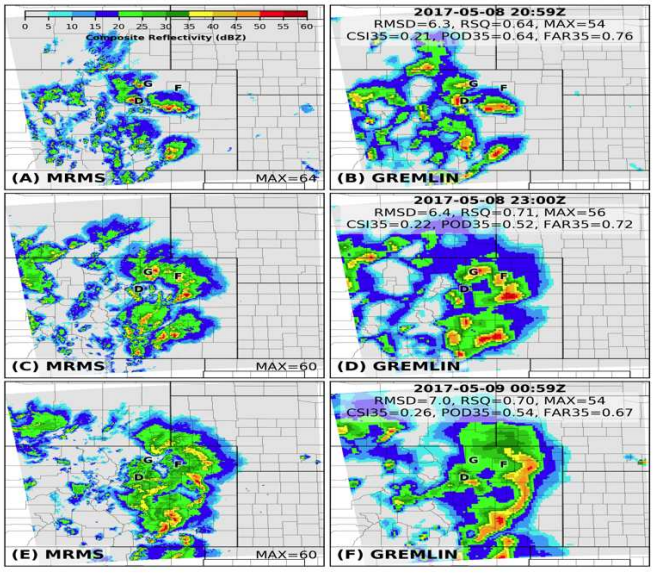

Figure 1.6. Colorado 2017-05-08 case: MRMS (a), (c), (e); GREMLIN prediction (b), (d), (f). Statistics are provided for root-mean-square-difference (RMSD), coefficient of determination (RSQ), maximum REFC value (MAX), critical success index at 35 dBZ (CSI35), probability of detection at 35 dBZ (POD35), and false alarm rate at 35 dBZ (FAR35). The cities of Denver, Greeley, and Fort Morgan are marked with D, G, and F.

Despite the challenges, Figure 1.6 shows that GREMLIN performs well for this case. In the early stages (Fig. 1.6a,b) GREMLIN captures the three distinct convective cores near Denver, Greeley, and Fort Morgan (marked D, G, and F). It correctly represented the location of the strongest echoes, although it also tended to overestimate their horizontal extent, and the fine-scale structure of the cores is not captured. Two hours later (Fig. 1.6c,d) as the storms began to transition to a convective line morphology, GREMLIN captured that transition well. GREMLIN properly located the strong echoes, although small areas that were distinct in MRMS tended to get merged in GREMLIN. Finally, after dark (Fig. 1.6e,f) and as the convection transitioned from distinct cells to lines, GREMLIN captured the basic shape and curvature of the lines, but tended to merge lines that were separate in MRMS.

Characterizing the spatio-temporal performance of the technique is complicated by the natural variation of convective morphology. In our training dataset, convection tends to be more widespread in the eastern U.S., while isolated convective cells are more common in the west. Since RMSD statistics are sensitive to the echo coverage fraction F, care must be taken to separate true regional biases from artificial biases that arise from natural regional variations in these properties. Figure 1.7a shows the RMSD versus the echo coverage fraction, F, defined here by the 20 dBZ radar reflectivity contour. More scattered precipitation with smaller F can be more accurately estimated with smaller RMSD. It also shows that eastern U.S. regions tend to have both larger F and larger RMSD. However, the easternmost locations do have errors greater than the average (black line given by RMSD = 2.2 $F^{0.36}$). Given that our training samples have a fairly uniform distribution from east-to-west (Fig. 1.2a), the fact that the predictions exhibit an "Oklahoma-centric" bias is notable and may be a consequence of using a loss function that is heavily weighted toward higher REFC values.

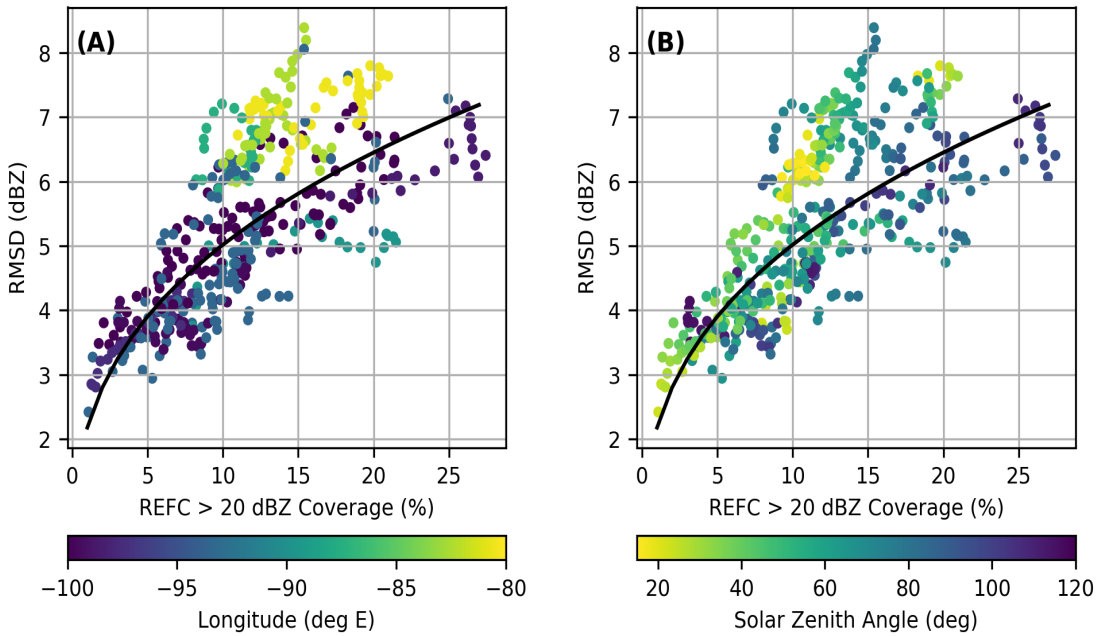

Figure 1.7. RMSD versus the percentage coverage of radar echoes > 20 dBZ where the color indicates the mean (a) longitude and (b) solar zenith angle for each sample.

The typical lifecycle is for convection to initiate with the heating of the daytime, and then grow upscale overnight. One might expect the large echo structures at night to validate better since the GREMLIN estimates tend to be more smoothed out than MRMS REFC. To look for biases in time, Fig. 1.7b gives the RMSD vs *F* as a function of the solar zenith angle, where sunset is 90°. It does show a population of samples that have both large *F* and small RMSD, however most nighttime samples are below the average line, even at smaller *F*. This good performance at night is notable given that our training samples emphasize late afternoon and early evening (Fig. 1.2b). It is possible this is a result of GLM having a 20% higher detection efficiency at night than during the day (*Marchand et al. 2019*). Not all daytime retrievals have lower skill, and the day/night distinction in skill is less clean than the east/west distinction. However, since daytime retrievals do have room for improvement, this argues that the solar reflective bands (visible and cloud particle phase/size bands) should be used. Overall, GREMLIN performs well. GREMLIN can accurately locate areas of strong echoes, which have

been difficult to capture with heritage methods (e.g., *Arkin and Meisner, 1987*).

1.3.2. Targeted Architecture Experiments

A key question raised by the results shown in Section 1.3.1 is: "*What is the network learning to produce such good skill?"* We use several different methods to answer this question, starting with targeted architecture experiments. Namely, we modify the GREMLIN architecture by removing specific capabilities. Analyzing the performance of the resulting restricted NNs tells us which capabilities of GREMLIN are most essential for its success and sheds light on how they are used.

We begin by removing the capability of GREMLIN to utilize information from radiance gradients and spatial context used by the network - done by replacing all 3x3 filters by 1x1 filters. Secondly, we trained models withholding sets of channels. Figure 1.8 provides results for a representative validation sample. For simplicity, we focus on the impact of gradients in Channel 13, which is the most important channel (Section 1.3.4), and of lightning information. The C13 T_Bs (Fig. 1.8a) exhibit very sharp spatial gradients from clear areas with $T_B > 275$ K to areas with radar echo with $T_B \sim 220$ K. Comparing with the spatial pattern of REFC (Fig. 1.8c) it can be seen that cold T_Bs are generally a good predictor that a particular pixel has REFC > 15 dBZ, but there is a low spatial correlation between the coldest $T_B < 215$ K and the higher REFC values > 35 dBZ. These areas of strong echoes correlate well with lightning (Fig. 1.8b), although the lightning is a bit smoother than REFC and there are spatial displacements. The latter may be due to a combination of residual parallax displacement errors and the effects of vertical wind shear.

Fig. 1.8d-i shows the progression of results for six NN models with increasing capabilities, from the most restricted model (Fig. 1.8d) to the full model, GREMLIN (Fig. 1.8i).

The 1x1 filter experiments are shown in the middle row (Fig. 1.8d,e,f), which represents the performance that could be expected from a traditional pixel-based retrieval. With C13-alone (Fig. 1.8d) the areas of REFC > 15 dBZ are reasonably well delineated, but it completely lacks any echoes > 35 dBZ. Combining GLM with C13 (Fig. 1.8e) shows huge improvements in the representation of echoes > 35 dBZ, although the spatial extent is a bit too large. Bringing in the other two channels (C07 and C09) in Fig. 1.8f does help reduce the errors a bit. So, without the use of spatial gradient information, lightning information is critical to obtaining any skill for higher REFC values.

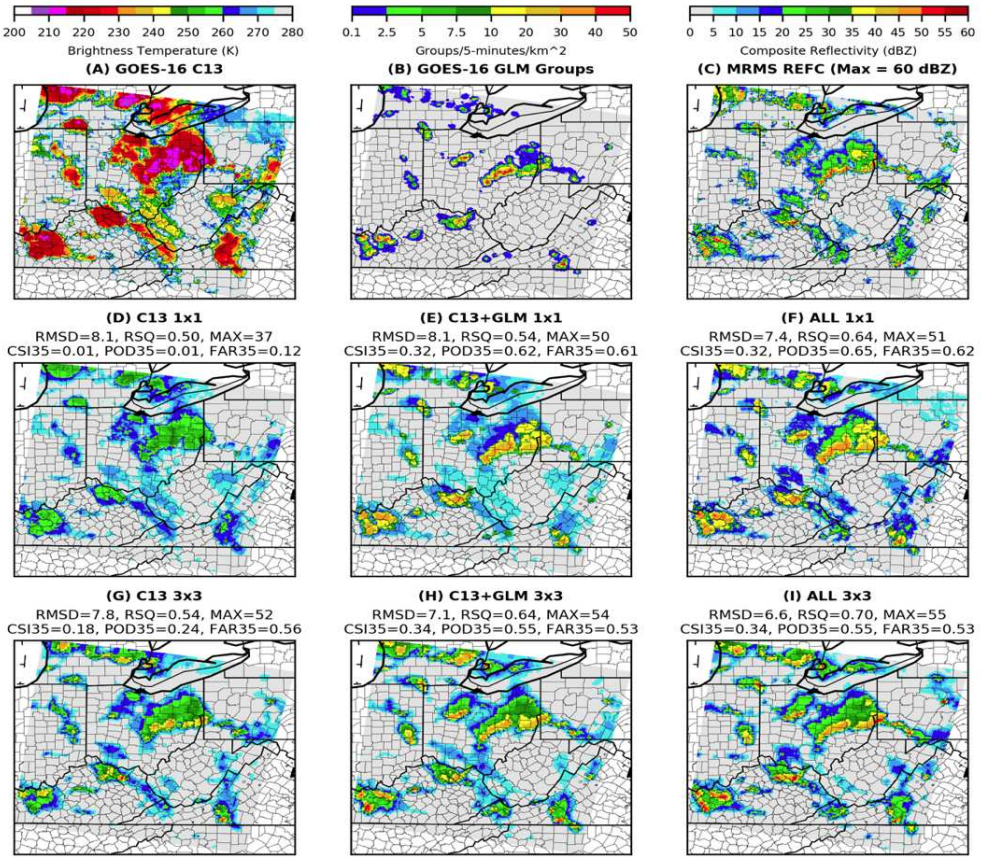

Figure 1.8. Validation sample 2019-07-02 23:30Z inputs: (a) GOES C13 and (b) GOES GLM; truth: (c) MRMS; and prediction for progression of six models with increasing capabilities, (d) 1x1 filters C13-only, (e) 1x1 filters C13+GLM, (f) 1x1 filters all channels, (g) 3x3 filters C13-only, (h) 3x3 filters C13+GLM, (i) 3x3 filters all channels (GREMLIN). Panels (d)-(i) provide the following statistics: root-mean-squared-difference (RMSD in dBZ), coefficient of determination (RSQ), maximum REFC (MAX in dBZ), 35-dBZ critical success index (CSI35), 35-dBZ probability of detection (POD35) and 35-dBZ false alarm rate (FAR35).

The bottom row (Fig. 1.8g,h,i) shows the results using 3x3 filters. Even with C13-alone, the use of gradient information and spatial context (Fig. 1.8g), produces marked improvements in skill, especially at the high REFC end. Compared with the 1x1 experiment (Fig. 1.8d) the probability of detection (POD) of 35 dBZ reflectivity jumps from 0 to 0.24, and the false alarm rate (FAR) of 0.56 is slightly better than using all channels with no spatial information (Fig. 1.8f). Adding lightning information (Fig. 1.8h) more than doubles the POD and reduces the FAR. Adding the other channels (Fig. 1.8i) helps as well, producing significant improvements in RMSD and R^2, also resulting in higher POD and lower FAR. We hypothesize that results of this quality (Fig. 1.8i) are sufficiently good to produce a positive impact on data assimilation.

The results for this example are consistent with those across all validation samples (Fig. 1.9 and Table 1.3). Without the benefit of spatial information and lightning (black and green lines in Fig. 1.9a), the RMSD at high REFC is as large as 25 dBZ. Note that removing spatial context but adding lightning (blue line Fig. 1.9a), makes the RMSD slightly worse for REFC in the range 20-35 dBZ, but produces large improvements above 35 dBZ, bringing the RMSD down to 15 dBZ. Adding spatial context yields additional large improvements (Fig. 1.9d). Combining spatial information and lightning produces the best results, with RMSD of 12 dBZ at the highest REFC. Without spatial information, lightning shows obvious value in increasing the POD (Fig. 1.9b) and reducing the FAR (Fig. 1.9c). In the absence of lightning information, adding the water vapor channel (green line Fig. 1.9b,c) does provide some improvements in POD and FAR, but not as much as lightning. Based on examining predictions, it appears the network correlates smaller differences between C13 and C09 with higher REFC. However, those areas of small C13-C09 difference tend to be more spatially extensive than REFC, with the result being that POD is improved, but FAR is slightly worse. This finding demonstrates the unique benefits of

lightning information to pinpoint the areas of strong updrafts and high REFC. When spatial information is used, the value of lightning is relatively less, but it still makes significant improvements in POD (Fig. 1.9e) and FAR (Fig. 1.9f). Further insights into how the network is using lightning and spatial information together is provided by use of attribution methods (Section 1.3.4).

Table 1.3. Categorical performance statistics for GREMLIN: probability of detection (POD), false alarm rate (FAR), critical success index (CSI), categorical bias (BIAS) for various composite reflectivity (REFC) thresholds.

REFC (dBZ)	POD	FAR	CSI	BIAS
5	0.92	0.23	0.72	1.19
10	0.85	0.18	0.72	1.04
15	0.80	0.22	0.65	1.03
20	0.71	0.31	0.54	1.03
25	0.63	0.40	0.45	1.05
30	0.55	0.46	0.38	1.01
35	0.51	0.57	0.33	1.06
40	0.52	0.57	0.31	1.23
45	0.43	0.65	0.24	1.24
50	0.37	0.77	0.14	1.17

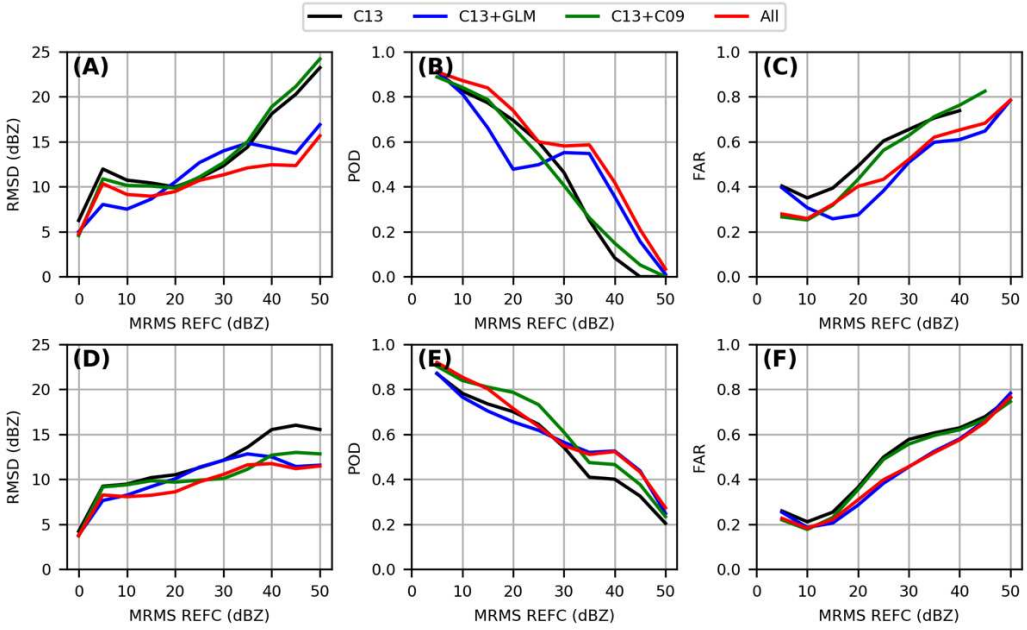

Figure 1.9. Statistics for 1x1 filters: (a) RMSD, (b) POD, and (c) FAR vs REFC for various experiments (line colors). Statistics for 3x3 filters: (d) RMSD, (e) POD, and (f) FAR vs REFC for various experiments (line colors).

GREMLIN predictions can be seen to have overly broad convective cores (e.g., Fig. 1.8), or, when using a continuous color-scale, predictions look blurrier than real radar images. This is an intrinsic aspect of CNNs related to the perception-distortion tradeoff (*Blau and Michaeli 2018*) for image-generating methods, which is a trade-off between producing images that look sharp but are less accurate (better perception) versus images that look blurry but are more accurate (less distortion). CNNs specialize in maximizing accuracy, e.g., minimizing the mean squared error, but results are blurry. CNN outputs are somewhat analogous to an ensemble mean field – it might be the best answer in a statistical sense, but it may not look physically realistic. In contrast, a different type of neural network, Generative Adversarial Networks (GANs), can produce results that are less statistically accurate, but could more closely resemble actual radar fields. GAN outputs are somewhat analogous to producing a *single* ensemble member. *Stengel et al. (2020)* apply GANs applied to a wind and solar data super-resolution application and discuss the trade-off in detail. In summary, increased uncertainty results in increased blurriness in CNN-generated images, while resulting in a larger potential spread between different GAN-generated images. Our interpretation of the broad convective cores generated by our CNN are thus that they are a result of uncertainty yielding blurry outputs, as outlined above. Specifically, our hypothesis is that the overly broad cores provide an indication of positional uncertainty translating cloud-top features into features deep inside the cloud. In our future work we plan to try out GANs and compare results with CNNs in terms of accuracy versus blurriness.

1.3.3. Examining the Effective Receptive Field

GREMLIN is a *purely* convolutional neural network, i.e., it does not have any fully connected (aka dense) layers. This means that any individual output neuron, i.e., any pixel of the estimated MRMS image, is connected to only a small group of input neurons corresponding to a

small spatial neighborhood of the output pixel in the input channels. This small area is known as a CNN's *Receptive Field* (*Luo et al. 2016*). For our application the receptive field tells us the maximal spatial context size and thus the maximal size of a meteorological feature that can be recognized and utilized by GREMLIN to determine the value of a single pixel of the estimated MRMS image.

One can calculate the maximal extent of the receptive field, aka the Theoretical Receptive Field (TRF), from the CNN architecture using a closed-form expression that depends on the filter sizes and strides for each layer (*Araujo et al. 2019*). Results for GREMLIN's TRF are provided in *Ebert-Uphoff and Hilburn (2020)*. However, pixels at the center of the receptive field have the largest impact, with impact decreasing rapidly for pixels further away in a roughly Gaussian distribution (*Luo et al 2016*). Here we take the approach to sample the actual distribution of the receptive field, the *Effective Receptive Field* (ERF, *Luo et al 2016*), to understand which size neighborhood truly has a significant impact. The ERF, which depends on the network's weights, changes during training. Thus, it cannot be calculated from architecture alone. Here, we develop an ERF approximation based on the SmoothGrad algorithm (*Smilkov et al. 2017*). The approximation is described in detail in Appendix A. The similarities between the receptive field and the radius of influence in DA applications suggests that the receptive field size – either the TRF or ERF – could potentially be used as an indication for choosing the radius of influence.

Figure 1.10 shows our approximation of ERF for GREMLIN for different lengths of training, ranging from an untrained model with random weights (Fig. 1.10a) to the final model trained for 100 epochs (Fig. 1.10d). Each ERF image in Fig. 1.10 shows the cumulative results across all four channels. Note that the ERF consistently occupies a region of less than 53x53 pixels or 159 x 159 km (red squares in Fig. 1.10) with the region of highest impact much smaller

than that, especially in the trained models. The ERF of the untrained model is the most spread out (Fig. 1.10a). Early training (Fig. 1.10b,c) seems to make the model put more emphasis toward the center, potentially as a sort of first-order approximation. The final model retains some focus in the center, but also spreads out more–potentially moving beyond the first-order approximation and taking additional detail into account. While the results in Fig. 1.10 are only ERF approximations (details in Appendix A), and vary across considered samples, output pixels, and random seeds used to train the CNN, we conducted many more experiments and found the trends in Fig. 1.10 to be representative of the overall behavior of the ERF distributions. Please see the detailed comments in Appendix A on the interpretation of such ERF approximations.

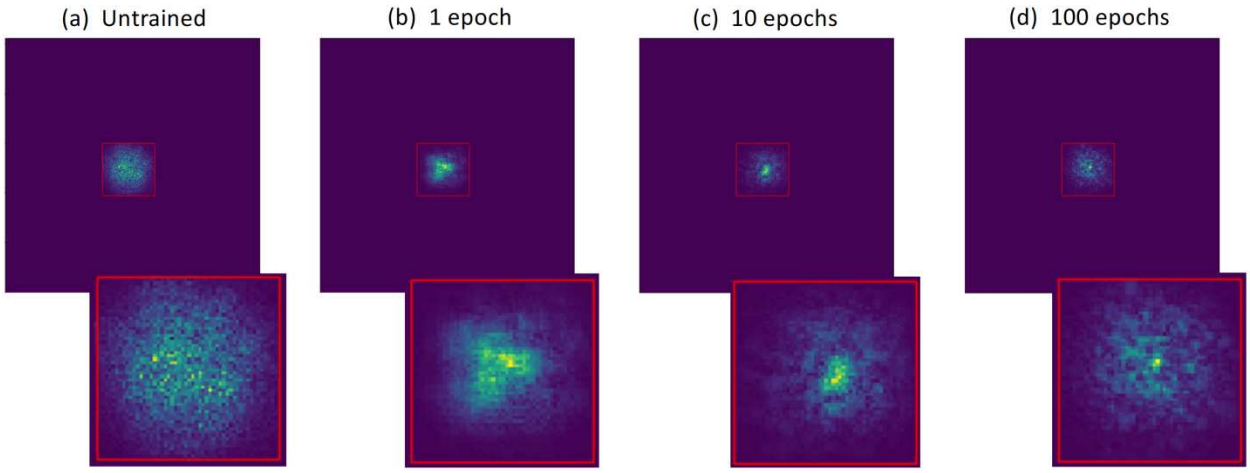

Figure 1.10. ERF approximation for four different models with identical architecture (architecture of GREMLIN), but different lengths of training, ranging from no training (a) to fully trained model, GREMLIN (d). For each image we show the ERF in the original 256x256 pixel (768x768 km) space of the input channels and a zoom-in of a 53x53 pixel (159x159 km) region (red box). Results are for Sample 68 and output pixel (125,125). (Note that the four models did not start out with the same random seed, thus cannot strictly be seen as a progression of training toward the final model, but rather as independently trained models with different training length.)

1.3.4. Applying Attribution Methods to Identify NN Strategies

To learn more about the underlying logic GREMLIN uses to derive its estimates, we use the method of layer-wise relevance propagation (LRP). Given an input sample and an output

pixel, LRP reveals where the neural network was primarily looking when deriving the output pixel's estimate. We find that LRP is better suited for this purpose than standard gradient-based methods because LRP takes a global view of this decision-making process, rather than just taking a local derivative as gradient-based methods do. Details of LRP are provided in Appendix B.

Figure 1.11 shows LRP results for GREMLIN for the same sample as in Fig. 1.10, but in this case focusing on a different output pixel, chosen for its proximity to strong lightning activity. All panels in Fig. 1.11 are zoomed into a neighborhood of the chosen output pixel. The first row shows the input channels and corresponding desired output (i.e., the MRMS observations). Because we suspected that the neural network was heavily reliant upon the gradient of the input channels, we show an approximation of the input channel gradient magnitudes in the second row. These gradient magnitudes were calculated by applying a Sobel operator (*Gonzalez and Woods, 2002*) to the input channels. The gradient estimates are not fed into the neural network; they are provided here simply to highlight the locations of the strongest gradients. The third row of Fig. 1.11 shows the first set of results, namely the LRP maps of where in the input channels the neural network pays attention to estimate the value of the chosen output pixel for this sample, along with the estimated MRMS results.

The LRP result for the GLM channel shows that the NN focused only on regions where lightning was present in that channel. The LRP results for the other channels show that even in those channels the NN's attention was drawn to focus on regions where lightning was present. We then performed a new experiment by modifying the input sample to have all lightning removed, i.e., the GLM channel was set to all zero values. For this case LRP showed us that the network's focus shifted entirely to the first three input channels, i.e., the ABI channels, as expected. More importantly, the focus shifted to two types of locations, namely areas where the

ABI input channels either have i) a *large gradient* or ii) *high brightness* (cold temperatures), as can be seen by comparing the three left-most panels of the first, second and fourth row. In fact, near the center of the fourth-row panels, the LRP patterns of the three ABI channels represent the union of the strongest gradient lines in the second row and the locations of strongest brightness in the first row. LRP vanishes further away from the center location, as expected given the nature of the ERF properties.

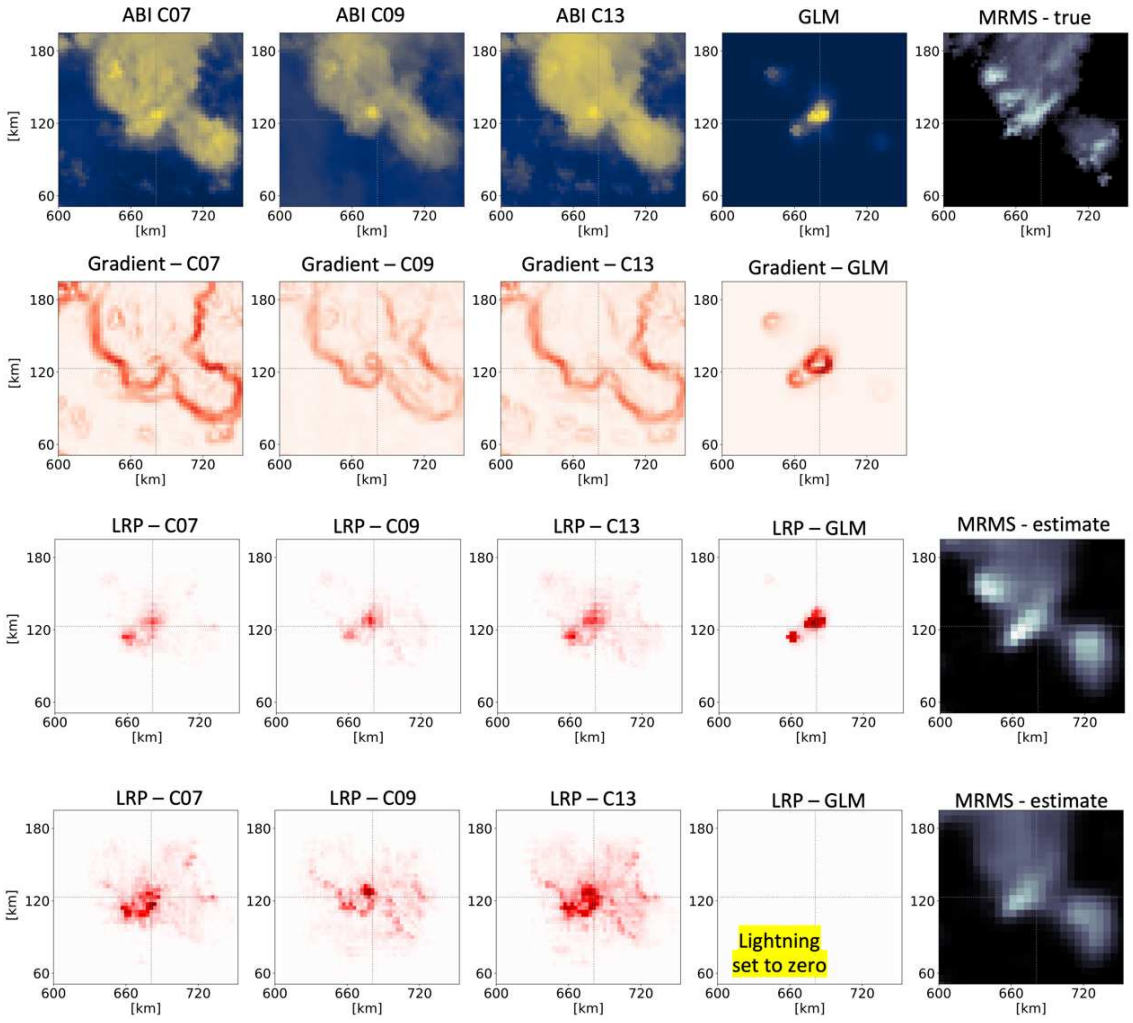

Figure 1.11. LRP results for GREMLIN for Sample 68 and output pixel (227,41). Top row shows the four input channels (left-to-right: ABI C07, ABI C09, ABI C13, GLM Groups) and the corresponding MRMS image (true values). Second row shows the gradient of the input channels calculated by applying a Sobel operator. Third row shows LRP results for the original four input channels and the chosen output pixel, and the MRMS estimate. The fourth row shows the equivalent of the third row, but after all values of the GLM channel were set to zero. Note that all images are zoomed in to a region centered at the pixel of interest.

These results indicate the following strategy used by GREMLIN: *whenever lightning is present near the output pixel, the NN primarily focuses on the values of input pixels where lightning is present, not only in the GLM channel, but in all four input channels.* It seems that the network has learned that locations containing lightning are good indicators of MRMS behavior, even in the other input channels. *In the absence of any lightning, the NN focuses on locations i) where the gradient is strong (primarily cloud boundaries), or ii) locations of very cold cloud tops.* It seems to have learned that those locations have the highest predictive power for estimating the output. Additional experiments confirmed these three strategies (lightning, cloud boundaries, cold cloud tops) of the final neural network for a wide selection of samples and output pixels.

1.3.5. Synthetic Inputs to Quantify Sensitivity to Radiance Gradients

The use of architecture experiments (Section 1.3.2) and attribution methods (Section 1.3.4) have demonstrated the importance of radiance gradients for retrieving high REFC values. In this section, we construct synthetic inputs and probe the network's response to quantify that sensitivity. For this purpose, we enlist a sum of Generalized Elliptical Gaussians (GEG) model. This model assumes an outer Gaussian (G_o) that represents the thunderstorm anvil, and an inner Gaussian (G_i) that represents the overshooting top. The synthetic brightness temperature (T) is a function of (x,y) with the following parameters: location x_0 and y_0, amplitude A, size S, aspect α, orientation θ, and sharpness (exponent) p for the outer and inner Gaussians, denoted with subscripts o and i, respectively:

$$\hat{x}_{o,i} = \left(x - x_{0,o,i}\right)\cos\theta_{o,i} - \left(y - y_{0,o,i}\right)\sin\theta_{o,i} \tag{4a}$$

$$\hat{y}_{o,i} = \left(x - x_{0,o,i}\right)\sin\theta_{o,i} + \left(y - y_{0,o,i}\right)\cos\theta_{o,i} \tag{4b}$$

$$T_{o,i} = exp\left(-1\left(\frac{\hat{x}_{o,i}^2}{2S_{o,i}^2} + \frac{\hat{y}_{o,i}^2}{2\left(S_{o,i}\alpha_{o,i}\right)^2}\right)^{p_{o,i}}\right) \tag{4c}$$

$$T = A_o T_o + A_i T_i \tag{4d}$$

Evaluating thousands of different parameter settings, the spatial patterns that most strongly activates the network, based on the maximum REFC, all resemble Fig. 1.12a. What the strongly activating patterns have in common, and what is different from the weakly activating patterns, are very large p_o and large p_i, meaning that the anvil and overshooting top have very sharp T_B gradients. We evaluated p_o and p_i ranging from 0.1 to 10. The other traits the strongly activating patterns have in common are that G_i is located near the edge of G_o and that $S_i << S_o$. We evaluated S_i ranging from 0 to S_o. The patterns producing a weak response tend to look unphysical from a meteorological perspective, indicating that the network has learned about realistic looking overshooting top signatures. This is a desirable property: rather than responding strongly to unphysical outlier inputs, it only responds strongly to patterns that look meteorological, although that does not rule out the possibility that the network could be fooled by a cleverly constructed counterexample.

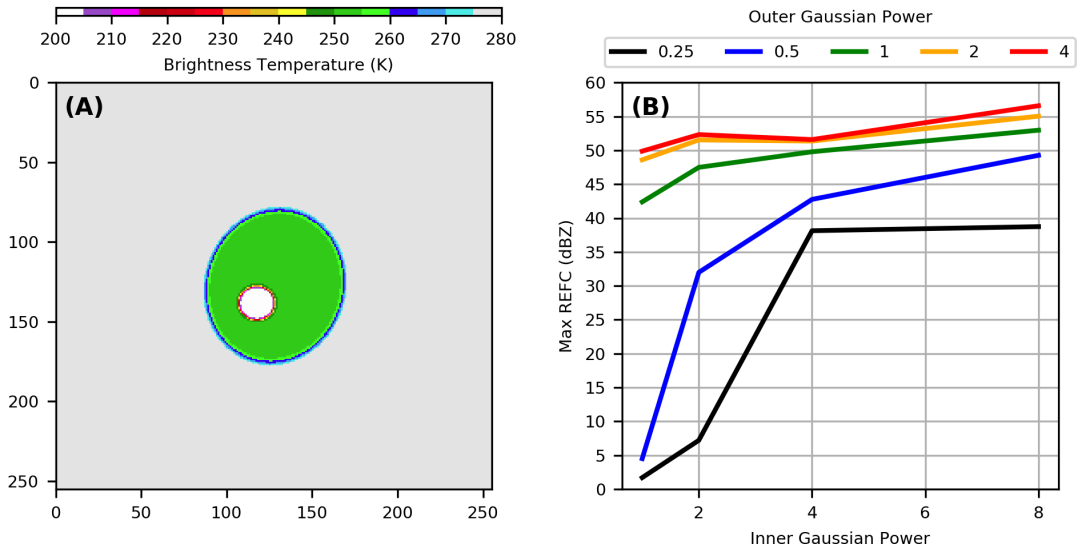

Figure 1.12. (a) Synthetic C13 T_B that produces the maximum REFC response for GREMLIN. This corresponds to parameters: $x_o = 128$, $y_0 = 128$, $A_o = 0.5$, $S_o = 30$, $\alpha_o = 1.2$, $\theta_o = 170^o$, $p_o = 10$, $x_i = x_o + dx_i \cos(\phi_i)$, $y_i = y_o + dy_i \sin(\phi_i)$, $A_i = 0.5$, $S_i = \rho_i S_o$, $\alpha_i = 1$, $\theta_i = 0^o$, $p_i = 10$, $dx_i = 15$, $dy_i = 15$, $\phi_i = 135^o$, $\rho_i = 0.25$. (b) Maximum REFC as a function of inner Gaussian power (x-axis) and outer Gaussian power (line color).

We explored outer sizes from 1-128 pixels, outer and inner aspects from 0.1 to 10, and outer and inner orientations 0-360º. Out of all the parameters of the GEG model, the ones that are most influential in producing high REFC values are p_o and p_i, and Fig. 1.12b characterizes the maximum REFC as a function of those parameters. The emergence of 35 dBZ echoes requires p_o to be 1 or greater, or p_i to be 4 or greater. Thus, the CNN does not just respond to gradients, but calibrates its response based on the sharpness of the brightness temperature gradient. Related to these idealized synthetic input experiments, future work will consider using observation system simulation experiments (OSSEs) to quantify errors associated with transferring from satellite observations to latent heat profiles.

1.4. Summary and Conclusions

In this chapter, we report on the training and evaluation of a CNN that uses ABI infrared channels and GLM lightning data to estimate MRMS REFC over eastern CONUS during the warm season. Since REFC follows an exponentially decreasing distribution, to get good performance at high values, we used a weighted loss function. This chapter demonstrated that the network is learning physically meaningful strategies to predict radar reflectivity from satellite radiances and lightning. A variety of approaches were examined to investigate what the network learned and how it makes its predictions. Channel withholding experiments showed that geostationary lightning observations are uniquely valuable for their ability to pinpoint locations of strong updrafts. Experiments withholding spatial information demonstrated that radiance gradients carry more information about high REFC values than the radiance values themselves. Layer-wise relevance propagation established that the CNN uses the information from ABI and GLM in a synergistic manner, where it interprets ABI radiance gradients in the context of whether GLM indicates the presence of lightning. Synthetic input experiments confirmed that the

sharper the gradient, the stronger the CNN response, at least for patterns that have an appearance reminiscent of meteorological convection.

Having established that the horizontal spatial patterns of radar reflectivity can be accurately estimated using GOES data, the next step in this research is to produce full 3D profiles of radar reflectivity for use as an input to data assimilation systems. The vertical information is provided using the profile model of *Lee et al. (2022)*, and several weeks of retrospective forecasts have been run *(Back et al. 2021, 2022)*. The results showed that replacing MRMS with GREMLIN in model initialization produces similar results over the whole of CONUS but produces better forecasts over the western U.S. where radar coverage is poor. Note that the current non-variational technique for initializing RAP/HRRR with radar reflectivity (*Dowell et al.* 2022, *Weygandt et al.* 2022) does not require characterization of uncertainty, however uncertainty information is required for variational approaches (*Barnes et al. 2021*).

Future work includes training and validation with a much larger dataset that includes samples from all times-of-year and using a three-way training-validation-testing split. We will also seek to provide a measure of confidence or uncertainty for use in data assimilation procedures. We also plan to try out GANs and compare results with CNNs in terms of accuracy versus blurriness. We emphasize this chapter is exploratory research and the current GREMLIN Version 1 model is unproven for estimating radar reflectivity for conditions outside of warm season convection over CONUS. Over CONUS the results are easy to validate using retrospective simulation experiments where the actual radar data are withheld and replaced by the GREMLIN estimates. However, the real value of the technique will come from its ability to fill in locations that lack radar coverage due to terrain blockage, which are mostly over the western U.S. and coastal/oceanic locations. Evaluating results in these locations is much more

difficult due to a lack of observations. However, MRMS sectors over the Caribbean (GOES-16), Hawaii (GOES-17), and Guam (Himawari-8) do provide observations, as do spaceborne radar reflectivity observations from the Global Precipitation Measurement (GPM) Dual-frequency Precipitation Radar (DPR). How well the model derived in this chapter will generalize to meteorological regimes outside of the training set is an open question. However, it is known that both lightning and storm characteristics are different over land versus ocean *(Nag and Cummins, 2017; Bang and Zipser, 2015)*. Thus, additional contextual information that is geographic or meteorological in nature may be needed, along with a deeper network to accurately depict features at the upper end of meso-α to synoptic scales.

CHAPTER 2: INTERPRETABLE GREMLIN[2]

2.1. Introduction

Convolutional neural networks (CNNs) are opening new opportunities in Earth remote sensing. For example, CNNs provide the ability to anticipate the presence of cloud structures beneath obscuring cirrus clouds, making greater usage of the visible and infrared radiances observed by geostationary satellites (*Meng et al. 2022*, *Haynes et al. 2022*, *Hilburn et al. 2021a*, *Veillette et al. 2018*). CNNs do this by using the information in image gradients and the spatial context in which a pixel is embedded, and in that way are mimicking how a human analyst visually interprets imagery. However, the black-box nature of CNNs hinders their trustworthiness: what exactly is the spatial context they use? This chapter seeks to provide insights by exposing an approximation of the CNN input feature space and using that to evaluate the CNN predictions.

The GREMLIN model (GOES Radar Estimation via Machine Learning to Inform NWP) uses a CNN to perform image-to-image translation from GOES radiances and lightning to Multi-Radar/Multi-Sensor (MRMS) composite reflectivity (REFC), showing good accuracy for warm season convection over CONUS (Contiguous United States) (*Hilburn et al. 2021a*). The explainable AI (XAI) technique of Layerwise Relevance Propagation (LRP, *Bach et al. 2015*, *Montavon et al. 2018*, *Lapuschkin et al. 2019*, *Mamalakis et al. 2022*) was used to gain insight

[2] Chapter 2 includes content from:
Hilburn, K., A., 2023: Understanding Spatial Context in Convolutional Neural Networks using Explainable Methods: Application to Interpretable GREMLIN. *Artif. Intell. Earth Syst.*, https://doi.org/10.1175/AIES-D-22-0093.1.
© American Meteorological Society. Used with permission. This preliminary version has been accepted for publication in *Artificial Intelligence for the Earth Systems* and may be fully cited. The final typeset copyedited article will replace the EOR when it is published.

on how GREMLIN makes accurate predictions, by showing in attention maps which channel and where in the image the network focuses its attention to predict radar reflectivity for a particular pixel. The use of LRP identified three physical strategies GREMLIN uses for predicting REFC (*Ebert-Uphoff and Hilburn 2020*, *Hilburn et al. 2021a*): (1) Lightning is a very strong predictor of strong radar echoes, (2) Cold brightness temperatures (TBs) are associated with stronger radar echoes, and (3) Stronger echoes are more likely on the cold side of strong TB gradients. These are all physically reasonable strategies, and the second strategy is the classical method for relating infrared observations to radar reflectivity and precipitation intensity in cold cloud-phase rain scenarios (e.g., *Arkin and Meisner 1987*).

While the insights provided by LRP were helpful for confirming these three strategies were encoded in GREMLIN, gaps in our knowledge remained. Had all the strategies used by GREMLIN been identified? The answer was not clear because LRP requires selection of individual pixels in particular samples for analysis. While the three strategies above were observed for the samples analyzed, LRP does not provide any guarantees for how GREMLIN would respond to new samples. It was obvious from LRP that GREMLIN was using spatial information, but it was not clear exactly how it was doing so in a quantitative manner. The inability to quantitatively define spatial context makes it hard to know how the model will generalize and thus how trustworthy its performance would be when confronted with novel scenarios. For example, to what degree does the preferred wind shear direction over CONUS, where GREMLIN was trained, influence the spatial structure of GREMLIN predictions? Thus, LRP is not satisfactory for full understanding, and this provides motivation for a new approach to dig deeper. Since GREMLIN is being considered for operational NOAA applications, it is important to gain a deep understanding of the model. Note that LPR is not the only XAI method

available, but an evaluation of various XAI methods is beyond the scope of this work. Newer methods such as Shapley additive explanations (*Lundberg and Lee 2017*) show promise to overcome limitations of LRP.

The objective of this chapter is to provide insight on the nature of spatial context utilized by a CNN through the development of an interpretable version of GREMLIN. This will lead to a model that is easier to understand, help elucidate how well the model will generalize to unseen regimes and identify the conditions in which predictions are uncertain due to lack of information content. The interpretation of a CNN is complicated by the presence of many layers (*Olah et al. 2017*), but if everything can be brought up into the first layer, it would make interpretation of the relationship between inputs and outputs much easier. Figure 2.1 illustrates our approach to this end, which is to pull the inner workings from inside the CNN out into a feature engineering step. By using a manual rather than automatic feature engineering approach, the features can be input into different model classes (e.g., neural networks, linear regression, random forests). Section 2.2 will discuss the three elements necessary to reproduce CNN accuracy are: nonlinear transformations of the input channels, use of predefined image kernels to capture spatial patterns, and use of an image pyramid to capture multi-resolution context. The interpretable approach produces maps where each pixel has a vector containing all the pieces of information used by the CNN. This mapping provides the desired interpretability by characterizing how data inside the network is being represented (*Gilpin et al. 2018*). Having extracted the CNN representation of the spatial context, the CNN can then be replaced with a different regression framework, such as a fully connected (dense) neural network (NN) or a linear regression. Use of linear regression represents the gold standard of interpretability because the resulting model has a weight for each input that tells us exactly how that input contributes to the prediction.

Thus, the fundamental approach of this chapter is knowledge distillation (*Hinton et al. 2015*) with a simpler proxy model that behaves like the original model but is easier to explain. The results are relevant to the original model provided the simpler model has a high degree of completeness relative to the original CNN model (*Gilpin et al. 2018*). At this point a quantitative definition of completeness is lacking and Section 2.3 provides several evaluations of model outputs as a function of the inputs to argue for completeness. Note that in this work, the linear model is being fit to the training dataset, as opposed to other interpretable approaches that use linear regression to approximate the full model in a local manner (LIME – Local Interpretable Model-Agnostic Explanations in *Ribeiro et al. 2016*). Fitting the model to the full dataset, gives information about the global properties of the model, not just local sensitivity about a particular input state.

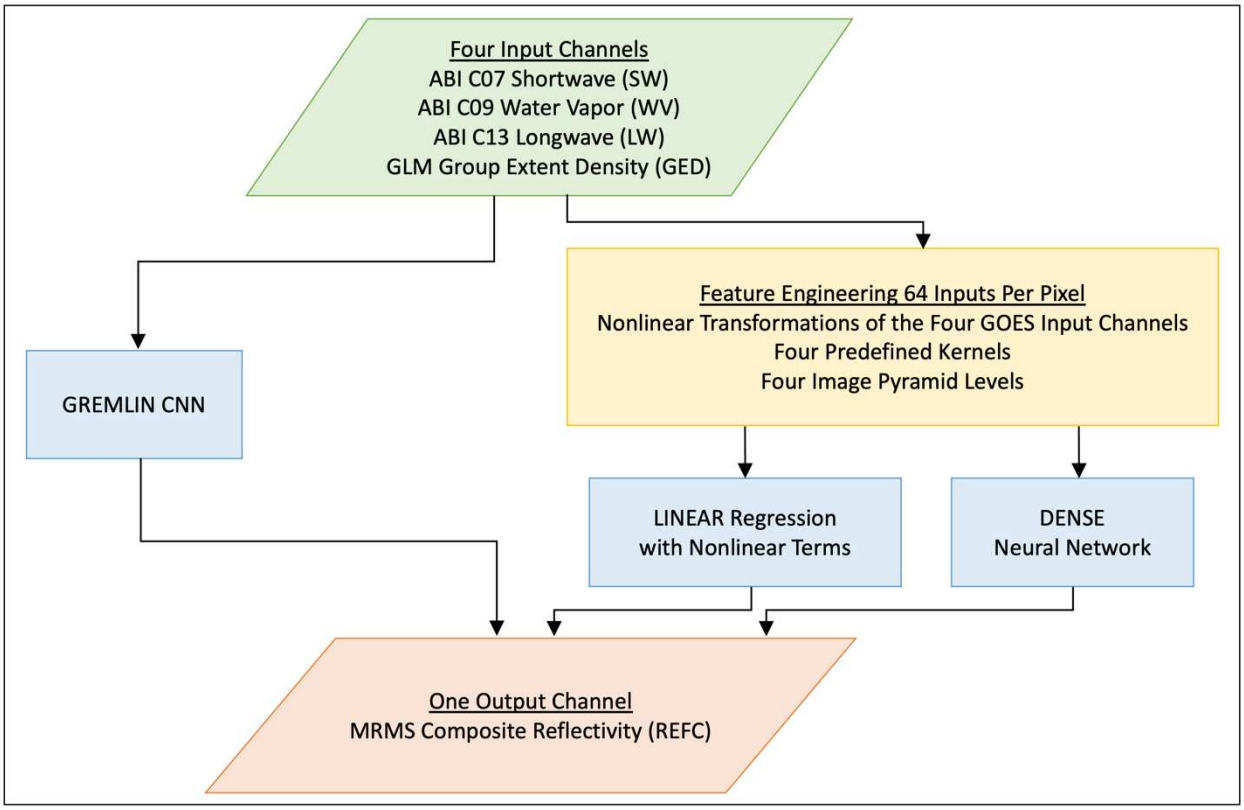

Figure 2.1. Schematic comparing the original convolutional neural network approach (left branch) versus the interpretable framework (right branch).

The outcome of this chapter is an Interpretable GREMLIN model. The term "interpretable" is used deliberately to indicate that the explainability has been built into the model right from the start (*Du et al. 2020* call this intrinsic interpretability), rather than being derived post-hoc from a trained black box model as with XAI. The terms interpretable and explainable are used in the same sense as *Došilović et al. (2018)*, where interpretable indicates mapping of concepts into a human understandable domain, while explainable indicates the contributions of a collection of features towards an output. Similarly, *Doshi-Velez and Kim (2017)* define interpretability as the ability to explain or to present in understandable terms to a human. *Flora et al. (2022)* provide a survey of the use of the terms interpretable and explainable in ML literature.

The main advantage of an interpretable model is that it provides building blocks to enable the trustworthiness needed to serve as an input to decision making activities. *Rudin (2019)* argues that for high stakes decisions, it is better to use a model that is interpretable from the start, rather than trying to explain a black-box model using post-hoc techniques because XAI has many pitfalls (*Molnar et al. 2022*) and can provide unreliable and misleading explanations. Also, the interpretable model offers a path forward towards understanding model biases, which is needed for ethical AI (*McGovern et al. 2022*). The main drawback of the interpretable methodology is that it explodes the size of the datasets involved. So, either more computer memory is required, or streamwise estimations approaches that do not hold the entire dataset in memory are required. Thus, this work confirms, as discussed in *Rudin (2019)*, that interpretable models require more effort to construct in terms of computation and domain expertise.

Consideration of these advantages and disadvantages to interpretable methodologies highlights that CNNs are very efficient implementations: they are quick and easy to train and

have small memory footprints. This work finds that developing linear regression models capable of replacing machine learning models requires substantially more effort than the traditional CNN development process. In addition, because of the bias-variance tradeoff in ML model development, the end user may be willing to sacrifice interpretability if it means reducing model bias. So, this chapter is not advocating that all CNNs be reformulated as described herein; there may be cases where that is not desirable. Instead, the interpretability approach in this chapter provides a new tool for analyzing satellite data and for understanding more complex ML models.

The chapter is structured as follows. Section 2.2 describes the data and methodology including use of an image pyramid (2.2.1), the prescribed convolutional kernels (2.2.2), nonlinear transformations used in data preprocessing (2.2.3), the linear regression model (2.2.4), finding the weights of the regression model (2.2.5), and handling unbalanced data (2.2.6). Section 2.3 provides results and discussion, confirming that the interpretable model matches the accuracy of the CNN using several metrics (2.3.1), providing an interpretation of the features and explaining spatial context (2.3.2), examining information in the gradient direction (2.3.3), multi-resolution information (2.3.4), multi-channel information (2.3.5), and temporal consistency (2.3.6). The chapter closes with summary and conclusions (Section 2.4).

2.2. Data and Methodology

The dataset used to train and evaluate GREMLIN's estimates of radar reflectivity is described in detail by *Hilburn et al. (2021a)*. It consists of brightness temperatures from three bands (Table 2.1) of the Advanced Baseline Imager (ABI, *Schmit et al. 2017*) and lightning group extent density (GED) from the Geostationary Lightning Mapper (GLM, *Goodman et al. 2013*) on GOES-16. All datasets were resampled to the High-Resolution Rapid Refresh (HRRR) CONUS 3 km grid using a drop-in-the-bucket technique (all data samples that fall within a grid

cell are averaged together) and matched in time to a 15-minute refresh rate. The purpose for resampling onto the HRRR grid was because the original use case for GREMLIN was initializing convection in the HRRR model. Parallax shift in ABI imagery was removed assuming a fixed 10 km cloud top height. While this is reasonable for deep convection over CONUS, it will overcorrect the shift for shallow storms. GREMLIN was trained to MRMS (*Smith et al. 2016*) composite reflectivity (REFC), which corresponds to the maximum reflectivity found in the available column data from the U.S. surface network of weather radars. There are other channels on ABI and other parameters from GLM that have been found to be helpful for GREMLIN, but they are not used in this work to be able to directly compare Interpretable GREMLIN results with the GREMLIN CNN. The dataset, named *CONUS2* (*Hilburn 2022*), was reduced to 1798 training and 448 testing samples by selecting 256 x 256-pixel image patches over a 92-day period from April to July 2019 to maximize the number of storm reports.

Table 2.1. ABI bands and their nicknames used in this chapter.

Number	Wavelength (μm)	Nickname
C07	3.9	Shortwave
C09	6.9	Water Vapor
C13	10.3	Longwave

2.2.1. Image Pyramid

The GREMLIN CNN architecture shown in Figure 2.2a is a simple version of a U-Net (*Ronneberger et al. 2015*). The model has four input channels, one output channel, is four layers deep, uses Convolution-Pooling blocks, and has 32 3x3 kernels in each convolutional layer. The key insight enabling Interpretable GREMLIN is that this architecture corresponds to the combination of an image pyramid (Figure 2.2b), a filter bank, and a regression framework. The image pyramid (*Burt and Adelson 1983*, *Adelson et al. 1984*) is the mechanism by which GREMLIN captures multiresolution information content. The convolutional kernels (ie., the

filter bank) are what captures the gradients and spatial patterns. The regression framework is what models the nonlinearity between inputs and outputs. For Interpretable GREMLIN, there are also four input channels and a four-level image pyramid. In the GREMLIN CNN, the image pyramid is internally constructed through 2x2 MaxPooling layers in the encoder branch and 2x2 UpSampling in the decoder branch. For Interpretable GREMLIN (Figure 2.2c), 2x2 MaxPooling is applied from 0 to 3 times, then the kernels are applied, and UpSampling is applied from 0 to 3 times to get each pyramid layer back to the original resolution. In developing Interpretable GREMLIN, applying a 3x3 binomial smoother before each pooling and after each upsampling was found to yield better predictions, but note that no smoothing layers were used in the GREMLIN CNN.

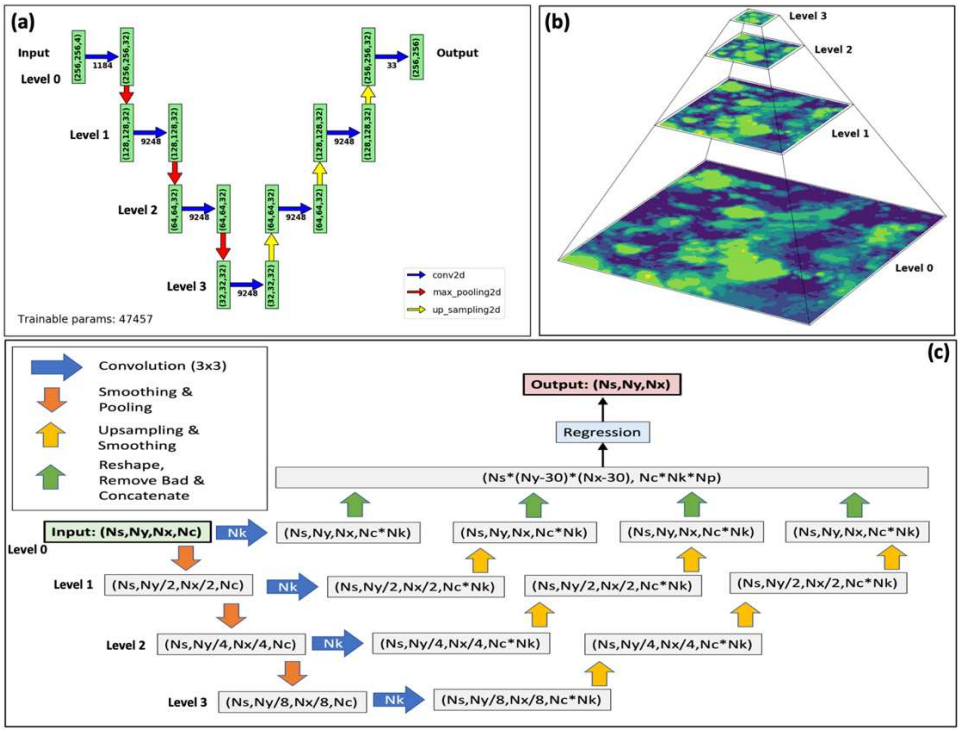

Figure 2.2. (a) GREMLIN model architecture where number of parameters are given under the blue arrows and image sizes are shown in green boxes. In this panel the convolutional kernels are learned. (b) Image pyramid corresponding to GREMLIN, where Level 0 is the original resolution input image, and three levels of pooling are applied. (c) The architecture of the interpretable model. In this panel the convolutional kernels are prescribed. Dimensions are number of samples N_s, number of channels N_c, number of kernels N_k, number of pyramid levels N_p, and the image dimensions N_x and N_y.

2.2.2. Convolutional Kernels

In the original GREMLIN CNN, the kernels are learned, while Interpretable GREMLIN has prescribed kernels. *Guilloteau and Foufoula-Georgiou (2020)* suggested that a small number of prescribed kernels can capture much of the information content learned by CNNs. Indeed, this work finds that a large filter bank was not necessary to reproduce GREMLIN accuracy, and in fact, for this application, only four kernels were found to be necessary to reproduce GREMLIN accuracy: identity (*I*), Sobel *DX*, Sobel *DY*, and Laplacian (*LAP*), given by the equations:

$$I = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{1a}$$

$$DX = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{1b}$$

$$DY = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{1c}$$

$$LAP = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1d}$$

$$SMOOTH = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{1e}$$

where Equation (1e) gives the smoothing kernel used in constructing the image pyramid. Such a simple filter bank may not be adequate for all types of problems, and at this point it is not possible to offer general guidelines for reformulating CNNs as interpretable models. Thus, as in ML development, there is a certain amount of trial-and-error that is involved to configure the interpretable version of the model. An advantage of these kernels is that they come with physical interpretations as edge and point detectors. The CNN learned many additional kernels with less obvious physical interpretations. Tests supplementing the model with additional kernels did not

find significant additional improvements to accuracy. The actual number of inputs for each pixel is then given by the product of the number of input channels N_c, the number of pyramid layers N_p, and the number of image kernels N_k. For Interpretable GREMLIN this is 4·4·4 = 64 inputs per pixel.

Bringing the internal workings of the CNN to the input preparation step provides the advantage that now the sensitivity of model outputs to model inputs can be determined, including the spatial context utilized by the CNN. It comes with the disadvantage however that now the input dataset is much larger; by a factor of 16 in this case. A minor point that is obscured with CNNs, but is obvious with the interpretable model, is that near image edges there is a loss of spatial context, unless some sort of edge padding is used. For this work, only pixels with full spatial context were utilized. For 3x3 kernels, 2x2 pooling, and 3x3 smoothing this results in dimensions of the valid data after each block of smoothing, pooling, upsampling, and smoothing of (*nx*-d_i, *ny*-d_i) where the pixels lost d_i for each pyramid level *I* follows the recurrence relationship

$$d_0 = 2 \tag{2a}$$

$$d_i = 2d_{i-1} + 2 \tag{2b}$$

so that the resulting images are reduced in size to (nx-30, ny-30), removing 15 pixels from each edge for this model with four pyramid levels.

2.2.3. Data Preprocessing

To keep the linear regression simple, it was found advantageous to apply a gamma correction (e.g., *Gonzalez and Woods 2002*) to the inputs to remove the mean nonlinearity, shown in Figure 2.3. The equations for scaling the inputs are given by

$$x_{scaled} = [(x - x_{min})/(x_{max} - x_{min})]^{\gamma} \tag{3a}$$

$$x_{scaled} = [(x_{max} - x)/(x_{max} - x_{min})]^{\gamma} \quad (3b)$$

where x_{min}, x_{max}, and the γ exponents are given in Table 2.2. The exponents were calculated by fitting a power-law to the mean radar reflectivity versus each of the four input channels. Equation (3a) is the regular scaling used for lightning, while Equation (3b) is the inverted scaling used for infrared bands. The water vapor band is the most linear input, and the lightning is the most nonlinear input. Note that the data were already scaled into the range 0-1 and applying an exponent to linearize the data is another way that the data preparation process resembles the same processes used to create RGB image products (e.g., *Miller et al. 2020*). The motivation behind the gamma correction is further discussed in Section 2.3.2.

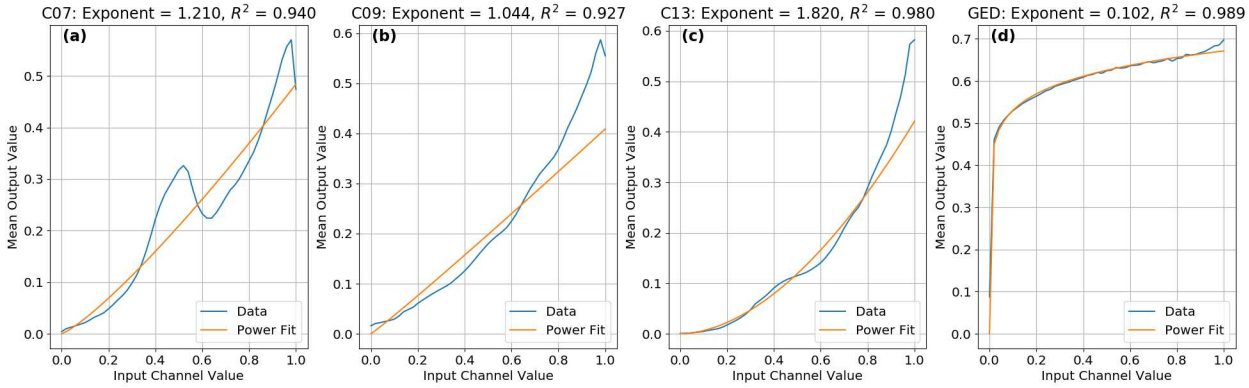

Figure 2.3. The mean radar reflectivity as a function of input value for the four input channels showing data (blue) and power law fit (orange) for (a) C07, (b) C09, (c) C13, and (d) GED.

Table 2.2. Scaling parameters for each input channel.

Channel	x_{min}	x_{max}	Exponent
C07	200 K	300 K	1.210
C09	200 K	250 K	1.044
C13	200 K	300 K	1.820
GED	0.1 groups (5 min)$^{-1}$ km^{-2}	50 groups (5 min)$^{-1}$ km^{-2}	0.102

2.2.4. Linear Regression Model

Given those inputs to the interpretable model, a regression framework is needed for making the predictions of REFC. Two approaches are used. The first uses a fully connected

dense NN (called DENSE), which serves as a nonlinear function approximator using a model with 2 hidden layers and 32 units/layer. Using this approach is a quick way to confirm that the interpretable model can indeed reproduce the accuracy of the original GREMLIN CNN architecture, and in some cases even performs better. However, a fully connected dense NN is not very interpretable, and so the second approach is to replace that with a linear model (called LINEAR). The DENSE model indicated that the ability to represent nonlinearity is an important contributor to ML accuracy, and this provided guidance on how much nonlinearity must be included in the linear regression. It was found that GREMLIN accuracy could be reproduced with a model of the form

$$y = \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{j \le i} w_{i,j} x_i x_j = \sum_{i=1}^{N} w_i z_i \tag{4}$$

where n is the number of inputs (x). This model includes linear functions of each input, two-way interaction terms, and quadratic functions of each input. The number of terms (z) in this model is given by $N = n + n(n+1)/2 = 2144$ for 64 inputs. Note that while linear regression is fully interpretable, the large number of features here means that this approach is technically not simulatable in the sense of *Murdoch et al. (2019)*. In other words, despite being a linear regression, the large number of terms hinders the ability to of a human to easily interpret the model, and additional methods are required to distill the model behavior into something digestible. Section 2.3 provides analysis along several dimensions of variability in the inputs to obtain interpretations.

An advantage of linear regression models over machine learning is that it easily allows calculation of the condition number κ of the model from

$$\kappa = \|C_x\| \|C_x^{-1}\| \tag{5}$$

where || || is the Frobenius norm. The condition number measures the sensitivity of a model output to changes in the inputs (e.g., *Press et al. 1992*). Starting with just the 64 linear terms in Eq. (4), the condition number is 3.1E+06. Adding the quadratic terms increases this to 4.6E+08 and including all the terms increases it to 2.1E+13. Thus, the linear model will be highly sensitive to noise in the inputs, and if one wanted to operationalize the linear model, further work on feature selection and regularization would be needed. However, what is important is that the process of constructing the linear model exposes an approximation of the effective input feature space of the CNN. It is the use of that input space that provides the interpretability shown and discussed in Section 2.3.

2.2.5. Finding Regression Weights

The standard approach to solving the generalized least squares problem (in Python, scipy.linalg.lstsq) is through applying singular value decomposition (SVD) to the normal equations. However, SVD involves calculating the inverse of a matrix that has the shape of the number of inputs (64) by the number pixels across all images (8E+7), which produces memory exhaustion. The standard approach for linear regression when the dataset is too large for memory is Stochastic Gradient Descent (SGD, in Python, sklearn.linear_model.SGDRegression. partial_fit). However, SGD was found to be prohibitively slow, and many passes over the dataset are required for convergence. Thus, the approach used to solve Equation (4) is employing the linear minimum mean squared error (MMSE) estimator.

From the orthogonality principle (*Papoulis and Pillai 2002*), the solution of (4) is

$$w^T = C_x^{-1} C_{yx} \tag{6}$$

where C_x is the autocovariance matrix and C_{yx} is the cross-correlation matrix. The memory required goes as the number of inputs and does not depend on the number of data points. These

matrices can be calculated with just one pass over the data, accumulating the sums of x_i, y, $x_i x_j$, and $y x_i$ for inputs i,j. This approach also has the advantage that ablation studies can be conducted without needing to re-fit the model: you simply drop the rows and columns in C_x and C_{yx} for the inputs you want to remove and recalculate w using Eq. (6). One might raise the question whether this type of model is properly called "machine learning". Since calculating the sums required to solve for the weights in Eq. (4) can only be performed by machine, this approach is perhaps the simplest form of machine learning. One important difference with ML is that this approach does not have an optimizer since weights are calculated explicitly, rather than through an iterative process.

2.2.6. Handling Unbalanced Data

The last detail in the methodology of Interpretable GREMLIN is dealing with the imbalanced nature of the dataset, given that a PDF of REFC falls off exponentially with increasing REFC. If not addressed, this behavior leads to a poor probability of detection / false alarm tradeoff, where strong echoes are severely underpredicted. Using a weighted MSE loss function, as in *Hilburn et al. (2021a)*, achieves a balance between underprediction and overprediction across the full range of REFC values. The loss function weights W are given by

$$W = e^{by^c} \tag{7}$$

where y is the true value (from MRMS), and the coefficients b and c are given for each model in Table 2.3. For the CNN and DENSE models, the weights were implemented in the loss function, and for the LINEAR model, it was implemented with weights for each observation. The weights were found, as in *Hilburn et al. (2021a)*, to produce the minimum categorical bias across the range of 5 to 50 dBZ in steps of 5 dBZ. The weights were found for the training dataset, to keep weights independent of the testing dataset.

Table 2.3. The number of parameters and loss function weights for each model.

	Parameters	b	c
CNN	47,457	5.0	4.0
DENSE	3,169	4.0	4.0
LINEAR	2,144	4.5	5.0

2.2.7. Evaluation Metrics

Model performance will be characterized using several metrics that compare the estimated (Y) and true (X) radar reflectivity pixel-by-pixel, then sum over the entire image. The Pearson correlation coefficient (R^2) and root-mean-square-difference (RMSD) are given by

$$R^2 = \frac{E[(X-E[X])(Y-E[Y])]^2}{(E[(X-E[X])^2])(E[(Y-E[Y])^2])} \tag{8a}$$

$$RMSD = \sqrt{E[(Y-X)^2]} \tag{8b}$$

where E is the expected value. Categorical statistics are calculated for a given REFC threshold by creating binary objects from both the predicted and true datasets using the greater-than operator, and then forming a contingency table of the hits H (echoes in both the predicted and true datasets), false alarms F (echoes in the predicted but not in the true dataset) and misses M (echoes in the true but not in the predicted dataset). The probability of detection (POD), false alarm ratio (FAR), critical success index (CSI), and the frequency bias (BIAS) are given by:

$$POD = \frac{H}{H+M} \tag{8c}$$

$$FAR = \frac{F}{H+F} \tag{8d}$$

$$CSI = \frac{H}{H+F+M} \tag{8e}$$

$$BIAS = \frac{H+F}{H+M} \tag{8f}$$

The statistics were calculated for the common subset of data in the center of the image with full spatial context (Section 2.2.2). Confidence intervals for each metric were estimated using

bootstrap resampling (In Python, sklearn.utils.resample). The resampling was performed on the level of samples (images), which produces larger confidence intervals than resampling on the level of pixels (i.e., having flattened all the images together). A total of 10,000 resamples were generated to bootstrap the distribution of the metric, and the 95% confidence interval was calculated by

$$CI = [\mu - 1.96\sigma, \mu + 1.96\sigma] \tag{8g}$$

where μ is the sample mean and σ is the sample standard deviation.

2.3. Results and Discussion

This section begins by showing how the interpretable model performance is materially similar to that of the CNN (Section 2.3.1). Then, the interpretable nature of the model will be used to interpret the input features (Section 2.3.2). The input space of the interpretable model will then be used to analyze all three models to explore the meaning of spatial context (Sections 2.3.3 and 2.3.4) and address the role of multi-channel information in Section 2.3.5.

2.3.1. Interpretable Model Performance

For the interpretable model to be useful, it must reasonably reproduce the accuracy of the CNN. To assess this performance in the context of nonlinearity, results are provided for an interpretable model that uses a dense neural network regression (DENSE) and the linear regression (LINEAR) described in Section 2.2.4. Figure 2.4 provides a comparison of the models using several performance metrics, and it shows that while the CNN has the highest overall accuracy, the interpretable models are not far behind and thus the interpretable models are mostly capturing CNN accuracy. The confidence interval of the R^2 (Fig. 2.4a) of the LINEAR model overlaps the DENSE model, and the DENSE model overlaps the CNN. The RMSD (Fig. 2.4b) of the interpretable models is higher than the CNN. The CSI (Fig. 2.4c) of all three models matches

well across reflectivity thresholds, except in the highest two bins for the LINEAR model. The BIAS (Fig. 2.4d) indicates that the interpretable models do exhibit somewhat more overprediction of echo areas at lower REFC. At higher REFC, the CNN and DENSE models have nearly identical performance, while the LINEAR model underpredicts the echo area. This disparity between DENSE and LINEAR models at higher REFC may suggest the importance of nonlinearity in reproducing the strongest echoes. Achieving that balance in a dataset where the frequency of occurrence falls-off exponentially is difficult, and it is possible that additional tuning of the interpretable models could bring the bias in line with the CNN, however for the purposes of this analysis, these results are comparable to the CNN. These results support the idea in *Rudin (2019)* that it is a myth there is always a necessary trade-off between accuracy and interpretability and that complicated models are required for top performance.

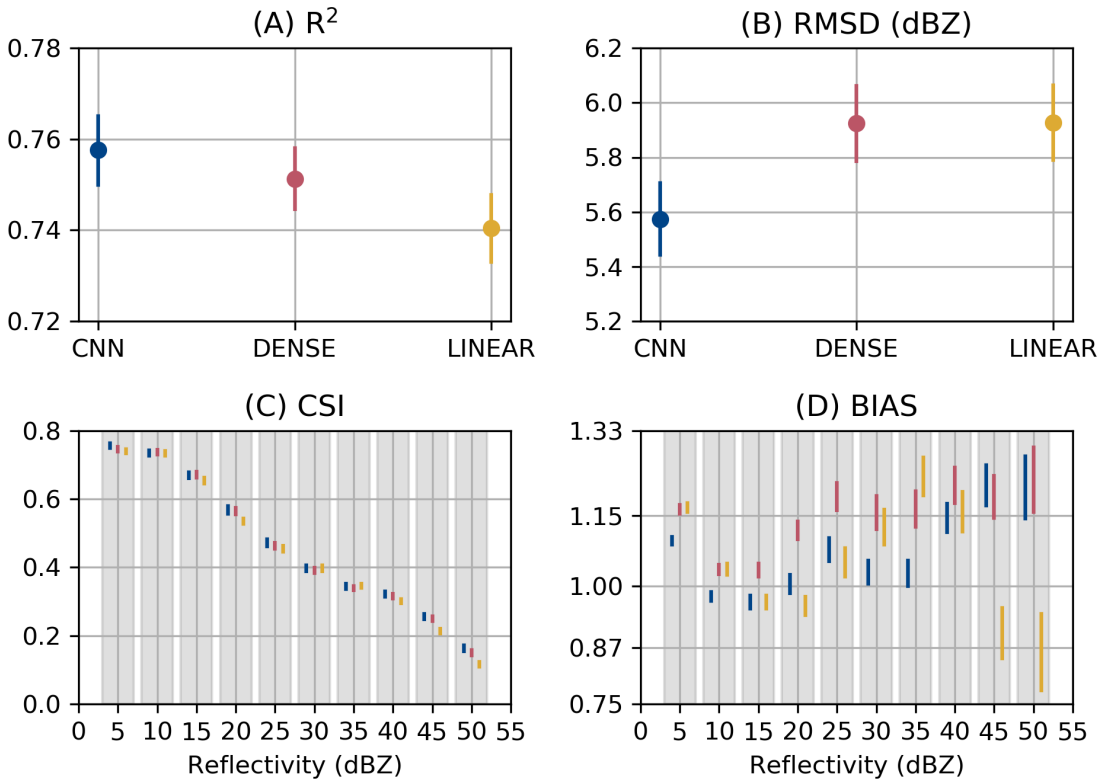

Figure 2.4. Performance statistics for the three models (CNN in blue, DENSE in red, and LINEAR in yellow), calculated from the testing dataset: (a) R^2, (b) RMSD, (c) CSI, and (d) frequency BIAS.

The interpretable models perform reasonably in a statistical sense, but it is essential to consider the spatial variability of model predictions to get a sense for how well the models perform where it matters the most in rendering realistic meteorological features. Figure 2.5 shows an example test case for the models. The tendency for overprediction of echo area is evident in the interpretable models, although in this case it provides significantly better POD than the CNN, and only slightly worse FAR. This yields better representation of the stratiform area east of Akron, Ohio (marked by "A") in the interpretable models. Note that the interpretable models have more detail in the spatial variability than the CNN, although changes to the CNN architecture could likely improve that, such as adding skip connections or adding additional convolutional layers after the last UpSampling layer. But it is encouraging that the interpretable models are following the patterns in the GOES input data (not shown). Note that while all the ML models get the basic distribution of weak and strong echoes correct, none of the ML predictions capture the same fine-scale details seen in MRMS, and thus meteorologists should be cautious about overinterpreting the meaning of a particular detail in the ML predictions.

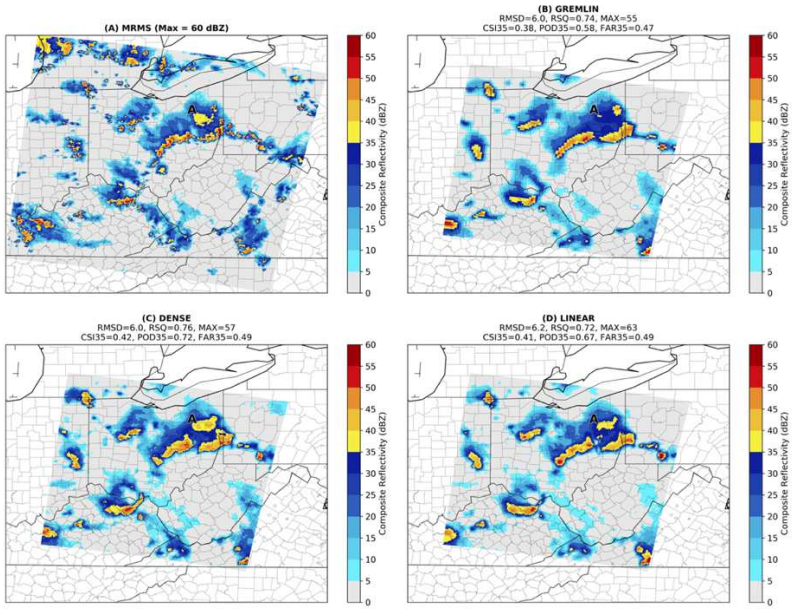

Figure 2.5. REFC for (a) MRMS (truth), (b) GREMLIN CNN, (c) DENSE interpretable model, (d) LINEAR interpretable model. Akron, Ohio is indicated "A".

2.3.2. Feature Interpretation

The motivating question of this research is what exactly is the spatial context used by CNNs to make accurate predictions of REFC? Having brought the kernel convolution and multiresolution representation from inside the CNN to the input space, this question can finally be answered. Figure 2.6 provides an overall summary of how the mean output varies as a function of each of the individual inputs. This figure, and subsequent figures in Section 2.3, were computed using the same methodology. The figures are showing the mean output, which comes from MRMS data and predictions from the CNN, DENSE, and LINEAR models. The input value comes from the interpretable model data pre-processing (Sections 2.2.1, 2.2.2, 2.2.3), and thus can be applied to analyze the MRMS data and CNN predictions as well. In other words, by exposing an approximation of the input space, it can be applied to analyze all models by computing bin-average statistics. Each panel in Figure 2.6 is a different combination of GOES input and kernel, and the figure shows results for the Level 0 of the image pyramid (results for the other levels are discussed in Section 2.3.4). Keep in mind that the ABI TBs were inversely scaled, so zero is warm and one is cold. The bin averages were constructed from the training dataset to characterize how well the models fit the training data and used a bin width of 0.02.

The identity kernel (left column of Figure 2.6) shows the first two of GREMLIN's strategies that were also identified by LRP: pixels with colder TBs tend to have stronger echoes, and lightning is associated with stronger echoes. There are small vertical offsets of the models relative to the data, which is the result of weighting the stronger echoes more using Eq. (7). The LINEAR model captures well the shape of the lightning nonlinearity (Figure 2.6m), which is because of the gamma correction described in Section 2.2.3, without which, the curve would

bend downwards for scaled GED above 0.5 and would lead to underprediction of the strongest echoes. In fact, it was Figure 2.6m that revealed the need for the gamma correction.

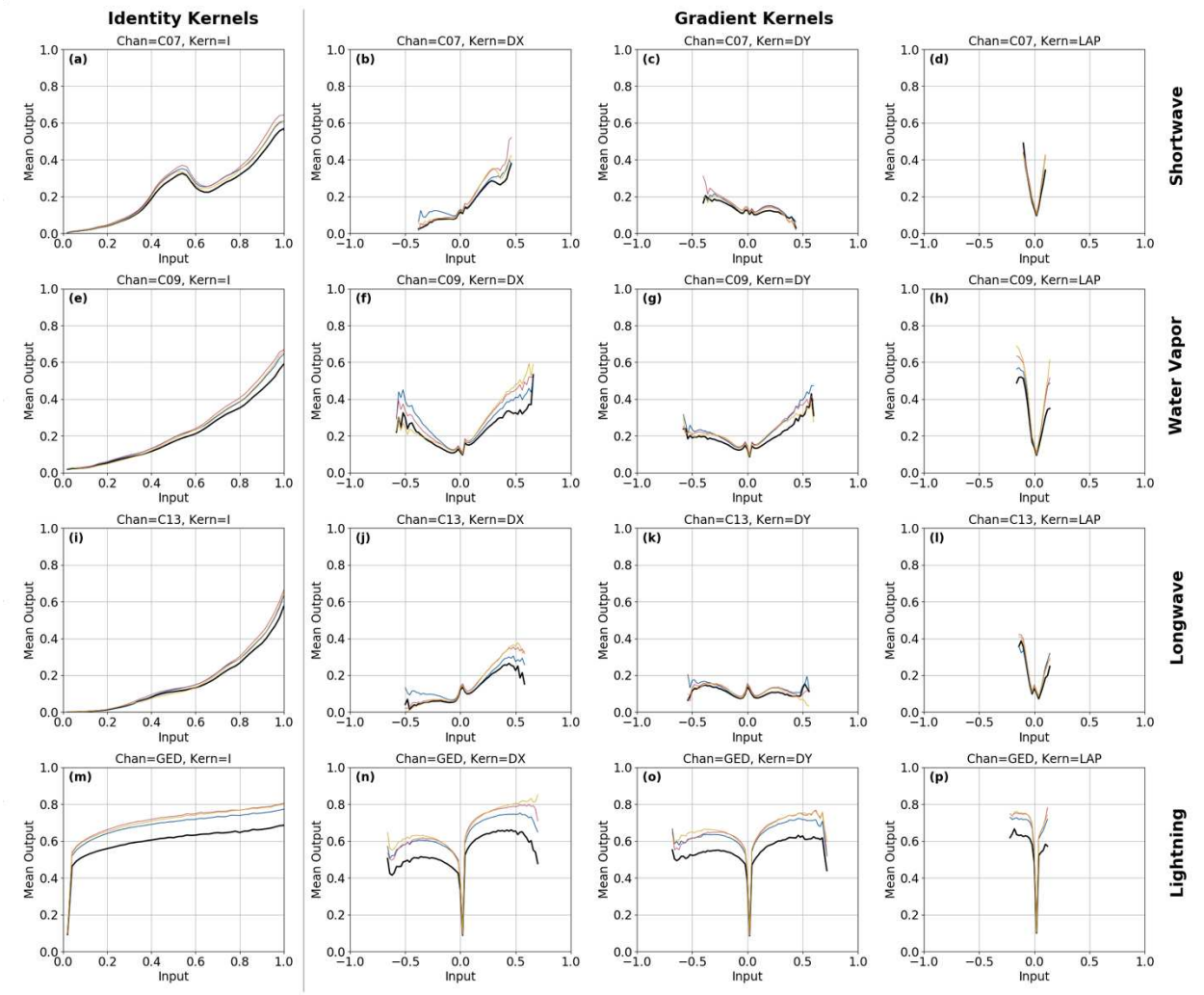

Figure 2.6. The mean output versus each input for Level 0 of the image pyramid. Shown are the MRMS data (black), CNN (blue), DENSE (red), and LINEAR (yellow). Each row is a different input channel, and each column is a different image kernel. A bin width of 0.02 is used, and bins with less than 10 points are masked.

Trying to use LRP to diagnose such an issue would be nearly impossible, and thus the interpretable model is already providing much more information that can be used to understand its performance and improve predictions. Comparing the LINEAR and DENSE models to the CNN provides insights into the important role of nonlinearity in model performance, a factor which is completely obscured in the CNN. The performance of a CNN is tuned through the

number of filters per layer and the number of layers in the model, which commingles factors related to pattern representation and nonlinearity. In the interpretable models, these factors are separated and can be tuned separately, providing more flexibility to create the best possible model with the smallest number of trainable parameters.

The gradient kernels (Figure 2.6 rightmost three columns) show the third strategy that was also found by LRP: strong gradients are associated with stronger echoes. For the Laplacian kernel (Figure 2.6d,h,l,p) and for the water vapor (Figure 2.6f,g) and lightning channels (Figure 2.6n,o), there is a minimum at zero, with increasing mean output as the gradients become stronger. The shortwave (Figure 2.6b,c) and longwave (Figure 2.6j,k) channels follow a similar form, but the minimum is not at zero, with an asymmetry in the model response to gradients that depends on the direction of the gradient. In other words, replacing the individual *Dx* and *Dy* gradient components with just the gradient magnitude would produce a less accurate model. The source of this asymmetry is likely related to the wind shear sampled in the training dataset. For example, for a situation where the dominant flow is from the southwest, there will be stronger brightness temperature gradients on the upshear side of the storm (i.e., the southwest side) compared to the downshear side of the storm (i.e., the northeast side) due to the smoothing effects of the anvil blow-off. Since all these models are trained over CONUS warm-season convection, it would suggest that if the models were applied to a different region, with a significantly different preferred wind direction, larger errors would likely be observed because the model would not generalize to unseen wind shear regimes. That might suggest wind shear should be included as a training parameter. Section 2.3.3 will investigate gradient direction information further.

Figure 2.6 provides a very useful overall characterization of the different versions of the

GREMLIN model, and it also demonstrates the three strategies learned by the CNN that were identified by LRP. However, Figure 2.6 does not tell the whole story of what is being learned because the 1D bin averages do not show interactions among variables. Once such interactions are considered, this becomes a very high-dimensional space, and fully characterizing the model would require more figures than can fit in a journal article. However, there naturally are correlations among channels, among levels, and among kernels, which mean that taking representative slices through the space can convey the important relationships. The slices are chosen based on domain knowledge, and although there are other less subjective ways to explore the high-dimensional space (e.g., t-SNE, *van der Maaten and Hinton 2008*), there are 64·63/2 = 2016 ways to make 2D slices through the 64-dimenional space, so the problem is overwhelming (not to mention higher-dimensional slices as well). Instead, the remainder of this section compares the performance of the different models using 2D bin averages to examine two aspects of spatial context: directional information (Section 2.3.3) and multiresolution information (Section 2.3.4); and to examine multi-channel information (Section 2.3.5).

2.3.3. Directional Information

It has already been shown that stronger TB gradient magnitudes are generally associated with stronger radar echoes. But an interesting question is whether the gradient directions also contain information, and whether this has a relationship with wind shear. Figure 2.7 provides the mean output as a function of both -*Dx* and -*Dy* kernel inputs together where the minus sign puts the directions in the meteorological direction convention (0° represents wind from the north and a gradient direction pointing to the south). All the models are learning a directional preference in the data: radar echoes are strongest when gradients have southwest orientation. For longwave (Figure 2.7a,b,c,d), which captures weaker echoes, the maximum mean output is in the west-

southwest direction; while for GED (Figure 2.7e,f,g,h), which captures stronger echoes, the maximum is in the southwest direction. The models show a stronger REFC response than the data, which is a consequence of using weighted loss functions to emphasize the strong echoes. A directional histogram for radar echoes exceeding 35 dBZ (not shown) has a mode at 201° (south-southwest) with the peak between south to southwest directions. The vertical wind shear, estimated from the HRRR 250 hPa wind components, has a mode at 250° (west-southwest) with most samples between south to northwest directions. This is suggestive that the directional relationships learned by these models are specific to CONUS, and if these models were applied to different wind shear regimes, such as the tropics, this directional information may not generalize. It would be difficult to get this insight from LRP, since LRP only considers one sample at a time, but consideration of the whole dataset is needed for this lesson to emerge. At this point it is just a hypothesis, and it would take examination (using an analysis like Figure 2.7) of models that are trained on a set of tropical samples to verify it.

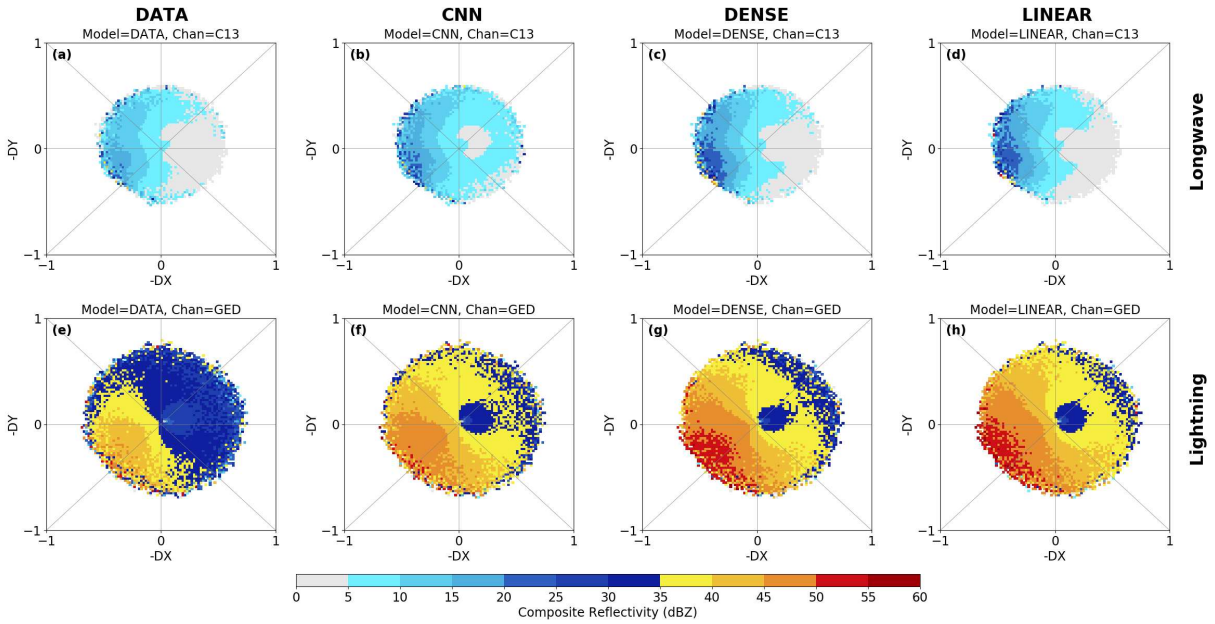

Figure 2.7. The mean output (color fill) versus the *Dx* and *Dy* kernel inputs for the longwave (top row) and lightning (bottom row) channels and each model (columns left to right: data, CNN, DENSE, and LINEAR) for Level-0.

2.3.4. Multiresolution Information

Figure 2.6 shows results for the base of the image pyramid (Level 0), and in an approximate sense, the results for other levels of the pyramid are similar, but the curves get progressively squashed down towards the x-axis (not shown). That is because the higher levels of the pyramid contain values that correspond to a larger portion of the image, and thus there is averaging over a greater range of output values for a given input value. However, the more interesting story emerges when considering the relationships between two different pyramid levels, which is how multiresolution information is encoded in the model.

Figure 2.8 presents results for the base of the pyramid (Level 0, highest resolution) versus the top of the pyramid (Level 3, lowest resolution). Figure 2.8a,b,c,d shows that not only are strong echoes associated with cold TBs but are maximized when the TBs are warmer on other pyramid levels. This implies a relationship between the strongest echoes and the distance from the cloud edge, determined by the pair of levels being considered. Even though Figure 2.8 is for the identity kernel, the use of image pyramiding is capturing information in spatial variability (results for the gradient kernels are noisy and not shown). Cold TBs on Level 0 implies positioning on the interior side of the cloud edge, while warmer Level 3 TBs implies that the cloud edge is nearby, and warmer ground pixels are contributing. There is a relative minimum along the 1-1 line when TBs are cold on both pyramid levels, reflecting the uncertainty in locating strong radar echoes when cold TBs cover large areas, unless there is texture in the TBs or lighting to provide clues.

Lightning has a somewhat similar pattern (Figure 2.8e,f,g,h), but not as strongly bifurcated along the 1-1 line. The weakest echoes are associated with little to no lightning on Level 0, but as the Level 3 lightning increases, it indicates nearby lightning, making moderate

echoes more likely. Thus, the use of multiple image pyramid levels as inputs to these models provides the capability to construct multiresolution representations of the phenomena. In other words, it provides the model the ability to locate strong echoes some distance inwards from cloud edge. Pixelwise retrieval methods simply cannot make use of this multiscale information but given how strong the mean response is in longwave, exceeding 50 dBZ in certain situations, this is clearly important to extracting full value from GOES-R Series observations.

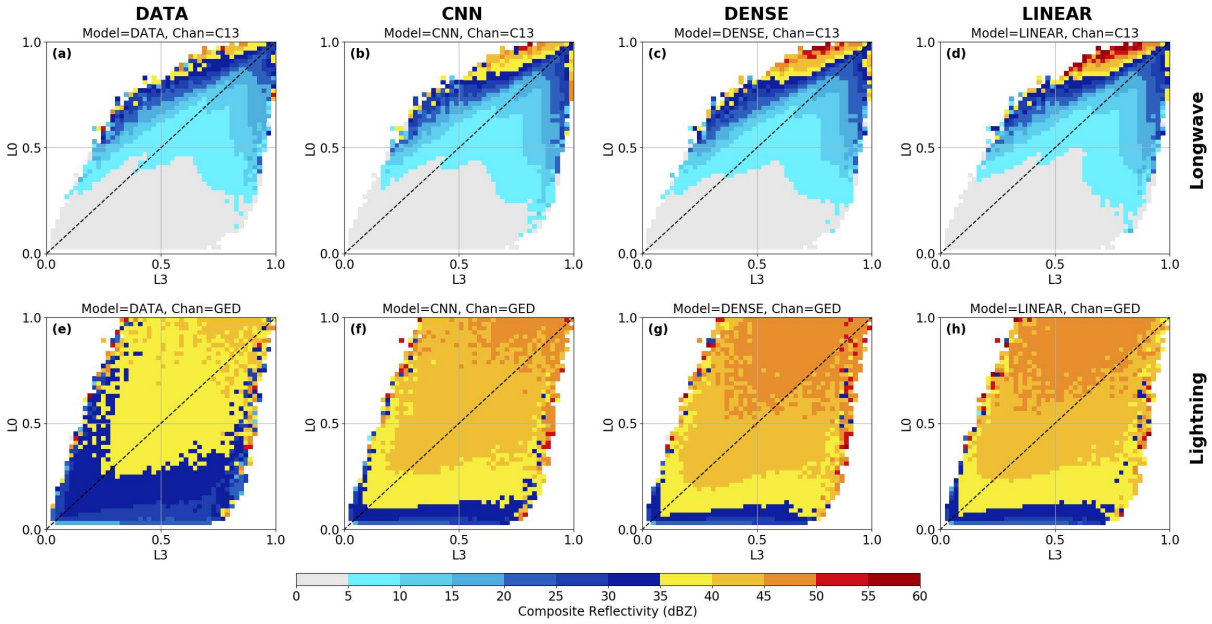

Figure 2.8. The mean output (color fill) vs Level-0 and Level-3 input values for the longwave band (top row) and lightning (bottom row) for each model (columns left to right: data, CNN, DENSE, LINEAR) for the identity kernel.

2.3.5. Multi-channel Information

The interpretable approach also provides new insights regarding the information content coming from multiple channels together. Figure 2.9 provides results for key channel combinations of interest and uses physical units to simplify the interpretation. While cold longwave TBs are associated with stronger echoes, that relationship is stronger for certain ranges of the shortwave infrared band (3.9 μm), which has a bimodal distribution with maxima near 210 K and 250 K due to night versus day. The shortwave band has a solar reflected component during

daytime, which augments the equivalent blackbody temperature (i.e., brightness temperature), leading to warmer brightness temperatures during daytime. Strikingly, the data and models have a feature with REFC > 40 dBZ occurring for cold longwave (< 210 K) but also very warm shortwave (> 275 K). Examining samples with those features reveals that the value in shortwave comes from the fact that it can see through thin cirrus better than longwave. When the warm shortwave infrared and cold longwave signature appears, it is because shortwave infrared can see through thin cirrus in the breaks between storms that have warmer surface contributions, while longwave only sees one large area of cold cloud. Examining the visible band for daytime cases confirms that shortwave infrared is seeing cloud edges and breaks between clouds that are obscured by very thin cirrus in longwave.

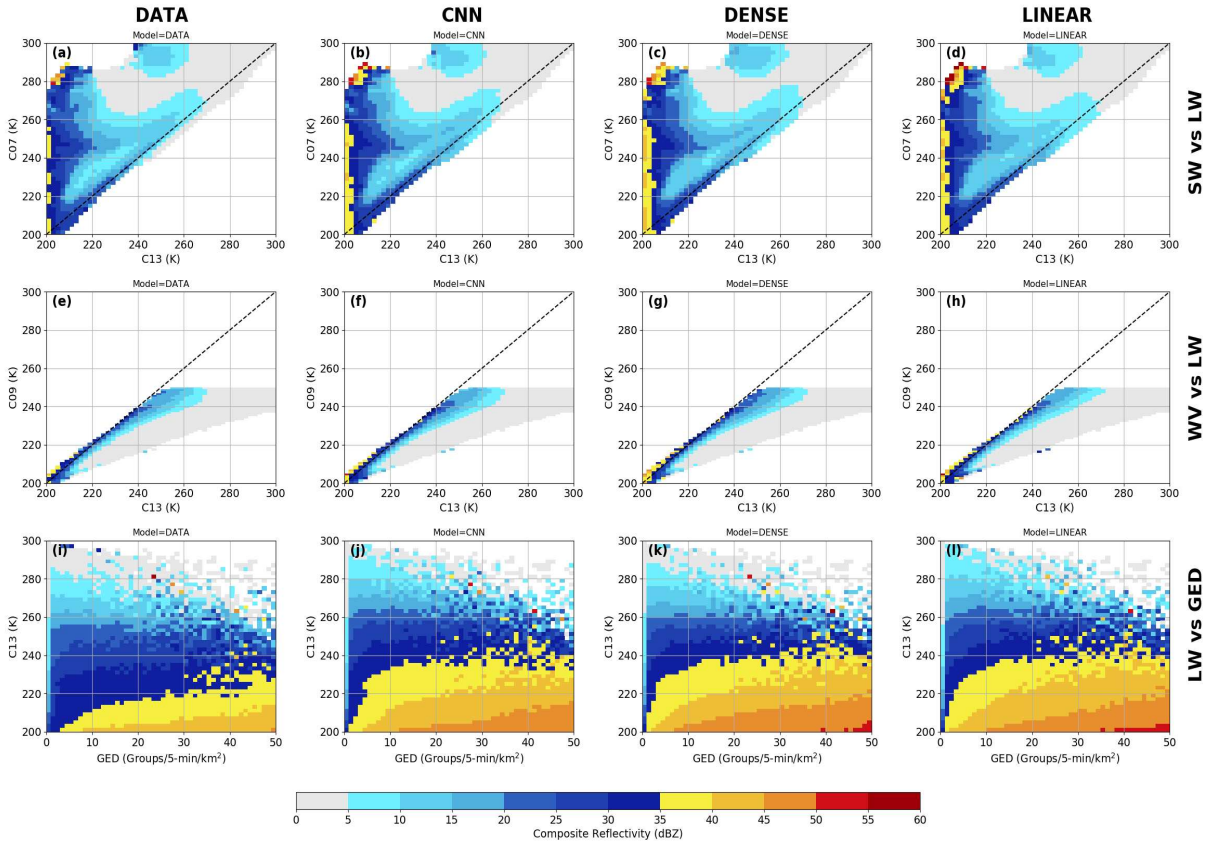

Figure 2.9. Mean output (color fill) vs channel combinations shortwave and longwave (top row), water vapor and longwave (middle row), and longwave and lightning (bottom row) for each model (columns left to right: data, CNN, DENSE, LINEAR) for Level-0 and the identity kernel.

Figure 2.9 also shows a familiar relationship where the strongest echoes are associated with small differences between the water vapor and longwave bands (*Schmetz et al. 1997*). This channel difference has been found to be valuable for distinguishing shallow from deep precipitation (*Kurino 1997*) and overshooting tops (*Bedka et al. 2012*). Usually, longwave is warmer than water vapor because longwave has greater sensitivity to the lower troposphere, however once clouds grow deep enough and thick enough, the difference goes to zero. The strongest echoes occur when water vapor is warmer than longwave, which is indicative of water vapor in the stratosphere (i.e., overshooting tops).

Finally, Figure 2.9 shows that while strong echoes are associated with cold TBs and lightning individually, when considered together, they provide additional information that can reduce false alarms. For example, when TBs are very cold, but there is little lightning, strong echoes are relatively less likely, for example in thin cirrus. Similarly, when lightning is strong, but TBs are warm, strong echoes are also relatively less likely. Physically, this situation can occur when the light from lightning reflects off nearby low clouds (*Wolf 2018*).

2.3.6. Temporal Consistency

One of the motivations behind the development of an interpretable model is the observation that loops of machine learning derived imagery can appear to be "jittery". This is most noticeable over stratiform precipitation regions where predicted echoes can blink on and off, which makes sense because these are the regions where the infrared band has the least information content. Quantifying temporal consistency is difficult (*Hilburn et al. 2021b*), but the most successful approach has been to examine time series of echo properties over an area that is small enough to capture a jittering area, but large enough so that storm motion is not a major contributor to the statistics. The hypothesis was that more complex models (i.e., greater number

of trainable parameters) might tend to have greater sensitivity to noise in the inputs and thus produce more jitter between images. However, the timeseries provided in Figure 2.10 suggest that is not the case. All three models (CNN, DENSE, LINEAR) follow the overall MRMS trends fairly well, and all three models have "choppier" time series than MRMS, with small fluctuations up and down about the slower moving trend. This shows the overall model complexity, which can be characterized by the total number of trainable parameters, does not by itself explain temporal jitter in predictions. Thus, the jitter is more likely due to noise inherent in the input data, rather than coming from the model. In fact, the CNN appears to have the smallest fluctuations, which could suggest that the CNN is learning kernels with superior noise suppression to the derivative kernels used in the LINEAR and DENSE models. Thus, it appears the model condition number (i.e., the amount of regularization in the model) is a more important factor in explaining temporal jitter.

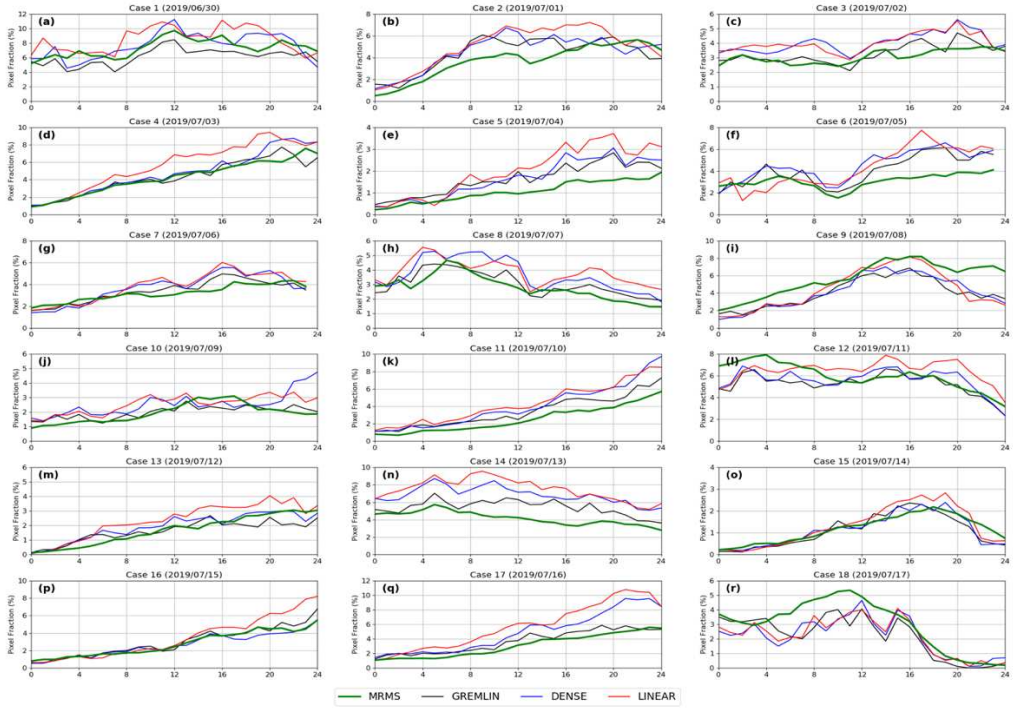

Figure 2.10. Timeseries of the fraction of pixels that have REFC > 35 dBZ for the 18 test cases (panels a-r), where each case has 25 timesteps, separated by 15 minutes. Shown are MRMS (green), CNN (black), DENSE (blue), and LINEAR (red).

2.4. Summary and Conclusions

This chapter used Interpretable AI to take a deeper look at the GREMLIN CNN that transforms GOES-R Series radiances and lightning into synthetic radar reflectivity fields. A knowledge distillation approach was taken and found the original 47,000 parameter CNN could be replaced by feature engineering plus a linear regression with as few as 2,000 parameters without a substantial loss of accuracy. The CNN used 32 learned filters with seven convolutional layers, resulting in a total of 5,280 different kernels across the CNN. However, the DENSE and LINEAR models were able to reasonably reproduce the accuracy of the CNN with just four prescribed kernels. This provides evidence that in the GREMLIN application, a huge filter bank is not necessary for explaining the CNN's accuracy.

The interpretable approach took the image pyramiding and kernel convolutions happening inside the CNN and brought them out to the preprocessing stage through feature engineering. This allows examining the sensitivity of the output to the individual inputs. This showed that the spatial context utilized by the CNN to make predictions is a combination of (1) spatial patterns identified by the four kernels and (2) a multiresolution representation provided by an image pyramid. Both factors are important for the additional accuracy provided by the CNN relative to a purely pixelwise approach. The interpretable approach has the benefit of disentangling the role of nonlinearity in CNN accuracy. In the CNN, the ability to resolve spatial context and represent nonlinearity are inherently mixed together when specifying CNN model architecture parameters. The nonlinearity was found to be an important factor, especially for the accuracy in strong echoes.

It should be noted that the interpretable approach to deconstructing a CNN taken herein will not necessarily work for all CNNs. For example, applying this approach to a CNN that

creates super-resolution Advanced Technology Microwave Sounder (ATMS) imagery was not able to successfully reproduce the CNN skill. This indicates that certain computer vision problems are harder than others and that CNNs can still outperform human feature engineering for certain tasks. This may have to do with the interpretable approach making predictions on a pixelwise level, and thus lacks the same spatially-aware decoder layers as a CNN.

Explainable AI (specifically LRP) had helped to identify three strategies of the GREMLIN CNN: (1) presence of lightning, (2) cold brightness temperatures, and (3) strong brightness temperature gradients. The Interpretable AI model developed herein was able to identify an additional five strategies. (4) Gradient direction: stronger echoes are more likely when gradients had a southwestern orientation. This is also the predominant wind shear direction in the training data, which raises questions how well such directional information will generalize to different regimes. (5) Multi-scale information: while stronger gradients are associated with stronger echoes, the stronger echoes are more likely when the gradients on other scales are weaker. (6) Shortwave-longwave multi-channel information: using warm shortwave TBs to find cloud edges where thin cirrus obscures them in longwave. (7) Longwave-water vapor multi-channel information: deep convection with stronger echoes is associated with small differences between these channels. (8) Longwave-lightning multi-channel information: cold TBs are more likely to have strong echoes when also associated with high flash rates. There was some evidence for strategy (8) from LRP, when lightning was set to zero the CNN changed how it interpreted the TBs, but the interpretable model clarified how. Since LRP provides heatmaps for each channel individually, it was not well suited for identifying the multi-channel strategies that were obvious using the interpretable model.

A potential concern with this approach is the Rashomon effect (*Rudin et al. 2022*), in

which a set of different models with similar performance may exist. That means one cannot just assume that what is learned by the simpler model is necessarily valid for the complex model even if they have similar performance. However, the analysis presented in Figures 2.5 – 2.9 shows that the CNN and simpler models are learning similar behavior, which is also seen in the data. The simpler models do not capture all aspects of the CNN, in particular the sequential application of convolutional filters and the use of a spatially aware decoder. So, it is not appropriate to conclude that the simpler models can capture every aspect of the CNN model.

It is important to point out that GREMLIN was trained over CONUS and this chapter only evaluates over CONUS. Thus, it is premature to draw conclusions regarding regime dependence of GREMLIN performance. However, the linear model approach used herein could naturally be extended to generalized additive models (GAMs) for a more complex treatment of nonlinearity (*Rudin et al. 2022*). By replacing the fixed coefficients in Eq. (4) with coefficients that vary by some regime characterizing variable (*Zhou and Hooker 2022*), it might be possible to extend the interpretable model to different regimes. This seems like a good fit for meteorological regime dependence where the underlying model does not change, but where thresholds may vary with regime. An alternative approach for incorporating regime dependence is a mixture of experts (*Hinton et al. 2015*) where different specialist models are trained on different subsets of the data (i.e., different regimes). That approach would make more sense where the underlying structure of the model does change significantly with regime.

In closing, this work is not intended to be an end-state for interpretability, but a beginning. Interpretable GREMLIN helped identify several new strategies that were not found using XAI, and based on physics, some strategies confer a greater degree of confidence than others. Extending this research, it ought to be possible to combine Interpretable GREMLIN with

an explanation producing system so that users could decide for themselves whether they chose to believe a particular prediction or not. However, should that be expressed in terms of a confidence flag as is common in satellite meteorology, or in terms of discrete scenarios (i.e., storylines in the sense of *Shepherd 2019*) depending on the strategy employed? A question is how would that work for a model such as GREMLIN that has images as outputs when "screen real estate" is so limited in Weather Forecast Offices? One possible approach is the ProbSevere approach (*Cintineo et al. 2018*) where mousing over a particular echo object provides metadata about what informs the prediction. However, this highlights the larger issue that interpretability is not just about physical science, but about social science, and input from users is required to determine what constitutes a good explanation for a particular application (*Miller 2019*). In other words, developing AI is the starting point, and it is the human-AI interaction that matters for decision making.

CHAPTER 3: GLOBAL GREMLIN

3.1. Introduction

The development of GREMLIN has focused on the CONUS region where abundant radar data exists for evaluation. However, the maximum value of GREMLIN will be for areas of the Earth lacking ground-based radar coverage. The latest version of the RAP model grid covers a much larger area than the previous version, and most of the new domain lacks ground-based radar. In addition to the original use case of initializing NWP models, GREMLIN could provide value to meteorologists for situational awareness and nowcasting over much of South America, which lacks good radar coverage in many areas. There are also a host of operational (nowcasting) needs for oceanic weather information, ranging from Coast Guard, to Navy, to ship-based commerce and transportation. Even manned space vehicle recovery! Thus, there is a significant need for geostationary satellite-based synthetic radar covering areas of the Earth that were not part of the original GREMLIN training dataset.

Extending any machine learning model outside of the regimes on which it was trained is prone to have unpredictable results. Relationships between inputs and outputs may change due to covariance shifts in the data, and this is certainly the case for infrared brightness temperatures and lightning (the most impactful predictors in GREMLIN) relative to radar echoes. *Liu and Zipser (2013)* document the significant structural differences between convection over land versus ocean. *Williams and Stanfill (2002)* examine various physical reasons why there is a pronounced contrast between lightning over land versus ocean. In addition to the amount of lightning, it is known that the characteristics of the lightning flashes themselves are different over land versus ocean (*Peterson et al. 2017*). *Rocque (2023)* documented relationships between

hail and lightning over South America that are very different from those over CONUS. Thus, to produce trustworthy predictions, GREMLIN must be validated for these different regimes. Depending on the results, it is likely GREMLIN will require additional training for these regimes and additional inputs to be able to distinguish different regimes.

Finding a discrepancy between model performance estimates at the development stage versus at the deployment stage is common enough that the medical field distinguishes between retrospective accuracy versus prospective accuracy (*Caruana 2023*). Retrospective accuracy is the estimate that comes from splitting your data into training, validation, and testing datasets. The hope is that this split will provide a sufficient estimate of accuracy in the real world. On the other hand, once a model is deployed in the real world, the running estimate of accuracy based on the data samples seen after deployment is known as prospective accuracy. The prospective accuracy is considered to provide the best correspondence between correctness and accuracy. Thus, an approach in the medical field is deployment in a "soft" mode where predictions are evaluated but not acted upon for operational decision making. A similar approach is being used with GREMLIN, where a Full Disk version of GREMLIN has been deployed on NOAA's GeoCloud, and a select group of interested users have been invited to evaluate model performance for their analysis and forecasting tasks in a purely demonstration/prototype mode.

While the CONUS region covers only about 3% of the Earth, it does contain different precipitation regimes, and analysis of CONUS data over locations that were not included in the training dataset is a starting point for evaluating regime biases in GREMLIN. In this chapter, GREMLIN predictions are evaluated over all locations and seasons, and biases related to surface temperature are found (Section 3.3.1). The relationship between lightning and radar reflectivity is explored, and systematic differences related to moisture are found (Section 3.3.2). These results

point to the need for re-training GREMLIN with additional inputs and plans for this are discussed in Section 3.3.3. Section 3.4 outlines plans for evaluating Version-2 GREMLIN outside of the CONUS region.

3.2. Data and Methodology

The expanded CONUS dataset used for evaluation is described in Section 3.2.1. The environmental data used to quantify regimes is described in Section 3.2.2. New datasets for evaluating GREMLIN over the ocean are described in Section 3.3.3. A Full Disk dataset for re-training GREMLIN is described in Section 3.3.4.

3.2.1. GREMLIN CONUS3 Dataset Construction

The GREMLIN prototype was developed using the *CONUS1* dataset, which consisted of manually selected storm samples, chosen to maximize the diversity in location, time-of-day, and convective mode. However, that dataset was too small for training a convolutional neural network (CNN) without resorting to data augmentation. Thus, the Version-1 GREMLIN model was trained on the *CONUS2* dataset, which was constructed based on capturing samples with the most storm reports, as described in Chapter 1. However, the nature of the dataset construction led to certain regions of the United States not being sampled (e.g., Florida, New England, Western U.S.), and the dataset focused on just the warm season from April to July. To thoroughly validate GREMLIN, a *CONUS3* dataset was constructed that includes all locations over the contiguous United States and all times-of-year. The GOES and MRMS data were resampled to the HRRR mass grid (3 km Lambert conformal conic with 1799 pixels in longitude and 1059 pixels in latitude) as described in Chapter 1. The sampling rate of the data is every 15 minutes, which provides 35K samples per year. The year 2018 was excluded because of lower GLM data quality, and 2019 was excluded to keep *CONUS3* fully independent from *CONUS2*. Thus, *CONUS3*

covers the period 2020-2022, and there are a total of 100,178 samples.

While the MRMS composite reflectivity already has quality controlled applied, artifacts remain, especially over particular regions of the Pacific Northwest and Texas Gulf Coast. These problem areas were screened by printing sorted lists of the number of pixels with echoes exceeding 65 dBZ and 50 dBZ thresholds for each sample, and then inspecting images for the top samples. About 80 samples with areas (spatial extent) of strong echoes that are unphysically large were found and removed from the dataset. There are isolated pixels with unphysically strong echoes corresponding to wind power farms, but these samples were not removed because they made a very small contribution to the overall total number of pixels. The quality control procedure described above does not screen for artifacts arising from ground clutter when the radars are in clear-air mode, but the impact from that is addressed in the analysis.

3.2.2. Environmental Data

To examine how biases vary with meteorological regime, analysis fields from the High-Resolution Rapid Refresh (HRRR) model are used. These data are available each hour, and the nearest in time fields are used for each sample. The HRRR version switches from v3 to v4 on 2020-12-02.

3.2.3. Over Ocean Validation Data

To validate GREMLIN over the ocean, MRMS data in CONUS coastal zones can be used, as well as MRMS domains covering the Caribbean, Hawaii, and Alaska. For those areas of the ocean where MRMS data are not available, reflectivity measurements from the NASA Global Precipitation Measurement mission GPM Dual-frequency Precipitation Radar DPR (*Iguchi et al. 2010*, *NASA PPS 2021*) were used. Figure 3.1 illustrates the data coverage from these sources. To construct composite reflectivity from the DPR, we used the 2A Ku files, which each contain a

half-hour of data. The KuPR swath is 245 km wide, and measurements have a nominal spatial resolution of 5 km. Because DPR is on a high inclination satellite, the data coverage is significantly sparse, as compared to MRMS coverage. We accumulated the available data on the GOES Full Disk grids every 10 minutes (GOES Mode 6). DPR provides vertical profiles (along the slant view path) with 250 m vertical resolution. However, for this research, the vertical maximum (composite reflectivity) is the desired quantity. In addition to checking the quality control information for each scan and checking the precipitation flag for each cell to ensure meteorological echoes, the most important part of calculating composite reflectivity from DPR is using the surface clutter flag that identifies the lowest vertical bin which is free of surface clutter. Initial evaluation of DPR against MRMS in the coastal regions of CONUS yielded two major findings. First, we found that the DPR-derived composite reflectivity has an effective noise floor around 15-20 dBZ, consistent with the results of *Toyoshima et al. (2015)*. Thus, DPR should not be used to train the model for reflectivity below this value. Second, to get unbiased agreement with MRMS composite reflectivity, it is necessary to first vertically smooth the DPR vertical profiles to better match the coarser MRMS vertical resolution before calculating the composite reflectivity.

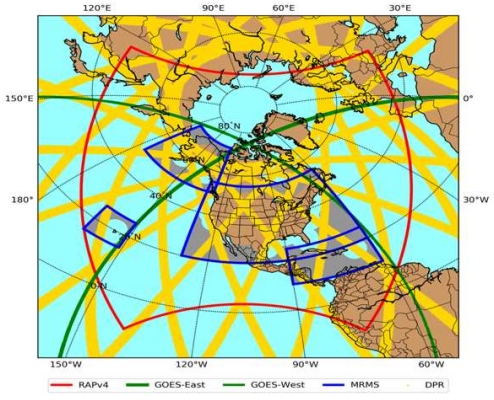

Figure 3.1. RAPv4 region (red), GOES-East/West Full Disk coverage out to a zenith angle of 86º (green thick/thin), MRMS coverage (blue, with no coverage in gray filled areas), and DPR swaths for one day (yellow). The DPR swath patterns precesses with coverage shifting about 8º in longitude each day.

3.2.4. GREMLIN Full Disk Dataset Construction

Preliminary testing has shown that the GREMLIN model trained on the 3 km HRRR grid can be applied to make predictions on the 2 km GOES Full Disk grid without significant obvious artifacts. However, validation using *CONUS3* indicates that for GREMLIN to generalize well to other regimes, additional environmental information will be needed. Moreover, results indicate that incorporating the regime dependence through simple post-hoc bias corrections is suboptimal. The results suggest that re-training GREMLIN to make use of the environmental information will be required. For both training and evaluation, it will be advantageous to have version of the GREMLIN dataset on the ABI Full Disk grids. Figure 3.2 compares the ABI Full Disk grid with the GLM Event grid, which shows that complete inputs are only available over a subset of the Full Disk, and data outside of the GLM field-of-regard will be masked as no data regions. For a truly global version of GREMLIN, other lightning datasets would be needed, such as the World Wide Lightning Location Network (WWLLN) or the Earth Networks Total Lightning Network (ENTLN). The GLM and MRMS datasets will be resampled to the 2 km Full Disk grid (5424 x 5424) and ABI channels at higher resolutions (0.5 and 1.0 km) will be downsampled to 2 km using the mean operator for the sake of reducing data volume. An alternative approach is to introduce the higher resolution ABI channels earlier in the network (*Lee et al. 2021*).

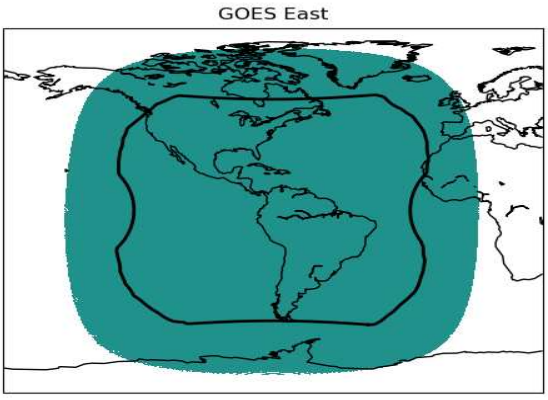

Figure 3.2. Comparison of the ABI Full Disk (blue fill) with the extent of the GLM field-of-regard (black line).

3.3. Results and Discussion

Section 3.3.1 describes validation of GREMLIN predictions, showing that the most significant biases are explained by a dependence on surface temperature. Section 3.3.2 explores the relationship between lightning and radar reflectivity, which shows a relationship has strong co-variability with the total atmospheric moisture. Finally, Section 3.3.3 describes plans to re-train and evaluate a Full Disk version of GREMLIN based on lessons learned herein.

3.3.1. GREMLIN Version-1 Validation

Figure 3.3 shows maps of root-mean-square-difference (RMSD) over the three-year period for each season. The RMSD is smallest in the summertime, which is consistent with the training period of GREMLIN. Over the Great Plains, summer RMSD is around 3-4 dBZ, while further east the RMSD is around 4-5 dBZ, which are in line with the values reported by *Hilburn et al. (2021a)* using the *CONUS2* testing dataset. It is interesting to note that Florida has larger errors, 5-6 dBZ, which happens to be a region that was not included in the *CONUS2* dataset. This suggests that the relationships between GOES ABI and GLM inputs with radar reflectivity are systematically different than those learned over the Great Plains and Upland South. The very small RMSD values over California and the West Coast reflect the negligible amount of radar echoes that occur during that season.

Figure 3.3 shows that RMSD increases in fall towards a maximum in winter and subsides in spring towards minimum values in summer months. In those seasons, the RMSD maps have a pronounced north-to-south gradient with larger errors further north. Examination of individual samples from each season confirms that the increase in RMSD is due to cold surfaces being mistaken as precipitating clouds since GREMLIN was not trained on synoptic-scale winter precipitation systems. Note that across the South, where cold temperatures and snow-covered

surfaces are less common, the RMSD remains around 3-5 dBZ year-round. Comparing these RMSD maps with maps of the bias (not shown) indicates that the increase in RMSD in cold seasons is driven primarily by an increase in bias, where GREMLIN overpredicts echo strength.

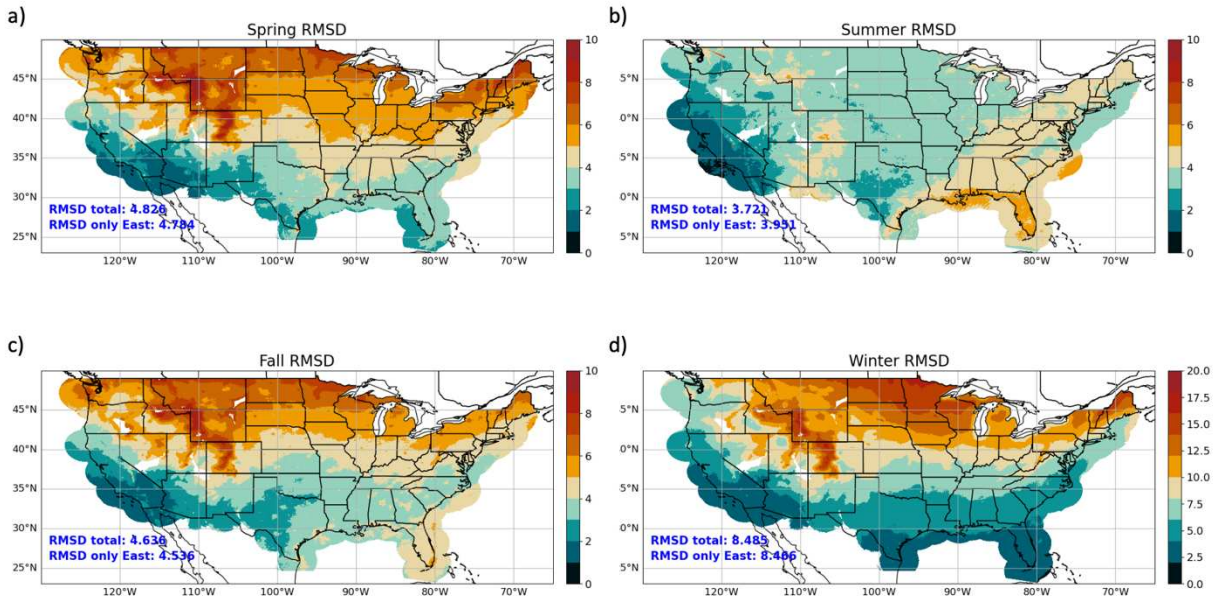

Figure 3.3. RMSE maps over CONUS for (a) spring (MAM), (b) summer (JJA), (c) fall (SON), and (d) winter (DJF). The "RMSE total" is calculated over all pixels, while "RMSE only east" is calculated for longitude greater than 105ºW. Note that the winter map has a different color scale.

Figure 3.4 shows the GREMLIN bias as a function of the surface temperature. There is a rapid increase in bias as the surface temperature decreases from 280 K to 250 K, with bias exceeding 20 dBZ when the surface temperature is colder than 250 K. Short of re-training GREMLIN with surface temperature as an input, two simple methods were tested for correcting this problem. The first method was to scale the inputs dynamically using surface temperature rather than a fixed value of 300 K, which is roughly the average surface temperature for the *CONUS2* training dataset. The second method was to apply the curve in Figure 3.4 as a post-hoc bias correction. The dynamical scaling approach had generally poor results, removing most echo in wintertime, which while reducing the bias, does not result in a useful product. Since GREMLIN was trained on data with fixed scaling, this approach is probably breaking the learned

relationships. The post-hoc bias correction gave better results, reducing wintertime false alarms greatly and bringing the RMSD in line with other seasons. However, examination of case studies revealed that since the correction reduces wintertime echoes in general, it also suffered from the problem of reducing the probability of detection more than desired. These results suggest that the issues GREMLIN has in winter cannot be removed with a simple correction, and that at a minimum, re-training will be necessary. *Lee and Hilburn (2023)* found that ABI derived products, such as the clear sky mask, are also useful in reducing false alarms in winter, but was not sufficient by itself. It remains an open question how much information content infrared-based techniques have for radar echoes in winter and note that there is generally very little lightning in winter.

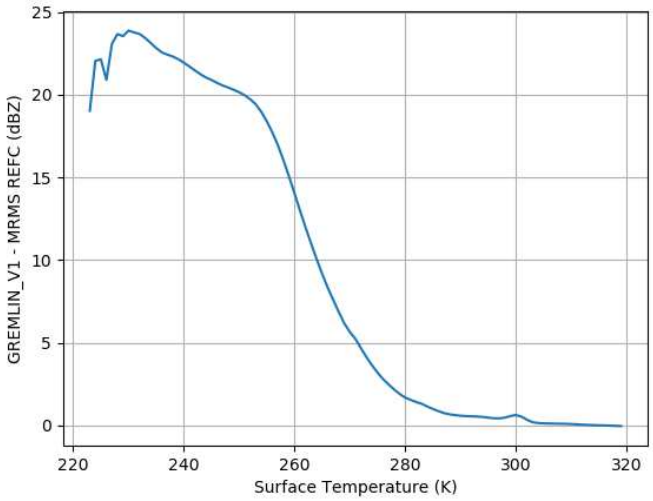

Figure 3.4. GREMLIN minus MRMS versus HRRR 2m temperature.

In summary, validation of GREMLIN over all locations across CONUS and over all times-of-year shows that to first order, errors in GREMLIN correlate with the surface temperature and thus exhibit a seasonal cycle. Larger errors were also found over Florida, which was not included in the training. Additional analysis in *Lee and Hilburn (2023)* examines differences in the inputs between Florida and the Great Plains, finding that enhanced lightning

activity over Florida relative to the rest of CONUS leads to higher false alarm rates for GREMLIN over Florida. The environmental factors involved in the relationship between lightning and radar reflectivity are explored in the next section.

3.3.2. Lightning and Radar Reflectivity Relationships

An estimate for the bulk relationship between lightning and radar reflectivity can be made by presupposing a simple dipole model for the charge structure in a thunderstorm (Figure 3.5). In such a model, there is a thunderstorm of height H, updraft W, and with negative (positive) space charges Q in the lower (upper) portions of the storm. Following *Price and Rind (1992)* and *Boccippio (2002)*, by Ohm's Law, the electrical power P is given by the product of the potential difference V and the current I:

$$P = V I \tag{1}$$

From Coulomb's Law, the potential due to a charge at distance H is

$$V \sim Q H^{-1} \tag{2}$$

From the definition of current density, the current is

$$I \sim (Q H^{-3}) W H^{2} \sim Q W H^{-1} \tag{3}$$

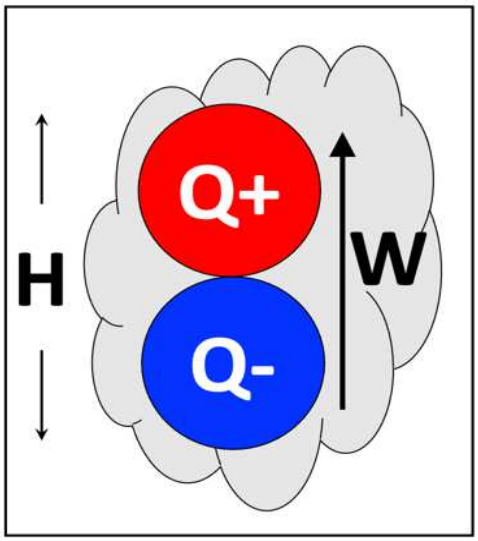

Figure 3.5. Dipole model of a thunderstorm where H is cloud depth, W is updraft, and Q are space charges.

Making the assumptions that: (a) the lightning flash rate L is proportional to the electrical power P, (b) the charge Q is proportional to the storm volume ~ H^3, and (c) over land $W \sim H$, gives

$$L \sim W H^4 \sim H^5 \sim W^5 \quad (4)$$

The assumptions behind this derivation are evaluated in *Boccippio (2002)* using TRMM Lightning Imaging Sensor (LIS) data, finding that the exponent on W may plausibly vary from about 2 to 10, where smaller exponents are associated with continental conditions and larger exponents for maritime conditions.

The definition of radar reflectivity (Z) in linear units ($mm^6\ m^{-3}$) gives

$$Z \sim D^6 \quad (5)$$

where D is the drop size. The terminal velocity V_T of a drop (*Gunn and Kinzer 1949*) is the balance between the drag force $F_D \sim v^2 D^2$ where v is the velocity, and the gravitational force $F_G \sim D^3$, so

$$V_T \sim D^{1/2} \quad (6)$$

Assuming $W \sim V_T$ gives

$$Z \sim W^{12} \quad (7)$$

Thus, the thunderstorm updraft is the key physical linkage between radar reflectivity and lightning, and expressing in logarithmic units:

$$\mathrm{Log}(Z) / \mathrm{Log}(L) = 12 / 5 \quad (8)$$

The *CONUS3* dataset provides GLM Group Extent Density (GED) with units of groups km^{-2} 5-min^{-1}. Consistent with GREMLIN scaling, GED < 0.1 are treated as "no lightning". To evaluate the relationship between lightning and radar reflectivity, the nonlinear transformation is applied

$$L\ (\mathrm{dBL}) = 10\ \mathrm{Log}_{10}\ (L/0.1) \quad (9)$$

Radar reflectivity is already expressed in logarithmic unit (dBZ) and note that $Z < 0$ dBZ is treated as "no echo".

Figure 3.6a gives a joint histogram of Z and L. The mean has $L = 11.4$ dBL and $Z = 28.1$ dBZ, which is almost on the 12/5 line. At lower L, the Z/L relationship often has much larger ratios. The relationship between Z and L defined by the mode of the joint distribution is decidedly nonlinear with jumps near $L = 10$ and 25 dBL. The slopes of segments between those jumps are flatter than the theoretical relationship. Most of the data (77%) falls between ratios of 12/2 to 12/10.

To quantify the spatial variability of the Z-L relationship, a quadratic model was fit to the data at each pixel:

$$Z = a + bL + cL^2 \tag{10}$$

where a is the intercept, b is the slope, and c is the concavity. A quadratic model was found to be superior to a linear model because the estimate of slope is less affected by radar ground clutter. More details are provided in Appendix C. Figure 3.6b presents a map of the slope coefficient. The slopes are in line with theory with values of 12/5 across the center of the country. There are smaller values to the south nearer the ocean, and larger values to the north. This pattern occurs because in the mean, there is more lightning to the south, but more echo to the north.

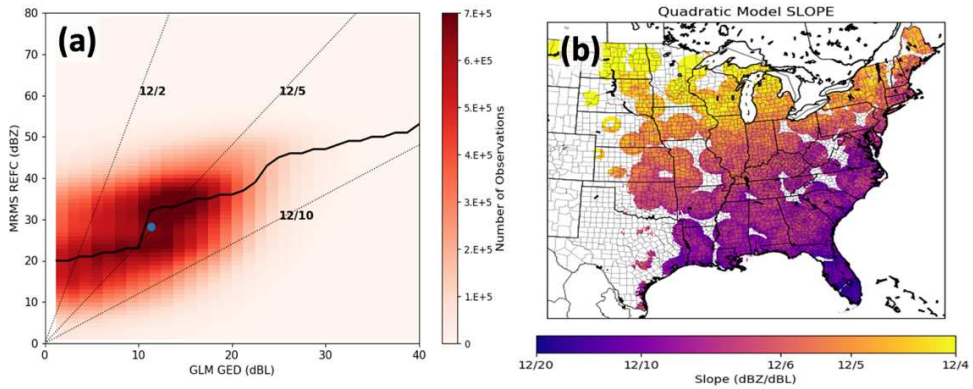

Figure 3.6. (a) Joint histogram of Z vs L, where Z is from MRMS. Heavy solid line is the mode, light dotted lines are theoretical ratios, and blue filled circle is the mean. **(b)** Map of Z-L slope.

Table 3.1 gives correlation coefficients between the *Z-L* slope and several relevant variables from HRRR. The Total Precipitable Water (TPW) has the highest correlation, followed by other moisture-related variables (specific humidity and dew point). This is consistent with Figure 3.6b, that shows the smallest slopes located over parts of CONUS nearest to the oceans. Surface temperature and the deep-layer wind shear also have strong correlations.

Table 3.1. Correlation between Z-L slope and various HRRR variables.

Variable	R^2
Total Precipitable Water (TPW)	0.762
2m Specific Humidity	0.755
2m Dew Point	0.733
Warm Cloud Depth (WCD = LCL – FRZ)	0.717
2m Temperature	0.697
Freezing Height (FRZ)	0.672
0-6 km Wind Shear	0.633
Convective Available Potential Energy (CAPE)	0.601
Warm Cloud Depth (WCD = CBH – FRZ)	0.433
Lifting Condensation Level (LCL)	0.067
Dew Point Depression	0.059
Cloud Base Height	0.041

The same type of analysis can now be applied to GREMLIN predictions of *Z* to assess how well GREMLIN regime variability matches reality. Figure 3.7a gives a joint histogram but using GREMLIN *Z* instead of MRMS. This shows GREMLIN has learned the 12/5 relationship in the data, but GREMLIN tends to have more probability in the range 12/2 to 12/5. GREMLIN also has a jump around $L = 25$ dBL, but the jump shown in MRMS at 10 dBL is shifted to lower values in GREMLIN. Figure 3.7b give a map of the slope coefficient for GREMLIN. The overall spatial pattern has some similarity to MRMS but note the different color scale from Fig. 3.6b. GREMLIN slope values are compressed with 12/5 for Florida, increasing to 12/3 over the

Northern Great Plains. Since GREMLIN does not have moisture as an input, this variation is more likely to reflect GLM detection efficiency, which decreases moving away from the subsatellite point.

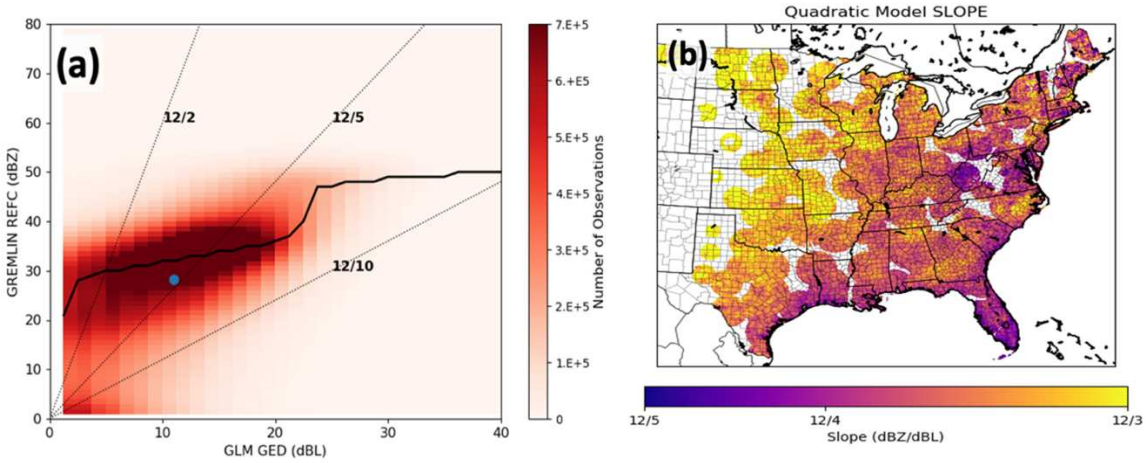

Figure 3.7. (a) Joint histogram for *Z* vs *L* using GREMLIN predicted *Z*. **(b)** Map of *Z*-*L* slope for GREMLIN.

These results showed that simple physical arguments yield *Z*-*L* scaling relationships that match the bulk properties (e.g., mean of the distribution) seen in radar and lightning datasets. The key physical linkage is the storm updraft. Spatial variability in the *Z*-*L* relationship primarily reflects the moisture environment and is well characterized by TPW. GREMLIN captures the 12/5 relationship, but the spatial variability is less than MRMS. TPW is not an input to GREMLIN, but these results suggest it should be to account for lightning regime. TPW information can come from numerical weather prediction or from satellite products (e.g., John Forsythe's global TPW product). This will be important as GREMLIN is extended to GOES Full Disk and then global domains. These results demonstrate that it is possible to interpret machine learning predictions to gain physical insights.

3.3.3. Training Version-2 GREMLIN

The results in the previous two sections highlight that GREMLIN will need information about temperature and moisture to properly account for different regimes. An open question is

whether additional environmental information will be needed for distinguishing regimes. For example, given the land/ocean contrast in lightning, should the surface type be a predictor, or should an estimate of the physical mechanism be used, such as using aerosol optical depth to quantify cloud condensation nuclei (*Stolz et al. 2017*)? There are certainly additional GOES inputs that should be included for maximum accuracy. *Hilburn et al. (2021c)* document that GLM lighting area improves GREMLIN predictions by helping distinguish small convective flashes from larger stratiform flashes (*Bruning and MacGorman 2013*). Information from visible and near infrared cloud particle size sensitive bands (e.g., 1.6 or 2.2 μm) is known to be beneficial for this application (*Hilburn 2020*). The Nighttime Microphysics RGB (*CIRA VISIT 2019*), which is popular with NWS forecasters, uses the reverse split window difference (12.4-10.4 μm), which is also useful during the day to distinguish thick versus thin portions of the thunderstorm anvil. The Severe Storms RGB (*EUMeTrain 2021*) has more structure in thunderstorm anvils than the popular Day Cloud Phase RGB (*CIRA VISIT 2020*) through its use of channel differences (water vapor 6.2 – 7.3 μm, infrared 3.9 – 10.8 μm, and near IR/visible 1.6-0.6 μm). Training the Version-2 GREMLIN model provides the opportunity to bring all these new channels into the model.

The other open question is regarding what changes to the GREMLIN model will be necessary for the best performance. For example, does GREMLIN need gradients of the environmental regime quantities, or is it just the value itself that matters? This would affect whether the environmental information comes into the network at the first input layer, or in some later input layer. There are also questions whether convolutional neural networks (CNNs) remain the best choice for image-to-image translation problems. There is emerging evidence that advanced architectures using attention and transformers may be better for decoding spatio-

temporal information (*Bansal et al. 2023*). Active research is examining whether the blurriness issue observed with CNNs can be improved through advanced architectures or loss functions. Note that while an interpretable version of the GREMLIN model exists (*Hilburn 2023*), machine learning is a more powerful and efficient tool for model development, and thus the interpretable model is expected to be used primarily for evaluating the nature of the regime information learned by the Version-2 GREMLIN model. Finally, the development of a new model provides the opportunity to include uncertainty estimates as an output (*Barnes et al. 2021*), which will be needed for the next generation of variational data assimilation the Rapid Refresh Forecast System (RRFS). Preliminary experiments with GREMLIN confirm that the *Barnes et al. (2021)* approach works well and produces realistic maps of uncertainty.

3.4. Summary, Conclusions, and Future Work

This chapter described the ongoing efforts to extend the original GREMLIN model to make trustworthy predictions over GOES Full Disk, and eventually on a nearly global domain using all geostationary satellites. Validation of GREMLIN using the *CONUS3* dataset found strong seasonal biases related to surface temperature, and the amount of usable information content in wintertime remains an open question. Lightning is an important predictor in GREMLIN, and analysis shows that the co-variability of lightning and radar reflectivity depends on the moisture regime. These results point to the need to include additional predictors in GREMLIN to account for regime. The accuracy of the Version-2 GREMLIN model will be quantified using available MRMS data over the Caribbean, Hawaii, and Alaska sectors, and over the open ocean using satellite-based radar observations from GPM DPR. The improvements to GREMLIN will be implemented in the processing on NOAA's GeoCloud, and feedback from select users at the National Weather Service, Ocean Prediction Center, National Hurricane

Center, Aviation Weather Center, and Department of Defense will be collected. This approach will allow quantifying both GREMLIN's accuracy and its fitness for a wide variety of end-user applications.

Future work on GREMLIN, supported by NASA, will assimilate GREMLIN into the NASA-Unified Weather Research and Forecasting (NU-WRF) model. Sensitivity analysis will be used to quantify the impact of latent heating uncertainties on short-term forecast accuracy. This will help answer the tough question of "how good is good enough" for estimating latent heating to assimilate into models. Sensitivity analysis will be used to evaluate hypotheses for why forecasts using synthetic radar data can outperform actual radar data. The hypotheses under consideration include: (a) GREMLIN has spatially smoother fields, (b) GREMLIN has stronger heating estimates, (c) differences in observed versus the simulated vertical profiles used by GREMLIN, and (d) GOES lead time over radar due to GOES observing cloud top features. GREMLIN vertical profiles will be compared against NASA GPM latent heating products to assign uncertainties to the *Lee et al. (2022)* vertical profile model. The focus is on GOES satellites, but note that additional future work supported by Department of Defense will extend Full Disk GREMLIN to other sensors in the ring of geostationary satellites.

Looking longer-term, private companies like Tomorrow.io are launching their own constellations of spaceborne radars. That will provide new datasets for evaluating and improving GREMLIN over the ocean. A constellation with a large number of satellites might make synthetic radar from geostationary sensors unnecessary, however, the rapid temporal refresh of contemporary geostationary satellite sensors is extremely valuable, and synthetic radar may still play a role in merging spaceborne radar into gap-free global products.

REFERENCES

Adelson, E. H., Anderson, C. H., Bergen, J. R., Burt, P. J., and Ogden, J. M., 1984: Pyramid methods in image processing. *RCA Engineer*, 33–41.

Agrawal, S., L. Barrington, C. Bromberg, J. Burge, C. Gazen, and J. Hickey, 2019: Machine learning for precipitation nowcasting from radar images. *arXiv.* https://arxiv.org/abs/1912.12132.

Araujo, A., Norris, W., & Sim, J., 2019: Computing Receptive Fields of Convolutional Neural Networks. *Distill*, 4(11), e21, https://distill.pub/2019/computing-receptive-fields/.

Arkin, P. A., and B. N. Meisner, 1987: The relationship between large-scale convective rainfall and cold cloud over the Western Hemisphere during 1982-84. *Mon. Wea. Rev.*, **115**, 51-74.

Ayzel, G., T. Scheffer, M. Heistermann, 2020: RainNet v1.0: a convolutional neural network for radar-based precipitation nowcasing. *Geosci. Model Dev.*, **13**, 2631-2644, https://doi.org/10.5194/gmd-13-2631-2020.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W., 2015: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS One*, **10**, https://doi.org/10.1371/journal.pone.0130140.

Back, A., S. Weygandt, C. Alexander, S. Benjamin, M. Hu, G. Ge, E. James, A. Kliewer, J. R. Mecikalski, D. Dowell, E. C. Bruning, K. Hilburn, and A. Sebok, 2021: Convection-indicating GOES-R products assimilating in the experimental UFS Rapid Refresh System. *AGU Fall Meeting*.

Back, A., A. Kliewer, J. R. Mecikalski, K. Hilburn, Y. Lee, E. Sebok, D. Dowell, E. C. Bruning,

M. Xue, R. Kong, S. Benjamin, E. P. James, C. R. Alexander, G. Ge, K. Pederson, and S. Weygandt, 2022: Novel Convection-Indicating Satellite Products Assimilated in Experimental Rapid Refresh Systems. *AMS Annual Meeting*, 26IOAS.

Bang, S. D., and E. J. Zipser, 2015: Differences in size spectra of electrified storms over land and ocean. *Geophys. Res. Lett.*, **42**, 6844-6851, https://doi.org/10.1002/2015GL065264.

Bansal, A. S., Y. Lee, K. Hilburn, and I. Ebert-Uphoff, 2023: Leveraging spatiotemporal information in meteorological image sequences: From feature engineering to attention-based neural networks. *Environmental Data Science*, 2: e31, 1-27, https://doi.org/10.1017/eds.2023.26.

Barnes, E. A., R. J. Barnes, and N. Gordillo, 2021: Adding uncertainty to neural network regression tasks in the geosciences. *arXiv*, https://arxiv.org/abs/2109.07250.

Bedka, K. M., R. Dworak, J. Brunner, and W. Feltz, 2012: Validation of satellite-based objective overshooting cloud-top detection methods using CloudSat cloud profiling radar observations. *J. Appl. Meteor. Climatol.*, **51**, 1811-1822, https://doi.org/10.1175/JAMC-D-11-0131.1.

Benjamin, S. G., 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, H. Lin, S. E. Peckham, T. L. Smith, W. R. Moninger, and J. S. Kenyon, 2016: A North American hourly assimilation and model forecast cycle: The Rapid Refresh. *Mon. Wea. Rev.*, **144**, 1669-1694, https://doi.org/10.1175/MWR-D-15-0242.1.

Blau, Y., and T. Michaeli, 2018: The perception-distortion tradeoff. *Conference on Computer Vision and Pattern Recognition*. 6228-6237.

Boccippio, D. J., 2002: Lightning scaling relations revisited. *J. Atmos. Sci.*, **59**, 1086-1104.

Bruning, E. C., and D. R. MacGorman, 2013: Theory and observations of controls on lightning flash size spectra. *J. Atmos. Sci.*, **70**, 4012-4029, https://doi.org/10.1175/JAS-D-12-0289.1.

Burt, P. J. and Adelson, E. H., 1983: The Laplacian pyramid as a compact image code. *IEEE Trans. Communications*, COM-31(4), 532–540.

Caruana, R., 2023: High accuracy is not enough – Not everything that is important can be measured. *AI2ES Webinar*, February 16.

Cintineo, J. L., M. J. Pavolonis, J. M. Sieglaff, D. T. Lindsey, L. Cronce, J. Gerth, B. Rodenkirch, J. Brunner, and C. Gravelle, 2018: The NOAA/CIMSS ProbSevere Model: Incorporation of total lightning and validation. *Wea. Forecasting*, **33**, 331-345, https://doi.org/10.1175/WAF-D-17-0099.1.

CIRA VISIT, 2019: Nighttime Microphysics RGB Quick Guide. https://rammb.cira.colostate.edu/training/visit/quick_guides/QuickGuide_GOESR_NtMicroRGB_Final_20191206.pdf.

CIRA VISIT, 2020: Day Cloud Phase Distinction RGB Quick Guide. https://rammb.cira.colostate.edu/training/visit/quick_guides/QuickGuide_DayCloudPhaseDistinction_final_v2.pdf.

Doshi-Velez, F., and B. Kim, 2017: Towards a rigorous science of interpretable machine learning. *arXiv*, https://arxiv.org/abs/1702.08608.

Došilović, F. K., M. Brčić, and N. Hlupić, 2018: Explainable Artificial Intelligence: A Survey. *41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. May 21-25, Opatija, Croatia.

Dowell, D. C., Alexander, C. R., James, E. P., Weygandt, S. S., Benjamin, S. G., Manikin, G. S.,

Blake, B. T., Brown, J. M., Olson, J. B., Hu, M., Smirnova, T. G., Ladwig, T., Kenyon, J. S., Ahmadov, R., Turner, D. D., Duda, J. D., and Alcott, T. I., 2022: The High-Resolution Rapid Refresh (HRRR): An Hourly Updating Convection-Allowing Forecast Model. Part I: Motivation and System Description. *Wea. Forecasting*, **37**, 1371-1395, https://doi.org/10.1175/WAF-D-21-0151.1.

Du, M., N. Liu, and X. Hu, 2020: Techniques for interpretable machine learning. *Communications of the ACM*. **63**, 68-77, https://doi.org/10.1145/3359786.

Ebert-Uphoff, I., and K. Hilburn, 2020: Evaluation, tuning, and interpretation of neural networks for working with images in meteorological applications. *Bull. Amer. Meteor. Soc.,* **101**, E2149-E2170, https://doi.org/10.1175/BAMS-D-20-0097.1.

Ebert-Uphoff, I., R. Lagerquist, K. Hilburn, Y. Lee, K. Haynes, J. Stock, C. Kumler, and J. Q. Stewart, 2021: CIRA guide to custom loss functions for neural networks in environmental sciences – Version 1. *arXiv*, https://arxiv.org/abs/2106.09757.

Elmer, N. J., E. Berndt, and G. J. Jedlovec, 2016: Limb correction of MODIS and VIIRS infrared channels for the improved interpretation of RGB composites. J. *Atmos. Oceanic Technol.*, **33**, 1073-1087, https://doi.org/10.1175/JTECH-D-15-0245.1.

EUMeTrain, 2021: SEVIRI Severe Storms RGB Quick Guide. https://www.eumetrain.org/sites/default/files/2021-05/SevereStormsRGB.pdf

Flora, M. L., C. K. Potvin, A. McGovern, and S. Handler, 2022: Comparing explanation methods for traditional machine learning models Part 1: An overview of current methods and quantifying their disagreement. *arXiv*, https://arxiv.org/abs/2211.08943.

Fuchs, B. R., S. A. Rutledge, B. Dolan, L. D. Carey, and C. Schultz, 2018: Microphysical and kinematic processes associated with anomalous charge structures in isolated convection.

J. Geophys. Res. Atmos., **123**, https://doi.org/10.1029/2017JD027540.

Geer, A., and Coauthors, 2018: All-sky satellite data assimilation at operational weather forecasting centres. *Quart. J. Roy. Meteor. Soc.*, **144**, 1192-1217, https://doi.org/10.1002/qj.3202.

Gilpin, L. H., D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, 2018: Explaining explanations: An overview of interpetability of machine learning. IEEE Fifth International Conference on Data Science and Advanced Analytics. https://doi.org/10.48550/arXiv.1806.00069.

Goodman, S., D. Mach, W. Koshak, and R. Blakeslee, 2012: GLM Lightning Cluster-Filter Algorithm. Version 3.0. *NOAA NESDIS STAR*, 73 pp, https://www.star.nesdis.noaa.gov/goesr/documents/ATBDs/Baseline/ATBD_GOES-R_GLM_v3.0_Jul2012.pdf.

Goodman, S. J., R. J. Blakesless, W. J. Koshak, D. Mach, J. Bailey, D. Buechler, L. Carey, C. Schultz, M. Bateman, E. McCaul Jr., and G. Stano, 2013: The GOES-R Geostationary Lightning Mapper (GLM). *Atmos. Res.*, **125-126**, 33-49, https://doi.org/10.1016/j.atmosres.2013.01.006.

Gonzalez, R. C., and R. E. Woods, 2002: Digital Image Processing, 2nd Ed., Prentice-Hall, 793 pp.

Guilloteau, C. and Foufoula-Georgiou, E., 2020: Beyond the pixel: Using patterns and multiscale spatial information to improve the retrieval of precipitation from spaceborne passive microwave imagers. *J. Atmos. Oceanic Technol.*, **37**, 1571–1591, https://doi.org/10.1175/JTECH-D-19-0067.1.

Gunn, R., and G. D. Kinzer, 1949: The terminal velocity of fall for water droplets in stagnant air.

J. Meteor., **6**, 243-248.

Gustafsson, N., and Coauthors, 2018: Survey of data assimilation methods for convective-scale numerical weather prediction at operational centres. *Quart. J. Roy. Meteor. Soc.*, **144,** 1218-1256, https://doi.org/10.1002/qj.3179.

Harris Corporation, 2016: Product Definition and User's Guide (PUG), Volume 5: Level 2+ Products. DCN-7035538, Revision-E, 699 pp, https://www.goes-r.gov/products/docs/PUG-L2+-vol5.pdf.

Haynes, K., Slocum, C., Knaff, J., Musgrave, K., and Ebert-Uphoff, I., 2022: Aiding Tropical Cyclone Forecasting by Simulating 89-GHz Imagery from Operational Geostationary Satellites. American Meteorological Society 35th Conference on Hurricanes and Tropical Meteorology, New Orleans, LA, 9-13 May (virtual).

Hilburn, K., 2020: Using solar reflective channels in GREMLIN. *CIRA ML Core Meeting*, April 22.

Hilburn, K. A., I. Ebert-Uphoff, and S. D. Miller, 2021a: Development and interpretation of a neural network-based synthetic radar reflectivity estimator using GOES-R satellite observations. *J. Appl. Meteor. Climatol.*, **60**, 3-21, https://doi.org/10.1175/JAMC-D-20-0084.1.

Hilburn, K., Y. Lee, and I. Ebert-Uphoff, 2021b: GREMLIN: GOES Radar Estimation via Machine Learning to Inform NWP. Fall AGU Meeting, A047.

Hilburn, K., Y. Lee, and I. Ebert-Uphoff, 2021c: Improving GREMLIN: A case study in AI application development. *3rd NOAA Workshop on Leveraging AI in Environmental Sciences*, September 15.

Hilburn, K. A., 2022: GREMLIN CONUS2 Dataset. http://dx.doi.org/10.25675/10217/235392.

Hilburn, K., A., 2023: Understanding Spatial Context in Convolutional Neural Networks using Explainable Methods: Application to Interpretable GREMLIN. *Artif. Intell. Earth Syst.*, https://doi.org/10.1175/AIES-D-22-0093.1.

Hinton, G., O. Vinyals, and J. Dean, 2015: Distilling the knowledge in a neural network. *arXiv*, https://arxiv.org/abs/1503.0253.

Honda, T., T. Miyoshi, G.-Y. Lien, S. Nishizawa, R. Yoshida, S. A. Adachi, K. Terasaki, K. Okamoto, H. Tomita, and K. Bessho, 2018a: Assimilating all-sky Himawari-8 infrared radiances: A case of Typhoon Soudelor (2015). *Mon. Wea. Rev.*, **146**, 213-229, https://doi.org/10.1175/MWR-D-16-0357.1.

Honda, T., S. Kotsuki, G.-Y. Lien, Y. Maejima, K. Okamoto, and T. Miyoshi, 2018b: Assimilation of Himawari-8 all-sky radiances every 10 minutes: Impact on precipitation and flood risk prediction. *J. Geophys. Res. Atmos.*, **123**, 965-976, https://doi.org/10.1002/2017JD027096.

Iguchi, T., S. Seto, R. Meneghini, N. Yoshida, J. Awaka, T. Kubota, 2010: GPM/DPR Level-2 Algorithm Theoretical Basis Document. December 2010, 72 pp.

James, E. P., Alexander, C. R., Dowell, D. C., Weygandt, S. S., Benjamin, S. G., Manikin, G. S., Brown, J. M., Olson, J. B., Hu, M., Smirnova, T. G., Ladwig, T., Kenyon, J. S., and Turner, D. D., 2022: The High-Resolution Rapid Refresh (HRRR): An Hourly Updating Convection-Allowing Forecast Model. Part II: Forecast Performance. *Wea. Forecasting*, **37**, 1397-1417, https://doi.org/10.1175/WAF-D-21-0130.1.

Jones, T. A., D. Stensrud, L. Wicker, P. Minnis, and R. Palikonda, 2015: Simultaneous radar and satellite data storm-scale assimilation using an ensemble Kalman filter approach for 24 May 2011. *Mon. Wea. Rev.*, **143**, 165-194, https://doi.org/10.1175/MWR-D-14-00180.1.

Jones, T. A., P. Skinner, N. Yussouf, K. Knopfmeier, A. Reinhart, X. Wang, K. Bedka, W. Smith, Jr., and R. Palikonda, 2020: Assimilation of GOES-16 radiances and retrievals into the Warn-on-Forecast system. *Mon. Wea. Rev.*, **148,** 1829-1859, https://doi.org/10.1175/MWR-D-19-0379.1.

Kong, R., M. Xue, A. O. Fierro, Y. Jung, C. Liu, E. R. Mansell, and D. R. MacGorman, 2020: Assimilation of GOES-R Geostationary Lightning Mapper flash extent density data in GSI EnKF for the analysis and short-term forecast of a mesoscale convective system. *Mon. Wea. Rev.*, **148**, 2111-2133, https://doi.org/10.1175/MWR-D-19-0192.1.

Kurino, T., 1997: A satellite infrared technique for estimating "deep/shallow" precipitation. *Adv. Space Res.*, **19**, 511-514.

Lapuschkin, S., S. Wäldchen, A. Binder, G. Montavon, W. Samek, and K.-R. Müller, 2019: Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications*, **10**, 1-8, https://doi.org/10.1038/s41467-019-08987-4.

Lee, Y., C. D. Kummerow, and I. Ebert-Uphoff, 2021: Applying machine learning methods to detect convection using Geostationary Operational Environmental Satellite-16 (GOES-16) advanced baseline imager (ABI) data. *Atmos. Meas. Tech.*, **14**, 2699-2716, https://doi.org/10.5194/amt-14-2699-2021.

Lee, Y., C. D. Kummerow, and M. Zupanski, 2022: Latent heating profiles from GOES-16 and its impacts on precipitation forecasts. *Atmos. Meas. Tech.*, **15**, 7119-7136, https://doi.org/10.5194/amt-15-7119-2022.

Lee, Y., and K. Hilburn, 2023: Validating GOES radar estimation via machine learning to inform NWP (GREMLIN) product over CONUS. *J. Appl. Meteor. Climatol.*, submitted.

Liu, C., and E. Zipser, 2013: Regional variation of morphology of organized convection in the

tropics and subtropics. *J. Geophys. Res. Atmos.*, **118**, 453-466, https://doi.org/10.1029/2012JD018409.

Lin, J., S. S. Weygandt, S. G. Benjamin, and M. Hu, 2017: Satellite radiance data assimilation within the hourly updated Rapid Refresh. *Wea. Forecasting*, **32**, 1273-1287, https://doi.org/10.1175/WAF-D-16-0215.1.

Line, W. E., T. J. Schmit, D. T. Lindsey, and S. J. Goodman, 2016: Use of geostationary super rapid scan satellite imagery by the Storm Prediction Center. *Wea. Forecasting*, **31**, 483-494, https://doi.org/10.1175/WAF-D-15-0135.1.

Lundberg, S. M., and S.-I. Lee, 2017: A unified approach to interpreting model predictions. 31st Conference on Neural Information Processing Systems.

Luo, W., Li, Y., Urtasun, R., & Zemel, R., 2016: Understanding the effective receptive field in deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (pp. 4898-4906).

Mamalakis, A., I. Ebert-Uphoff, and E. A. Barnes, 2022: Neural network attribution methods for problems in geoscience: A novel synthetic benchmark dataset. *Envir. Data Science*, **1**, E8. https://doi.org/10.1017/eds.2022.7.

Marchand, M., K. Hilburn, and S. D. Miller, 2019: Geostationary Lightning Mapper and Earth Networks lightning detection over the Contiguous United States and dependence on flash characteristics. *J. Geophys. Res. Atmos.*, **124**, 11552-11567, https://doi.org/10.1029/2019JD031039.

McGovern, A., R. Lagerquist, D. J. Gagne II, G. E. Jergensen, K. L. Elmore, C. R. Homeyer, and T. Smith, 2019: Making the black box more transparent: Understanding the physical implications of machine learning. *Bull. Amer. Meteor. Soc.*, **100**, 2175-2199,

https://doi.org/10.1175/BAMS-D-18-0195.1.

McGovern, A., I. Ebert-Uphoff, D. J. Gagne II, and A. Bostrom, 2022: Why we need to focus on developing ethical, responsible, and trustworthy artificial intelligence approaches for environmental science. *Environmental Data Science*, **1**, E6 1-15, https://doi.org/10.1017/eds.2022.5.

Meng, F., T. Song, and D. Xu, 2022: Simulating tropical cyclone passive microwave rainfall imagery using infrared imagery via generative adversarial networks. *IEEE Geosci. Rem. Sens. Letts.*, **19**, https://doi.org/10.1109/LGRS.2022.3152847.

Miller, S. D., M. A. Rogers, J. M. Haynes, M. Sengupta, and A. K. Heidinger, 2018: Short-term solar irradiance forecasting via satellite/model coupling. *Solar Energy*, **168**, 102-117, https://doi.org/10.1016/j.solener.2017.11.049.

Miller, S. D., D. T. Lindsey, C. J. Seaman, and J. E. Solbrig, 2020: GeoColor: A blending technique for satellite imagery. *J. Atmos. Oceanic Technol.*, **37**, 429-448, https://doi.org/10.1175/JTECH-D-19-0134.1.

Miller, T., 2019: Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, **267**, 1-38, https://doi.org/10.1016/j.artint.2018.07.007.

Molnar, C., G. König, J. Herbinger, T. Freiesleben, S. Dandl, C. A. Scholbeck, G. Casalicchio, M. Grosse-Wentrup, and B. Bischl, 2022: General pitfalls of model-agnostic interpretation methods for mmachine learning models. Chapter in "xxAI – Beyond Explainable AI". Eds. A. Holzinger, R. Goebel, R. Fong, T. Moon, K.-R. Müller, and W. Samek, Springer, 39-68, https://doi.org/10.1007/978-3-031-04083-2_4.

Montavon, G., Samek, W., & Müller, K. R., 2018: Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, **73**, 1-15.

https://doi.org/10.1016/j.dsp.2017.10.011.

Murdoch, W. J., C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu, 2019: Definitions, methods, and applications in interpretable machine learning. *Proc. Nat. Acad. Sci.*, **116**, 22071-22080, https://doi.org/10.1073/pnas.1900654116.

Nag, A., and K. L. Cummins, 2017: Negative first stoke leader characteristics in cloud-to-ground lightning over land and ocean. *Geophys. Res. Lett.*, **44**, 1973-1980, https://doi.org/10.1002/2016GL072270.

NASA Precipitation Processing System (PPS), 2021: File Specification for GPM Products, Version 7.04 TKIO 3.97.18, 18-August-2021, 2515 pp, https://gpm.nasa.gov/resources/documents/file-specification-gpm-products.

NOAA NESDIS, 1998: Earth Location User's Guide (ELUG), Revision 1. DRL 504-11, NOAA/OSD3-1998-015R1UD0, 99 pp.

Okamoto, K., Y. Sawada, and M. Kunii, 2019: Comparison of assimilating all-sky and clear-sky infrared radiances from Himawari-8 in a mesoscale system. *Quart. J. Roy. Meteor. Soc.*, **145**, 745-766, https://doi.org/10.1002/qj.3463.

Olah, C., Mordvintsev, A. and Schubert, L., 2017: Feature visualization. *Distill*, *2*(11), p.e7. https://doi.org/10.23915/distill.00007.

Otkin, J. A., and R. Potthast, 2019: Assimilation of all-sky SEVIRI infrared brightness temperatures in a regional-scale ensemble data assimilation system. *Mon. Wea. Rev.*, **147**, 4481-4509, https://doi.org/10.1175/MWR-D-19-0133.1.

Papoulis, A., and S. U. Pillai, 2002: Probability, Random Variables, and Stochastic Processes, 4th Ed., McGraw Hill, 852 pp.

Peterson, M., W. Deierling, C. Liu, D. Mach, and C. Kalb, 2017: The properties of optical

lightning flashes and the clouds they illuminate. *J. Geophys. Res. Atmos.*, **122**, 423-442, https://doi.org/10.1002/2016JD025312.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, 1992: Numerical recipes in Fortran 77: The Art of Scientific Computing, 2nd Ed., Cambridge University Press.

Price, C., and D. Rind, 1992: A simple lightning parameterization for calculating global lightning distributions. *J. Geophys. Res.*, **97**, D9, 9919-9933.

Ribeiro, M. T., S. Singh, and C. Guestrin, 2016: "Why should I trust you?" Explaining the predictions of any classifier. *arXiv*, https://arxiv.org/abs/1602.04938.

Rocque, M., 2023: Influence of Terrain on the Characteristics and Life Cycle of Convection Observed in Subtropical South America. *Ph.D. Dissertation*. CSU Atmospheric Science Department.

Roebber, P., 2009: Visualizing multiple measures of forecast quality. *Wea. Forecasting*, **24**, 601-608, https://doi.org/10.1175/2008WAF2222159.1.

Ronneberger, O., P. Fischer, and T. Brox, 2015: U-Net: Convolutional networks for biomedical image segmentation. *MICCAI*, 234-241, https://arxiv.org/abs/1505.04597.

Rudin, C., 2019: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, **1**, 206-215, https://doi.org/10.1038/s42256-019-0048-x.

Rudin, C., C. Chen, Z. Chen, H. Huang, L. Semenova, C. Zhong, 2022: Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistical Surveys*, **16**, 1-85, https://doi.org/10.1214/21-SS133.

Rutledge, S. A., K. Hilburn, A. Clayton, B. Fuchs, and S. D. Miller, 2020: Evaluating Geostationary Lightning Mapper flash rates within intense convective storms. *J.*

Geophys. Res. Atmos., **125**, e2020JD032827, https://doi.org/10.1029/2020JD032827.

Samsi, S., C. J. Mattioli, and M. S. Veillette, 2019: Distributed deep learning for precipitation nowcasting. *IEEE High Performance Extreme Computing Conference (HPEC)*.

Sawada, Y., K. Okamoto, M. Kunii, and T. Miyoshi, 2019: Assimilating every-10-minute Himawari-8 infrared radiance to improve convective predictability. *J. Geophys. Res. Atmos.*, **124**, 2546-2561, https://doi.org/10.1029/2018JD029643.

Schmertz, J., S. A. Tjemkes, M. Gube, and L. van de Berg, 1997: Monitoring deep convection and convective overshooting with Meteosat. *Adv. Space Res.*, **19**, 433-441.

Schmit, T., M. Gunshor, G. Fu, T. Rink, K. Bah, and W. Wolf, 2012: GOES-R Advanced Baseline Imager (ABI) Algorithm Theoretical Basis Document for Cloud and Moisture Imagery Product (CMIP). Version 3.0. *NOAA NESDIS STAR*, 63 pp, https://www.star.nesdis.noaa.gov/goesr/docs/ATBD/Imagery.pdf.

Schmit, T. J., P. Griffith, M. M. Gunshor, J. M. Daniels, S. J. Goodman, W. J. Lebair, 2017: A closer look at the ABI on the GOES-R series. *Bull. Amer. Meteor. Soc.*, **98**, 681-698, https://doi.org/10.1175/BAMS-D-15-00230.1.

Schultz, C. J., W. A. Petersen, and L. D. Carey, 2009: Preliminary development and evaluation of lightning jump algorithms for the real-time detection of severe weather. *J. Appl. Meteor. Climatol.*, **48**, 2543-2563, https://doi.org/10.1175/2009JAMC2237.1.

Schultz, C. J., L. D. Carey, E. V. Schultz, and R. J. Blakeslee, 2015: Insight into the kinematic and microphysical processes that control lightning jumps. *Wea. Forecasting*, **30**, 1591-1621, https://doi.org/10.1175/WAF-D-14-00147.1.

Shepherd, T. G., 2019: Storyline approach to the construction of regional climate change information. *Proc. R. Soc. A*, **475**, 1-16, https://doi.org/10.1098/rspa.2019.0013.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., & Wattenberg, M.. 2017: Smoothgrad: removing noise by adding noise. *arXiv*, https://arxiv.org/abs/1706.03825.

Smith, T. M., V. Lakshmanan, G. J. Stumpf, K. L. Ortega, K. Hondl, K. Cooper, K. M. Calhoun, D. M. Kingfield, K. L. Manross, R. Toomey, and J. Brogden, 2016: Multi-radar multi-sensor (MRMS) severe weather and aviation products: Initial operating capabilities. *Bull. Amer. Meteor. Soc.*, **97**, 1617-1630, https://doi.org/10.1175/BAMS-D-14-00173.1.

Snyder, J. P., 1987: Map projections – A working manual. *U.S. Geological Survey Profession Paper 1395*. 397 pp, https://doi.org/10.3133/pp1395.

Stengel, K., A. Glaws, D. Hettinger, and R. N. King, 2020: Adversarial super-resolution of climatological wind and solar data. *Proc. Nat. Acad. Sci.*, 11 pp, https://doi.org/10.1073/pnas.1918964117.

Stolz, D. C., S. A. Rutledge, J. R. Pierce, and S. C. van den Heever, 2017: A global lightning parameterization based on statistical relationships among environmental factors, aerosols, and convective clouds in the TRMM climatology. J. *Geophs. Res. Atmos.*, **122**, 7461-7492, https://doi.org/10.1002/2016JD026220.

Su, A., H. Li, L. Cui, and Y. Chen, 2020: A convection nowcasting method based on machine learning. *Advances in Meteorology*, **2020**, https://doi.org/10.1155/2020/5124274.

Svaldi, A., 2017: Hailstorm that hammered west metro Denver May 8 is costliest ever for Colorado. *Denver Post*. Published 23-May-2017.

Toms, B. A., Barnes, E. A., & Ebert-Uphoff, I., 2019: Physically Interpretable Neural Networks for the Geosciences: Applications to Earth System Variability. *arXiv*, https://arxiv.org/abs/1912.01752.

Toyoshima, K., H. Masunaga, and F. A. Furuzawa, 2015: Early evaluation of Ku- and Ka-band

sensitivities for the Global Precipitation Measurement (GPM) Dual-frequency Precipitation Radar (DPR). *SOLA*, **11**, 14-17, https://doi.org/10.2151/sola.2015-004.

Van der Maaten, L., and G. Hinton, 2008: Visualizing data using t-SNE. *J. Machine Learning Research*, **9**, 2579-2605.

Veillette, M. S., E. P. Hassey, C. J. Mattioli, H. Iskenderian, and P. M. Lamey, 2018: Creating synthetic radar imagery using convolutional neural networks. *J. Atmos. Oceanic Technol.*, **35**, 2323-2338, https://doi.org/10.1175/JTECH-D-18-0010.1.

Vicente, G. A., J. C. Davenport, and R. A. Scofield, 2002: The role of orographic and parallax corrections on real time high resolution satellite rainfall rate distribution. *International Journal of Remote Sensing*, **23**, 221-230, https://doi.org/10.1080/01431160010006935.

Walther, A., W. Straka, and A. K. Heidinger, 2013: ABI algorithm theoretical basis document for daytime cloud optical and microphysical properties (DCOMP). Version 3.0, 11-June-2013, 66 pp, https://www.star.nesdis.noaa.gov/goesr/documents/ATBDs/Baseline/ATBD_GOES-R_Cloud_DCOMP_v3.0_Jun2013.pdf.

Weygandt, S. S., Benjamin, S. G., Hu, M., Alexander, C. R., Smirnova, T. G., and James, E. P., 2022: Radar Reflectivity–Based Model Initialization Using Specified Latent Heating (Radar-LHI) within a Diabatic Digital Filter or Pre-Forecast Integration. *Wea. Forecasting*, **37**, 1419-1434, https://doi.org/10.1175/WAF-D-21-0142.1.

Wilks, D. S., 2006: Statistical methods in the atmospheric sciences, 2nd Ed., Academic Press, 627 pp.

Williams, E., and S. Stanfill, 2002: The physical origin of the land-ocean contrast in lightning activity. *C. R. Physique 3*, 1277-1292.

Williams, E., Mushtak, V., Rosenfeld, D., Goodman, S., and Boccippio, D., 2005: Thermodynamic conditions favorable to superlative thunderstorm updraft, mixed phase microphysics and lightning flash rate. *Atmos. Res.*, **76**, 288-306.

Wolf, P., 2018: Utilizing radar and satellite to provide meaningful lightning initiation and cessation information for effective decision-making. *FDTD Satellite Applications Webinars*, https://rammb.cira.colostate.edu/training/visit/satellite_chat/20180926/.

Zhang, Y., F. Zhang, and D. J. Stensrud, 2018: Assimilating all-sky infrared radiances from GOES-16 ABI using an ensemble Kalman filter for convection-allowing severe thunderstorms prediction. *Mon. Wea. Rev.*, **146**, 3363-3381, https://doi.org/10.1175/MWR-D-18-0062.1.

Zhang, Y., D. J. Stensrud, and F. Zhang, 2019: Simultaneous assimilation of radar and all-sky satellite infrared radiance observations for convection-allowing ensemble analysis and prediction of severe thunderstorms. *Mon. Wea. Rev.*, **147**, 4389-4409, https://doi.org/10.1175/MWR-D-19-0163.1.

Zhou, Y., and G. Hooker, 2022: Decision tree boosted varying coefficient models. Data Mining and Knowledge Discovery. https://doi.org/10.1007/s10618-022-00863-y.

APPENDIX A: METHOD FOR APPROXIMATING THE EFFECTIVE RECEPTIVE FIELD

To get an estimate of the ERF we want to calculate and visualize how much each location in the input channels affects a specific output pixel in a considered neural network. A simple way to do so for a given input sample and chosen output pixel is to calculate the gradient of the output neuron with respect to the neurons in the input channels. Calculating this gradient is a common task in neural networks and built-in routines are readily available in neural network computing environments. However, the results tend to be noisy, and we thus use a modification of this approach, namely the SmoothGrad algorithm by *Smilkov et al. (2017)*. SmoothGrad calculates the gradient with respect to the input neurons several times, each time adding Gaussian noise to each pixel of each input channel before calculating the gradient, and then returns the average result. This approach, as the title of the *Smilkov et al. (2017)* aptly states, removes noise (in the results) by adding noise (in the input channels).

We use the SmoothGrad implementation of the "tf-explain" package (see https://tf-explain.readthedocs.io/en/latest/) with 100 samples and a noise level of 1.0. Note that this noise level is chosen extremely large on purpose (keep in mind that our inputs are scaled to values between just 0 and 1), because that makes the results less dependent on the specific sample that was chosen for the estimation. When interpreting the resulting ERF estimates for a neural network model one should keep in mind that the results vary based on i) chosen input sample, ii) chosen output pixel, and iii) random noise generated by SmoothGrad. Thus, it is important to generate estimates for variations of all these parameters and ensure that results are representative of the general trends. A property we noticed varying across those parameters is the presence of a few high intensity pixels in the resulting maps. Their number and location can vary and thus

should not be assigned special meaning. Aside from such details the overall distribution is fairly consistent, namely how diffuse the ERF is and how far it stretches out from the center. More generally, *results from this ERF approximation method should be seen as a random sample drawn from a given distribution, rather than each pixel value given specific meaning.*

APPENDIX B: LAYER-WISE RELEVANCE PROPAGATION (LRP)

A key idea of layer-wise relevance propagation is that it seeks to track *relevance* backward from an output neuron to the input image, by tracking backwards which neurons in the prior layer were most responsible for the values of a neuron in the later layer. To do so LRP does not use any of the built-in backpropagation rules of neural networks and develops instead its own set of customized rules. By applying those rules iteratively, an overall estimate of relevance in the input space is obtained. LRP is a fairly complex topic, and the details are beyond the scope of this chapter. For a detailed introduction see *Bach et al. (2015)*, *Montavon et al. (2018)*, or *Toms et al. (2019)*.

We are using the implementation of LRP in the "innvestigate" package for Tensorflow (see https://innvestigate.readthedocs.io/en/latest/). We are using the alpha-beta rule (Eq. (60) in *Bach et al. (2015)*) with alpha=1 and beta=0, to only approximate positive attribution, i.e., to identify locations for which higher activation values tend to make high values at the output *more* likely. We had to use a few methods to make this implementation work for our purpose. Firstly, we flattened the output layer of the NN into a vector to be able to prescribe which output pixel we want to examine. Secondly, we did not use the standard heatmap visualization provided by the package, but instead split the heatmap result for LRP into its separate channels and plotted them separately. *For the interpretation of LRP results one needs to keep in mind that LRP uses approximation rules and that it was specifically designed for classification tasks, not regression tasks, so results should always be interpreted as showing overall trends but should not be interpreted on a pixel-by-pixel level.*

APPENDIX C: FITTING A QUADRATIC MODEL

Define a quadratic model:

$$y = c_0 + c_1 x + c_2 x^2$$

Iterate over the data and accumulate the sums for each pixel:

$$M_x = \frac{1}{N}\sum x$$

$$M_{xx} = \frac{1}{N}\sum x^2$$

$$M_y = \frac{1}{N}\sum y$$

$$S_{xx} = \sum x^2 - \frac{1}{N}\left(\sum x\right)^2$$

$$S_{xxx} = \sum x^3 - \frac{1}{N}\sum x\left(\sum x\right)^2$$

$$S_{xxxx} = \sum x^4 - \frac{1}{N}\left(\sum x^2\right)^2$$

$$S_{yx} = \sum yx - \frac{1}{N}\sum y \sum x$$

$$S_{yxx} = \sum yx^2 - \frac{1}{N}\sum y \sum x^2$$

Then the coefficients of the model are given by:

$$c_2 = \frac{S_{yxx}S_{xx} - S_{yx}S_{xxx}}{S_{xxxx}S_{xx} - S_{xxx}^2}$$

$$c_1 = \frac{S_{yx}S_{xxxx} - S_{yxx}S_{xxx}}{S_{xxxx}S_{xx} - S_{xxx}^2}$$

$$c_0 = M_y - c_1 M_x - c_2 M_{xx}$$

DATA AVAILABILITY

The datasets created in this research are available at:

- Hilburn, Kyle (2023), GREMLIN CONUS1 Manually Selected Storms Dataset, Dryad, Dataset, https://doi.org/10.5061/dryad.m905qfv60.
- Hilburn, Kyle (2023), GREMLIN CONUS2 Dataset, http://dx.doi.org/10.25675/10217/235392.
- Hilburn, Kyle (2023), GREMLIN CONUS3 Dataset for 2020, Dryad, Dataset, https://doi.org/10.5061/dryad.h9w0vt4nq.
- Hilburn, Kyle (2023), GREMLIN CONUS3 Dataset for 2021, Dryad, Dataset, https://doi.org/10.5061/dryad.zs7h44jf2.
- Hilburn, Kyle (2023), GREMLIN CONUS3 Dataset for 2022, Dryad, Dataset, https://doi.org/10.5061/dryad.2jm63xstt.

The trained Version-1 GREMLIN model and accompanying code (including the Color Vision Deficiency accessible colormap for radar data) are available in the repository https://doi.org/10.5281/zenodo.7832223.