Interpretable Deep Learning for Spatial Analysis of Severe Hailstorms

DAVID JOHN GAGNE II, SUE ELLEN HAUPT, DOUGLAS W. NYCHKA, AND GREGORY THOMPSON

National Center for Atmospheric Research, Boulder, Colorado

(Manuscript received 5 September 2018, in final form 9 May 2019)

ABSTRACT

Deep learning models, such as convolutional neural networks, utilize multiple specialized layers to encode spatial patterns at different scales. In this study, deep learning models are compared with standard machine learning approaches on the task of predicting the probability of severe hail based on upper-air dynamic and thermodynamic fields from a convection-allowing numerical weather prediction model. The data for this study come from patches surrounding storms identified in NCAR convection-allowing ensemble runs from 3 May to 3 June 2016. The machine learning models are trained to predict whether the simulated surface hail size from the Thompson hail size diagnostic exceeds 25 mm over the hour following storm detection. A convolutional neural network is compared with logistic regressions using input variables derived from either the spatial means of each field or principal component analysis. The convolutional neural network statistically significantly outperforms all other methods in terms of Brier skill score and area under the receiver operator characteristic curve. Interpretation of the convolutional neural network through feature importance and feature optimization reveals that the network synthesized information about the environment and storm morphology that is consistent with our understanding of hail growth, including large lapse rates and a wind shear profile that favors wide updrafts. Different neurons in the network also record different storm modes, and the magnitude of the output of those neurons is used to analyze the spatiotemporal distributions of different storm modes in the NCAR ensemble.

1. Introduction

The size of a hailstone depends on the path of the hail through a storm and how favorable the environment is for hail growth along that path (Foote 1984). The path and growth conditions are influenced by both the vertical thermodynamic profile of the environment and the morphology of the storm in both the horizontal and vertical directions. Diagnostic tools that most effectively encapsulate all of these factors should have the most skill in discriminating between large and small hail at the surface. Environmental information derived only from the bulk thermodynamic profile has shown some skill in predicting large hail. Edwards and Thompson (1998) found no correlation between integrated radar or sounding indices and hail size. More recently, Manzato (2012) identified instability indices calculated at the 850- to 500-hPa levels, such as the lifted index and Showalter index, as having the highest correlation with hail occurrence. Johnson and Sugden (2014) and Púčik et al. (2015) both find deep-layer wind shear to discriminate hail size better than thermodynamic and low-level wind parameters. Expanding from scalar indices to full vertical profiles, the HAILCAST 1D hail growth model (Brimelow et al. 2002, 2006; Jewell and Brimelow 2009; Adams-Selin and Ziegler 2016) produces hail size estimates by simulating hail embryos growing in a steady-state updraft based on a full vertical wind and temperature profile and adjusting the hail growth based on localized variations in temperature and moisture. HAILCAST has demonstrated skill in diagnosing hail size from proximity soundings (Jewell and Brimelow 2009) and convection-allowing model environments (Adams-Selin et al. 2019). In this paper, we demonstrate that incorporating both vertical profile and spatial information into a deep learning hail size diagnostic model can provide both increased hail size analysis skill and insight into important factors for hail growth.

The importance of storm morphology to hail growth has been documented in observational and idealized modeling studies. Multiple observational studies (e.g., Nelson 1983; Foote 1984) have found a strong connection between the width of a thunderstorm updraft and the amount of hail growth. Idealized modeling studies of supercells have identified how small changes in the

Corresponding author: David John Gagne II, dgagne@ucar.edu

DOI: 10.1175/MWR-D-18-0316.1

moisture and wind profile can alter storm morphology and hail growth. Grant and van den Heever (2014) varied midlevel relative humidity in idealized supercell simulations and found that the resulting changes in wind flow and storm structure changed the mechanisms of hail growth and locations of maximum hail fall. Dennis and Kumjian (2017) found that increasing wind shear in the east–west direction elongates the updraft in that direction and promotes more hail growth by creating a larger hail embryo source region and increasing residence time in favorable regions of the updraft for hail growth. However, increasing wind shear in the north–south direction decreased hail growth by separating the embryo source regions from the available hydrometeors. Given the limited information about morphology found in sounding studies, and the limited sample size of existing observational and idealized modeling studies, combining the two sources of information with algorithms designed to process large amounts of spatial data may lead to greater understanding about what factors are most important for the growth of large hail.

Many automated weather forecasting algorithms contain methods for encoding spatial information, but these methods typically encode more information about the distribution of field values as opposed to structural features. Spatial encoding is typically performed either within the bounds of an object or within a neighborhood surrounding a fixed point. Object-based encodings offer the advantages of focusing on the relevant area of a discrete object and the reduced memory and processing loads from only calculating information about a limited number of objects versus a large number of grid points. Lakshmanan and Smith (2009) discuss the variety of spatial and temporal statistics that can be extracted from tracked storm objects, and Gagne et al. (2017) and Lagerquist et al. (2017) demonstrated how object-based machine learning methods produce skilled hail and severe wind forecasts, respectively. If the interactions among multiple objects are relevant for the prediction of a particular process, a spatiotemporal relational network framework can encode relationships, such as the distance and orientation between two objects, as well as interactions, such as whether objects overlap or if one object is contained within another object (McGovern et al. 2014).

The performance of object-based encodings can be limited by how the object is defined and by the way information is extracted from the object. The choice of area and intensity thresholds in the object-finding algorithm can have a large impact on the population of objects and their characteristics due to some potential objects being excluded or merged together, as Haberlie and Ashley (2018) have demonstrated for automated mesoscale convective system climatologies. Extracting environment information only from within a region defined by a radar reflectivity or other storm threshold (e.g., Gagne et al. 2017; Lagerquist et al. 2017) risks ignoring potentially valuable data from the inflow region of the storm. Expanding the area of interest to encompass a fixed-size region around a storm may help account for some of these issues.

Neighborhood spatial encodings apply a transformation to the values of a quantity over the area surrounding a fixed point in order to produce a low-dimensional representation of the region's spatial structure. Principal component analysis (PCA; Pearson 1901), also known as empirical orthogonal functions in the atmospheric science community, has been commonly used to identify regimes at both weather and climate time scales. Clustering of PCs has been used to identify weather regimes in which separate statistical models have been trained for weather prediction tasks (e.g., Greybush et al. 2008). Self-organizing maps (SOMs; Kohonen 1982) are a neural-network-based clustering and dimensionality reduction technique that has recently been used to identify near-storm atmospheric profiles (Nowotarski and Jensen 2013) and spatial configurations of the significant tornado parameter (Anderson-Frey et al. 2017) favorable for tornadoes. Unsupervised encoding methods can summarize the main features of a dataset without requiring labels, and both the components of PCA and SOMs can be visualized for interpretation. However, the features encoded by PCA and SOMs are not chosen to maximize predictive skill, and the top PCs or SOMs may not be the most important features for a given problem. The PCA and SOM features may also be artifacts of the choice of encoding method or model hyperparameters. For example, most of the variance in a time series of temperature may be explained by diurnal or seasonal cycles, but those forcings have limited utility for predicting large temperature changes produced by local weather phenomena, such as storms. The spatial weights for each PC can be strongly influenced by the shape of the domain, resulting in the same ‘‘Buell patterns'' no matter what kind of input data are used (Buell 1975, 1979; Richman and Lamb 1985; Richman 1993).

Deep learning methods (LeCun et al. 2015) offer the ability to encode spatial features at multiple scales and levels of abstraction with the explicit goal of encoding the features that maximize predictive skill. Machine learning models in the deep learning family typically consist of neural networks with multiple specialized or sparsely connected hidden layers, as opposed to traditional artificial neural networks, which contain one or two densely connected hidden layers. The specialized layers either encode spatial or temporal structure in the data, or they transform the input data to improve the model optimization process. The encodings are learned through a numerical optimization process rather than being developed through empirics. This automated encoding process has enabled deep learning models to create high-level representations of entities in a given dataset and to use those representations to generate more accurate predictions. Deep learning models have produced state-of-the-art performance on image recognition tasks (Krizhevsky et al. 2012). An open question is whether these deep learning models can encode spatial weather features and relate them to associated but unresolved severe weather hazards.

The purpose of this research is to evaluate if deep learning models can encode spatial weather data in a way that improves skill and physical interpretability of severe hail predictions over more traditional spatial encoding methods. A supervised convolutional neural network model is evaluated against unsupervised spatial mean and PCA encodings to determine whether the additional complexity of deep learning results in a significant improvement in the ability of the model to discriminate between storms that will produce simulated severe hail and those that do not. The inputs to all models are permuted to determine which inputs have the most impact on model performance. The internal encodings of each model are interrogated using feature interpretation methods to reveal what storm structures are associated with a high probability of severe hail.

2. Data and methods

a. Model output

Spatial storm information originates from the NCAR convection-allowing numerical weather prediction (NWP) model ensemble (Schwartz et al. 2015). The NCAR ensemble consists of 10 WRF-ARW V3.6.1 (Skamarock and Klemp 2008) members run at 3-km grid spacing over the contiguous United States initialized at 0000 UTC each day. The ensemble uses the Thompson microphysics scheme (Thompson et al. 2004, 2008; Thompson and Eidhammer 2014), MYJ planetary boundary layer (PBL) scheme (Mellor and Yamada 1982), Noah land surface model (Chen and Dudhia 2001), and the RRTMG radiation scheme (Mlawer et al. 1997). The forecast period of interest lies between forecast hours 12 and 36 (1200 to 1200 UTC the following day), which corresponds to the NOAA Storm Prediction Center Day 1 convective outlook valid period. The ensemble ran daily from April 2015 to December 2017, but on most days only surface variables and severe weather diagnostics were archived.

TABLE 1. List of the input fields for each machine learning model and the mean and standard deviation (SD) of each field to three significant figures.

Variable	Pressure level	Mean	SD
Geopotential height	500 hPa	5790 m	65.1 m
Geopotential height	700 hPa	3110 m	49.7 m
Geopotential height	850 hPa	1480 m	45.3 m
Temperature	500 hPa	261 K	2.58 K
Temperature	700 hPa	279 K	2.37 K
Temperature	850 hPa	290 K	2.91 K
Dewpoint	500 hPa	254 K	8.90 K
Dewpoint	700 hPa	275 K	4.15 K
Dewpoint	850 hPa	286 K	3.81 K
Zonal wind	500 hPa	10.1 m s^{-1}	7.44 m s^{-1}
Zonal wind	700 hPa	6.14 m s^{-1}	6.62 m s^{-1}
Zonal wind	850 hPa	2.74 m s^{-1}	7.05 m s^{-1}
Meridional wind	500 hPa	8.25 m s^{-1}	7.45 m s^{-1}
Meridional wind	700 hPa	7.48 m s^{-1}	6.39 m s^{-1}
Meridional wind	850 hPa	6.70 m s^{-1}	7.41 m s^{-1}

For the period from 3 May to 3 June 2016, instantaneous state variables describing weather conditions on isobaric levels were also archived. The NCAR ensemble model output variables and levels used as inputs to the machine learning models are listed in Table 1. The 850- and 700-hPa isobaric levels are used because they describe conditions within the inflow region of the storm, and the 500-hPa level is used because it captures conditions within the hail growth zone, which typically ranges between −10° and −30°C (Nelson 1983). We selected state variables instead of derived storm diagnostics, such as CAPE and wind shear, because most storm diagnostics are vertically integrated quantities that can hide the subtle variations in the storm environment important for discriminating severe hazards.

To build a dataset of storms, we extract every strong updraft from each NCAR ensemble member over the period of interest. The enhanced watershed method (Lakshmanan et al. 2009; Gagne et al. 2017) identifies storm tracks in the hourly maximum upward vertical velocity field that exceeded 10 m s^{-1} with a minimum storm track area of 10 grid cells to exclude updrafts only partially resolved horizontally. The updraft speed field should identify most individual storm cells in the model, including pulse thunderstorms, cells embedded in a line, and supercells, all of which can produce severe hail. Some small, weak, and stationary storms may be ignored by these criteria. No storm identification method will capture every storm perfectly, but these criteria produce a large sample of both hail- and non-hail-producing storms to evaluate. A 96-km-wide square patch is extracted around the centroid of each storm track in order to capture both the storm and the immediate surrounding environment. All of the input variables (Table 1) are instantaneous fields extracted at the start time of the updraft swath. Each input field value x_i is transformed to $\hat{x}_i$ based on the training data distribution mean u_i and standard deviation s_i such that the transformed distribution has mean zero and standard deviation one:

$$\hat{x}_i = \frac{x_i - u_i}{s_i}. \tag{1}$$

Each storm patch extracted from each NCAR ensemble member at each forecast time is treated as an independent sample, and all evaluation statistics are calculated on a storm-by-storm basis. The independence assumption is not strictly true because storms in similar environments will have similar characteristics, but because the storms are extracted from 12- to 36-h model integrations, the storms in each ensemble member exhibit different storm initiation times and locations, resulting in different convective evolutions.

Instead of using hail reports or radar-estimated hail size, the diagnosed maximum diameter of the graupel-hail species in the Thompson microphysics scheme at the lowest model level within bounds of the storm swath is used to determine whether a simulated storm produced hail greater than 25 mm in diameter. This size is equivalent to the National Weather Service size threshold for severe hail since 2010. Because we are using simulated hail sizes as the target for the ML models, we are effectively producing 1-h-ahead hail size predictions within the world of the NWP model. Using simulated hail sizes instead of hail reports or radar-estimated hail sizes ensures the most direct connection between simulated storm features and hail at the surface by eliminating any spatial and temporal displacement errors between a predicted storm and the resulting hail swath. We have chosen to perform a perfect model experiment instead of calibrating toward observations because our primary goal is to increase our understanding of the relationship between storm and environmental features and severe hail, rather than to increase forecast skill.

The Thompson microphysics hail representation provides a reasonable balance between realism and computational efficiency with its representation of graupel and hail. To represent both graupel and hail with a single microphysical species, the Thompson scheme varies the intercept parameter $N_{o,g}$ as a function inversely proportional to the graupel mixing ratio q_g (Thompson et al. 2004, 2008). For large q_g, $N_{o,g}$ is greatly reduced, which then increases the number concentration of larger diameter graupel-hail. The density of the graupel-hail species is held constant at 400 kg m^{-3}, which is a more graupel-like density. While sensitivity tests in Thompson et al. (2008) showed that intercept parameter changes outweigh the effect of density changes over a 12-h storm simulation, the density of the hail species can affect the fall speed and trajectory of hail within the simulated storm (Morrison and Milbrandt 2011).

The Thompson hail size diagnostic is a microphysics-agnostic method for estimating a reasonable maximum hail size directly from the particle size distribution (PSD) of a hail or graupel species. The gamma or exponential distribution used to represent the PSD extends to infinite sizes, but the number concentration does decay to the point where the probability of seeing a hailstone above a certain size is infinitesimal. The Thompson hail size diagnostic thus integrates backward across a range of logarithmically spaced diameters until it reaches a number concentration that ensures that a hailstone of that diameter would be detected. The diagnostic concentration threshold in the NCAR Ensemble during the 2016 Spring Experiment was 0.0001 particles m^{-3}, which roughly corresponds to expecting to find a hailstone of that size within the area of two American football fields. The current diagnostic concentration threshold in WRF version 4.0.3 is 0.0005 particles m^{-3}. This concentration threshold can also be replaced with a maximum percentile threshold. To validate that the Thompson hail size diagnostic and microphysics scheme approximate real-world hail behavior, we examine both the distribution of hail sizes and how well the diagnostic predicts severe hail when severe hail is reported in the vicinity of a simulated storm (Fig. 1). The hail reports are collected by the NOAA/National Weather Service in the Storm Data publication and are distributed online by the Storm Prediction Center (Storm Prediction Center 2019). Because the hail diagnostic uses logarithmic-spaced bins instead of linear, a direct histogram of the hail size values would produce a distorted hail size distribution. Instead, we resample the hail sizes within each bin from a uniform distribution, such that each bin contains the same number of hail events but with a continuous range of values. The distribution of predicted hail sizes (Fig. 1) shows a peak in frequency around 12.5 mm with an exponential decrease with diameter. The storm-maximum hail sizes exhibit a fatter tail and two separate regimes of decreasing frequency due to the parameterized relationship between graupel-hail mixing ratio and the intercept

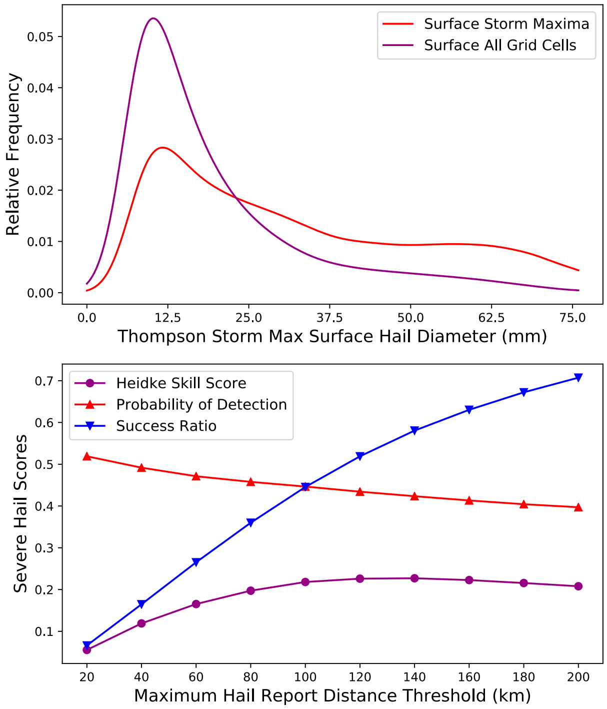

FIG. 1. (top) The distribution of the maximum (red) and all storm grid cell (purple) surface hail diameters in mm from the Thompson hail size diagnostic for each storm in the NCAR ensemble. (bottom) The probability of detection, success ratio, and Heidke skill score of the Thompson hail size diagnostic as a function of the maximum search radius for hail reports near a simulated storm from the NCAR ensemble.

parameter. The small, positive Heidke skill score (Wilks 2011) at different search radii from each storm to the hail reports shows that the NWP model can create storms that produce severe hail in areas where severe hail was reported, but the scheme also produces many false alarms based on the small Success Ratio, or ratio of true positive events to all positive predictions (Roebber 2009).

b. Machine learning models

We evaluate a series of machine learning models with spatial encodings of increasing complexity. Because we are more interested in evaluating the quality of the spatial encoding than that of the machine learning model, we use the same algorithm, a logistic regression, to transform our spatial encodings into hail probabilities. A logistic regression is a linear regression model with a logistic, or sigmoid $\{[1 + \exp(-x)]^{-1}\}$, nonlinear transformation applied to the output, so that the output values range between zero and one. Because of this transformation, logistic regressions are often used to estimate the probability of an event with binary outcomes. The ''logistic mean'' baseline model feeds the spatial mean of each input variable into a logistic regression. In the second model, ''logistic PCA,'' PCA transforms each input variable field independently to create a vector representation where each component is orthogonal, and the components are ordered by the percentage of variance explained in the original data. The top five principal components from each variable feed into a logistic regression. PCA applied to spatial fields can detect spatial gradients of varying frequencies and directions based on which patterns explain the largest proportion of variance in the data.

Both of the logistic regression models are trained using the *scikit-learn* (v. 0.19; Pedregosa et al. 2011) machine learning library. Lasso, or L1 norm, regularization (Tibshirani 1996) of the logistic regression model weights performs feature selection as part of the model optimization process by penalizing the absolute values of the regression weights to the point where less relevant input weights are set to 0. For both models, the magnitudes of the regularization weights vary during the grid search by powers of 10 from 0.1 to 0.001 to determine the most skilled set of inputs. The Lasso regularization magnitude and number of principal components (3 or 5) are selected through a grid search evaluation on a validation set consisting of storms from a subset of the ensemble members not included in the training set.

The deep learning model used in this study is a convolutional neural network (ConvNets; LeCun et al. 1990), which learns to encode features in multivariate spatial or temporal data with a series of convolutional layers in order to identify features at a particular spatial scale. Dimension reduction layers condense information spatially and help make the model invariant to slight differences in the locations of certain features. The ConvNets is built using the Keras high-level deep learning library (Chollet et al. 2015) with the Tensorflow low-level backend (Abadi et al. 2016).

A convolutional layer consists of a set of optimized feature maps, which are small patches of weights applied in a sliding window pattern over the spatial extent of the input grid. The output of each feature map is the sum of the product of the weights and the input values. Following the convolution operation, an activation function, or nonlinear transformation, is applied to the resulting field. ConvNets often use some variant of the rectified linear unit [ReLU; $\max(0, x)$] activation function because it preserves the magnitude of positive signals as they travel forward and backward through the network (LeCun et al. 2015). Convolution filters are applied across all inputs simultaneously, which allows them to identify correlated patterns

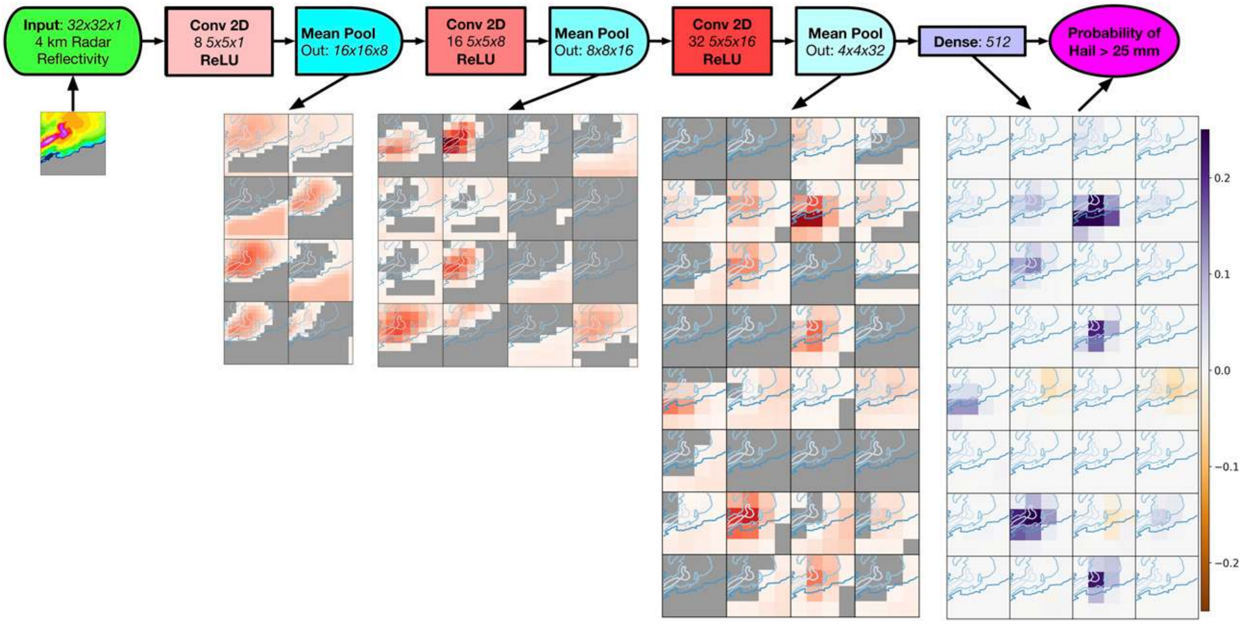

FIG. 2. Diagram of an example convolutional neural network for estimating the probability of hail from simulated radar reflectivity at 4 km above ground level. Each set of maps show the intensity of neuron activations for each convolutional filter in that layer after mean pooling has been performed on the data. Redder colors indicate a larger magnitude of activation. The original reflectivity image is overlaid on each filter activation map as a set of blue contours. The final set of maps shows the last filter activation multiplied by the matching weight in the dense output layer.

across multiple input variables or the results of previous convolutions. Figure 2 displays an example of how storm data are transformed and rescaled within a ConvNet. Simulated radar reflectivity is used in this figure for illustrative purposes but is not included as an input in the main machine learning model evaluation. The first set of convolutional filters broadly identifies areas of high and low simulated radar reflectivity. The second set of filters highlights a more diverse set of features, including the core of the storm, areas with strong intensity gradients, and low radar reflectivity areas.

Spatial dimension reduction is performed either through a pooling layer, in which the mean or maximum of a 2 × 2 gridcell region of the input is calculated, or through the use of strided convolutions, in which the convolution window is shifted by two grid cells instead of one and results in an output half the length and width of the input. The effect of the spatial dimensionality reduction is seen in the second and third layers of the original network (Fig. 2), in which each pixel covers a larger portion of the original storm. Convolutional layers deeper in the network operate on combinations of previous convolutional features and effectively cover a larger portion of the input space. Following the series of convolution and dimension reduction layers, the resulting feature cube is flattened into a feature vector. This vector is then input into a densely connected output layer with a sigmoid activation function. This layer assigns an independent weight to each element of the flat feature vector, sums them together, and rescales the output to create a probability. This final dense layer performs the same operations as a logistic regression. In Fig. 2, the strongest positive weights match up with the reflectivity core and the gradient between the storm edge and the core, which is where the updraft and hail growth region are typically located. The network produced a 98% chance of severe hail for this example, which correctly verified.

Given its large number of weights, how does a convolutional neural network minimize the chances of overfitting to noise in the data and failing to generalize? First, the convolutional layers impose a strong prior assumption on the model about the spatial structure of the data (Goodfellow et al. 2016). The convolution filters effectively share the same weights across all parts of the image and focus on a local area at any given moment. This reduces the effective number of weights that are fit and updates the weights multiple times per input sample. Dropout regularization (Srivastava et al. 2014), which gives each input a fixed probability of being set to zero on a pass through the network, promotes independence among the different weights in the model. Ridge, or L2 norm, regularization adds a penalty term to the error function that prefers solutions with small magnitude weights, which enables regression models to find reasonable solutions even with multiple correlated inputs at the expense of slightly higher training set error (Goodfellow et al. 2016). Batch normalization (Ioffe and Szegedy 2015) centers and rescales the internal values of a neural network in order to optimize the network to a certain error level faster and enable higher learning rates to be used.

The ConvNet configuration used in this study is shown in Fig. 3. The network consists of three strided convolutional layers with square 5 gridcell filters that combine the task of convolution and spatial dimensionality reduction. The number of convolutional filters is doubled in each successive layer. We train and validate separate ConvNets with a range of hyperparameters, or model settings that are fixed during training, in order to find a model configuration that should generalize well to unseen testing data. The initial number of filters is varied between 16 and 32. Dropout rates of 0.1 and 0.3 are validated. ReLU and Leaky ReLU activation functions are tested. Ridge (L2 norm) regularization is applied to each convolutional filter with a regularization coefficient of either 0.01 or 0.001. The Stochastic Gradient Descent and Adam (Kingma and Ba 2015) optimizers are both tested with learning rates of 0.001 and 0.0001. All networks are trained for 15 epochs with a batch size of 128 examples. While these hyperparameter settings do test some of the network architecture sensitivities, they are by no means comprehensive, and additional parameter search and architecture tuning could yield better results than what is shown.

The weights of a neural network are iteratively optimized through the process of stochastic gradient descent via back propagation. A batch, or random sample, of training data is drawn without replacement and sent forward through the network to generate predictions. The prediction error, or loss $L(x, y)$, is calculated, and then the partial derivative of the error with respect to each weight, or the gradient ∇_{w_i}, is determined. The expansion of the gradient for a particular weight includes the gradients for all other weights between that neuron and the output layer. Next, the gradient calculation has to be propagated backward through the network until gradients are found for all weights, hence the term back propagation. Once the gradients are calculated, each weight w_i is updated by stepping in the direction opposite the gradient by a proportional factor called the learning rate (α) such that

$$w_i = w_i - \alpha\nabla_{w_i}L(x, y). \qquad (2)$$

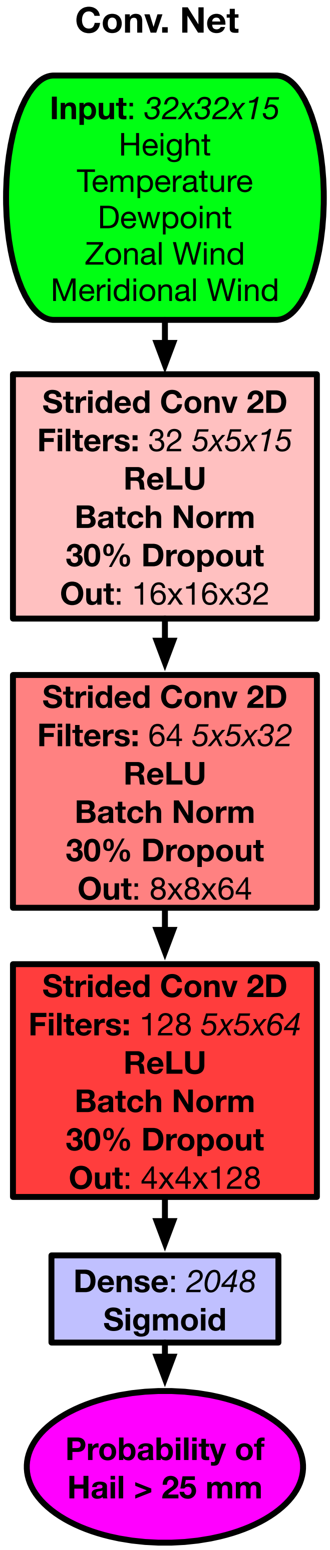

FIG. 3. Schematic of the convolutional neural network used for severe hail prediction. Each red block consists of a convolutional filter, a nonlinear activation function, and any regularizing transforms performed after the activation. The architecture components and settings shown produce the highest validation scores in the repeated hyperparameter searches.

The prediction error function used for this study is the Brier Score, which is equivalent to the mean squared error for probabilistic forecasts. The gradient, or derivative, of the Brier Score is the difference between the forecast probability and whether the event occurred. Stochastic gradient descent is performed using an optimization function with a specified learning rate, which controls the magnitude of the update at each step.

We assess the quality of the probabilistic forecasts from each machine learning model using standard probabilistic verification metrics. Many preprocessing factors, including the choice of hyperparameters and the training and test set composition affect the skill of each model. To control for some of these issues, a hyperparameter grid search, training, and testing procedure is conducted 30 times for each model type. A hyperparameter grid search is an exhaustive evaluation of every combination of discrete hyperparameter values that is commonly used for machine learning model tuning. During each search, the training and test storm data are split based on the NCAR Ensemble run initialization date with storms from 70% of the run dates used for training and storms from 30% of the run dates used for testing. For the hyperparameter grid search, storm patches from seven of the ten members are kept for training and storm patches from the remaining three members are held out for validation. Because this is a perfect model experiment and the storm extraction begins only after the model has run for at least 12 h, the ensemble members should be independent enough to minimize sharing of information across the training and validation examples. After the grid search has completed, the machine learning model is trained on the set of hyperparameters that produce the highest Brier skill score. The training for this machine learning model uses storms from all NCAR Ensemble members. This final model is then tested on storms from the ensemble runs on the held out test dates. This whole procedure repeats 30 times with different training and test run dates selected to determine the sensitivity of the hyperparameter settings to the training and test set composition. The test set scores aggregate across all 30 iterations. Permutation tests determine the statistical significance of the scores, and bootstrap resampling of the statistics from each iteration generates 95% confidence intervals on each verification diagram.

3. Hail model evaluation results

The primary evaluation statistics are the Brier skill score (BSS) and the area under the receiver operating characteristic (ROC) curve (AUC) (Mason 1982). The BSS can be decomposed into two terms and a scaling factor (Murphy 1973), as shown in Eq. (3):

TABLE 2. Evaluation scores for the different machine learning models. Every score except BSS reliability is positively oriented such that larger values mean better performance. The scores below are the means of the individual scores of 30 models trained and tested on different samples of training and testing ensemble run dates. Bold numbers indicate the best value for each metric.

Model	BSS	BSS reliability	BSS resolution	AUC
Logistic mean	0.12	0.017	0.13	0.75
Logistic PCA	0.29	**0.008**	0.30	0.85
ConvNet	**0.36**	0.019	**0.37**	**0.88**

$$\mathrm{BSS} = \frac{\frac{1}{N}\sum_{k=1}^{K} n_k(\overline{o}_k - \overline{o})^2 - \frac{1}{N}\sum_{k=1}^{K} n_k(p_k - \overline{o}_k)^2}{\overline{o}(1-\overline{o})}. \tag{3}$$

The first term in the numerator is the resolution, which is mean squared difference between the severe hail relative frequency of forecasts with a certain probability $\overline{o}_k$ and the observed climatological frequency of the event $\overline{o}$. The second term of the numerator is the reliability, which is the mean squared difference between the forecast probability p_k and the severe hail relative frequency. The scaling factor in the denominator is the uncertainty, which is only a function of the severe hail relative frequency of the event in the test set. Because each testing sample will have a slightly different severe hail relative frequency of severe hail, the reliability and resolution terms are scaled by the uncertainty, so that the terms can be aggregated properly. The three machine learning hail models are compared in Table 2. The ConvNet has the highest BSS with a statistically significant improvement ($\alpha < 0.01$ based on a permutation test) over the second most skilled model, the logistic PCA. In terms of BSS components, the ConvNet has the worst reliability but the best resolution. Since the differences in reliability are relatively small, the increase in resolution provided by the more sophisticated encodings is very helpful for this problem.

The components of the BSS are visualized with the attributes diagram, which plots the severe hail relative frequency of an event against the forecast probability (Fig. 4). A dashed horizontal line indicates the climatological probability of the event, and gray-shaded areas highlight where the BSS resolution exceeds the BSS reliability, contributing to a positive BSS. An inset plot shows the frequency of each forecast probability, providing a measure of sharpness. In the attributes diagram (Fig. 4), all of the models except for the logistic mean have observed relative

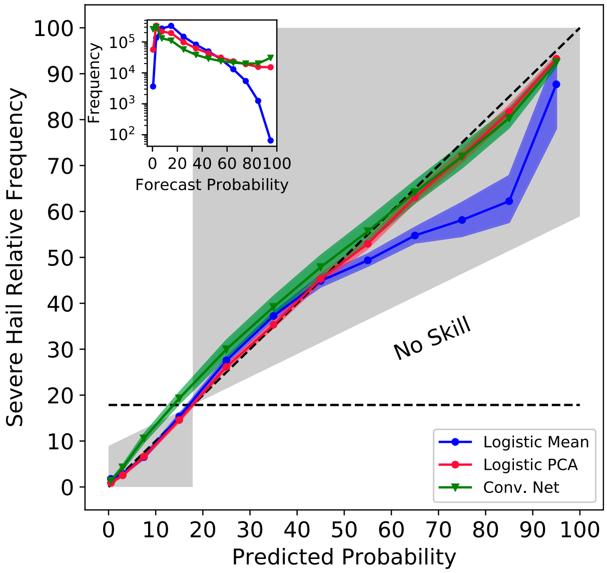

FIG. 4. Attributes diagram comparing the binned forecast probabilities of each spatial machine learning model with their observed relative frequencies. Ideally the curve for each model should fall along the dashed diagonal line. The shaded areas around each curve indicate the 95% bootstrap confidence interval for the severe hail relative frequency. The gray shaded area indicates the region where points on the curves contribute positively to the Brier skill score. The inset panel displays the frequency of all binned forecast probabilities for each model.

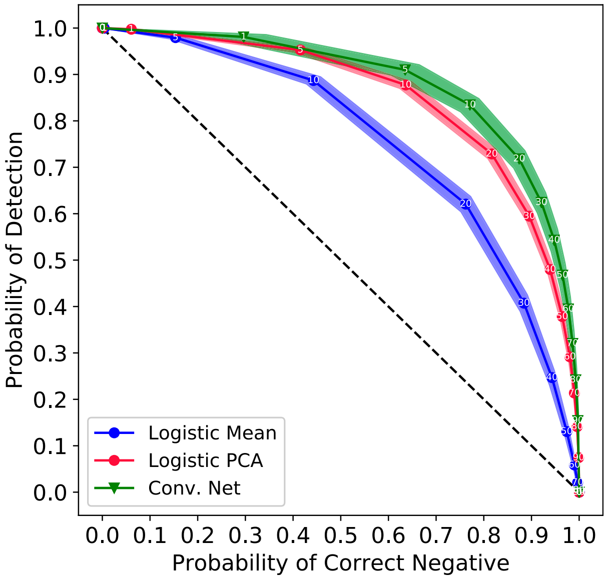

FIG. 5. ROC curve comparing different spatial machine learning methods. The ROC curve for a given model shows how POD and POCN vary by splitting probability forecasts into binary forecasts over series of increasing thresholds. The points from low probability thresholds appear in the top-left corner, and the points from high probability thresholds appear in the bottom-right corner. The shaded areas indicate the 95% bootstrap confidence interval for each model's ROC curve. The dashed line indicates where performance is no better than random.

frequencies that are close to the diagonal. The logistic mean model exhibits large deviations from the diagonal at high probability thresholds. The ConvNet displays a slight underforecasting bias at 20% probability and then an overforecasting bias at 80%. However, the ConvNet also has more forecasts with greater than 80% and less than 10% probability than any of the other models, which contributes to its high BSS resolution score. The logistic PCA models show very little deviation from the reliability diagonal.

The probabilistic hail forecasts are also evaluated with ROC curves and the related performance diagram. The ROC curve (Mason 1982) evaluates probabilistic forecasts by converting them into a series of binary deterministic forecasts and calculating the probability of detection (POD), the ratio of hits to the total number of positive events and probability of correct negative (POCN), the ratio of true negative events to the total number of negative events, at each probability threshold. The curves display how much varying the probability threshold affects the trade-off between minimizing misses and false alarms. The area under the ROC curve (AUC) is a measure of total prediction skill in which AUC scores greater than 0.5 have more skill than a random prediction and a score of 1 results in perfectly discriminating between positive and negative events. The conv.net has the highest AUC followed by the logistic PCA, and logistic mean with statistically significant separations among the different models (Table 2). The ROC curves for each model (Fig. 5) indicate that the ConvNet and logistic PCA consistently perform better across probability thresholds. Each point on the curve indicates an increase in probability by 10% from upper left to lower right. The models with higher AUC also have more probability thresholds with POD above 0.5, so these models can discriminate events with a low amount of false alarms even at higher probability thresholds.

Performance diagrams (Roebber 2009) perform a similar function to the ROC curve but replace POCN with the success ratio, which is the ratio of hits to the total number of positive forecasts. A performance diagram for the hail model evaluation is shown in Fig. 6. The performance diagram ignores the number of true negative events, so it is easier to differentiate forecasts that perform well on rare events. The performance diagram also displays critical success index (curved filled contours), a measure of accuracy, and frequency bias, a measure of the ratio of false alarms to misses, on the same diagram. There is more separation between the ConvNet and logistic PCA models.

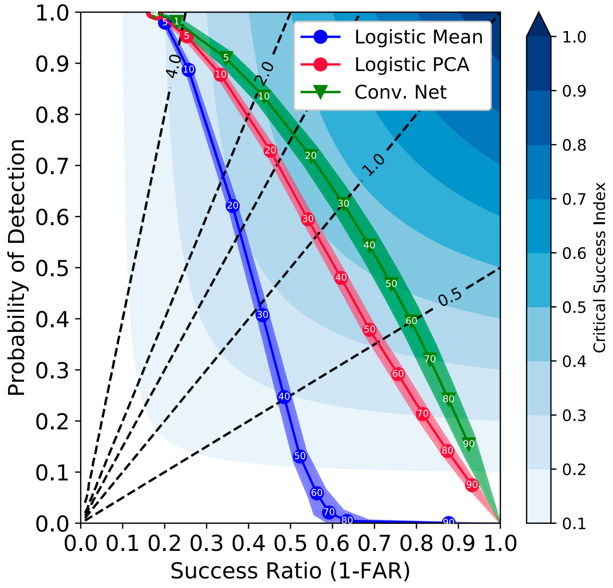

FIG. 6. Performance curve comparing different spatial machine learning methods. The curves show the probability of detection and success ratio as the forecast probability threshold increases (low probability thresholds are at the top left of the diagram and high probability thresholds are at the bottom right). The shaded areas indicate the 95% bootstrap confidence interval for each model's curve. The filled contours show regions of increasing critical success index. The dashed lines are contours of equal frequency bias. Points with frequency bias greater than one have more false alarms than misses, and points with frequency bias less than one have more misses than false alarms.

The training times for each machine learning model are shown in Table 3. The mean single model training time is calculated by dividing the mean search time by the number of parameter combinations and multiplying by the number of processes running simultaneously (8). The ConvNet trains faster than the logistic PCA model because the ConvNet was relatively small and can take full advantage of GPU parallelism and optimizations. The logistic PCA model requires performing 15 PCA decomposition and transformation procedures followed by an iterative fitting of a logistic regression, which are performed serially on one CPU. A parallel implementation of the PCA process could lead to much faster logistic PCA fitting times.

The repeated hyperparameter searches for each machine learning model reveals settings that are robust across variations in the training data. For both logistic mean and logistic PCA, a low inverse Lasso penalty of 0.1 produces the lowest validation errors and leads to most or all of the variables being selected. The logistic PCA models with 5 principal components per variable consistently outperform those with 3. For ConvNets, ReLU activations produce better results than Leaky ReLU activations. Dropout rates of 30% perform best for ConvNets on this problem. Increasing the number convolutional filters results in better performance most of the time.

TABLE 3. Summary of computational time for training each machine learning model. Deep learning models were trained on eight NVIDIA K40 graphical processing units (GPUs), and the logistic mean and logistic PCA models were trained on 8 Intel Xeon E5–2670 processors. Times are in minutes.

Model name	Parameter combinations	Mean search time	Mean single model training time
Logistic PCA	6	16	21.33
ConvNet	64	30	3.75
Logistic mean	3	0.6	1.59

4. Machine learning model interpretation

Users of machine learning models may not trust their output if they do not understand the model's decision-making process. Key questions for interpreting machine learning models include: which model inputs have the largest impact on the predictions and what features are encoded within a model's latent space? The first question can be addressed through model-agnostic interpretation methods, which treat the model as a black box and only operate on the inputs and outputs. Model-agnostic interpretation methods allow for apples-to-apples comparisons of model structures and can highlight how each model's assumptions and settings change the resulting model behavior. Permutation feature importance (Breiman 2001) ranks input variables based on how randomizing their values affects prediction error. First the model error on a set of examples is calculated. Then the values of each variable are permuted, or shuffled, among the examples. Then, the error is recalculated on the permuted data. A larger change in error is associated with higher importance.

Variable importance scores are shown in Fig. 7. The ConvNet and logistic mean models feature similar rankings with high importance for geopotential height and temperature. The variables extracted at 850 hPa tend to rank higher than variables from other pressure levels. The large drop in scores seen for all variables may be due to the shuffling of input fields affecting the values of five variables instead of one. The ranking of the absolute values of logistic PCA regression coefficients reveals a similar pattern to the logistic mean with geopotential height and temperature variables having the highest weights. The logistic PCA rankings differ from the other models potentially because permuting any of the input fields varies five of the inputs to

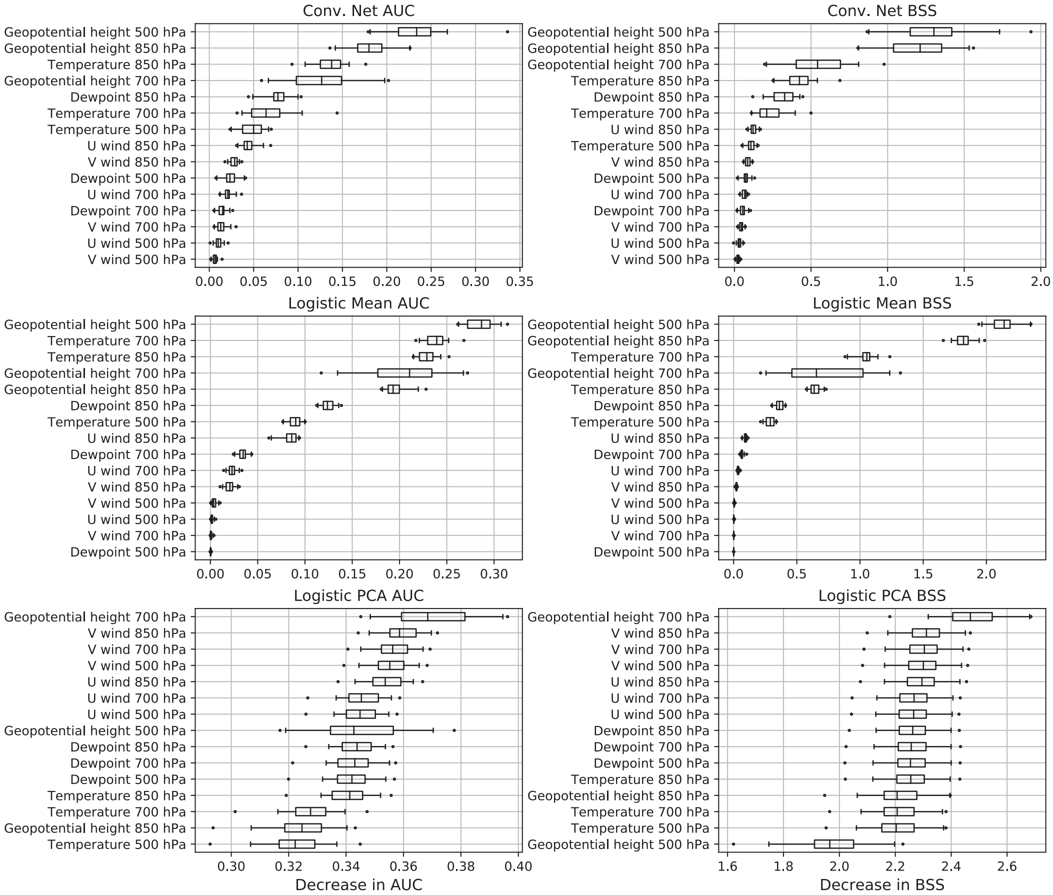

FIG. 7. Variable importance scores in terms of the decrease in AUC and BSS for each of the hail forecast models. The whiskers indicate the 2.5 and 97.5 percentiles of the distribution of importance scores from the 30 optimized machine learning models from each resampling of the training data.

the logistic regression, resulting in a major loss in skill. Although variable importance reveals which variables have the largest impact on predictive skill, it does not inform us about why each variable is important.

To provide more insight into why each variable is important, we utilize feature visualization by optimization, or backward optimization (Olah et al. 2017) on each machine learning model. The goal of feature visualization by optimization is to find the set of input values that maximize the activation of a particular neuron or set of neurons within a trained neural network. A logistic regression can be modeled as a one-layer neural network with no hidden layers, so the same process can be applied to the logistic mean and logistic PCA models as well. The results of this process should provide insight into what features each model has encoded. First, a neural network is trained. Then, the user picks a neuron or node to activate and creates a loss function that calculates the squared difference between the current output value (a_i) and the desired output value (a_d):

$$E = (a_i - a_d)^2. \tag{4}$$

Next, the user selects an initial input example to send through the network and be updated. This initial input can consist of all zeros, small random values, or a sample from the training set. The input example propagates forward through the network to the selected neuron, and the resulting activation is compared with the desired activation through the loss function. The gradient of each component of the input with respect to the activated neuron is calculated through back propagation. Because the gradient at the input is often very small, the gradient values are divided by the standard deviation of the gradient values. The input values are updated by subtracting the gradient multiplied by a learning rate from each input value [Eq. (2)]. Finally, the process is repeated until the neuron activation for a given input matches the desired activation. The resulting input field should contain relevant features for activating the selected

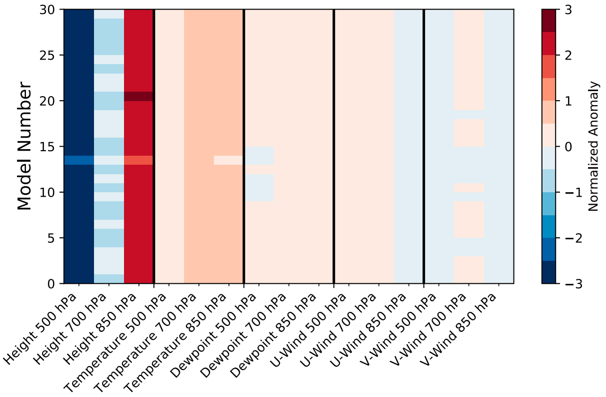

FIG. 8. Magnitudes of the fields backward optimized by the logistic mean model to maximize the probability of severe hail. All values are scaled in terms of normalized anomalies from the training data mean value of each input variable.

neuron. The gradient values from a single iteration of this process can also be used to create saliency maps that identify where in the input space a particular neuron is focusing its attention (Simoyan et al. 2014). The strided convolution architecture used in the ConvNets creates a checkerboard pattern in the feature optimization fields because of uneven propagation of the gradient (Odena et al. 2016). To remove the checkerboard pattern while preserving the spatial structures in the feature optimization output, a Gaussian filter is applied to each field and the smooth field is scaled by the standard deviation so that the input values range from −3 to 3. Using average pooling layers instead of strided convolutions or max pooling for spatial dimensionality reduction also reduces the checkerboard artifacts.

When applied to the output, the feature visualization by optimization process should reveal what input features lead each machine learning model to predict a high probability of hail. The strongest input anomalies in the logistic mean model (Fig. 8) come from the 850- and 500-hPa geopotential heights, which show positive and negative anomalies, respectively. The temperature fields show decreasing positive anomalies with decreasing pressure. Dewpoint and winds have consistent small anomalies, which is consistent with the small permutation feature importance assigned to them.

Feature visualization applied to the logistic PCA model (Fig. 9) illuminates more details consistent with the logistic mean visualization but providing more spatial context. The primary features are the same positive and negative geopotential height anomalies at 850 and 500 hPa along with a north–south temperature gradient in each field. Positive dewpoint anomalies appear at each level. The winds at 850 hPa show a weakly confluent pattern, the winds at 700 hPa have a slight rotation signature, and the winds at 500 hPa are from the south within the area of the height anomaly but are from the west outside of it.

The results of the input feature visualization on the output layer of a ConvNet are shown in Fig. 10. Many of the trained ConvNets produce an isolated storm similar to the top panel of Fig. 10. The input fields include many features associated with severe thunderstorms, including confluent warm moist air at low levels, directional wind shear with height, and strong lapse rates based on the change in temperature and height anomalies between 850 and 500 hPa. The ConvNet visualization generally matches the same patterns as the logistic PCA,

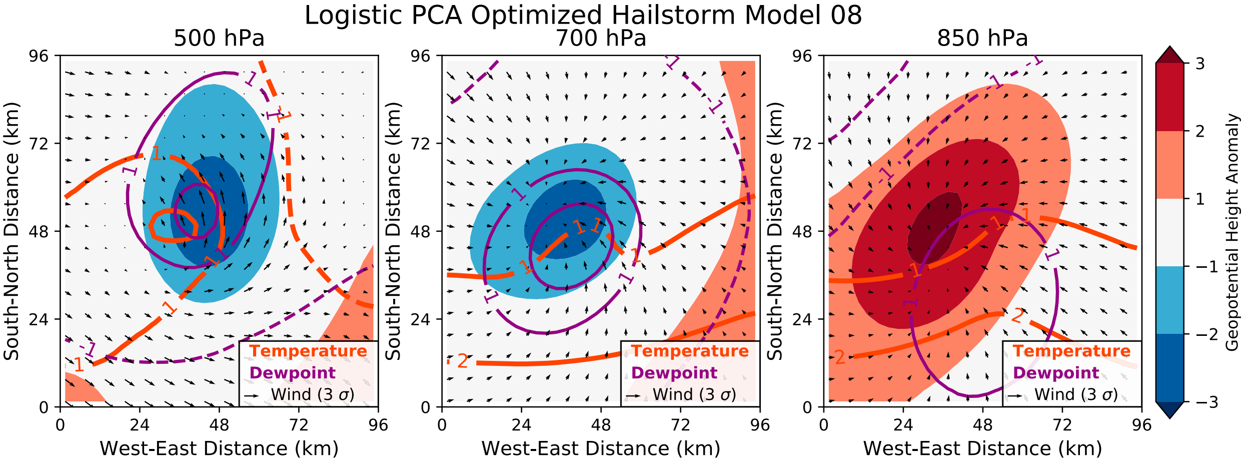

FIG. 9. A selection of input thermodynamic fields backward optimized by logistic PCA to maximize the probability of severe hail. The filled contours indicate geopotential height (red is positive and blue is negative), the orange–red contours indicate temperature, the purple contours show dewpoint, and the arrows display the wind vectors. All contours are scaled in terms of normalized anomalies from the training data mean value of each input variable.

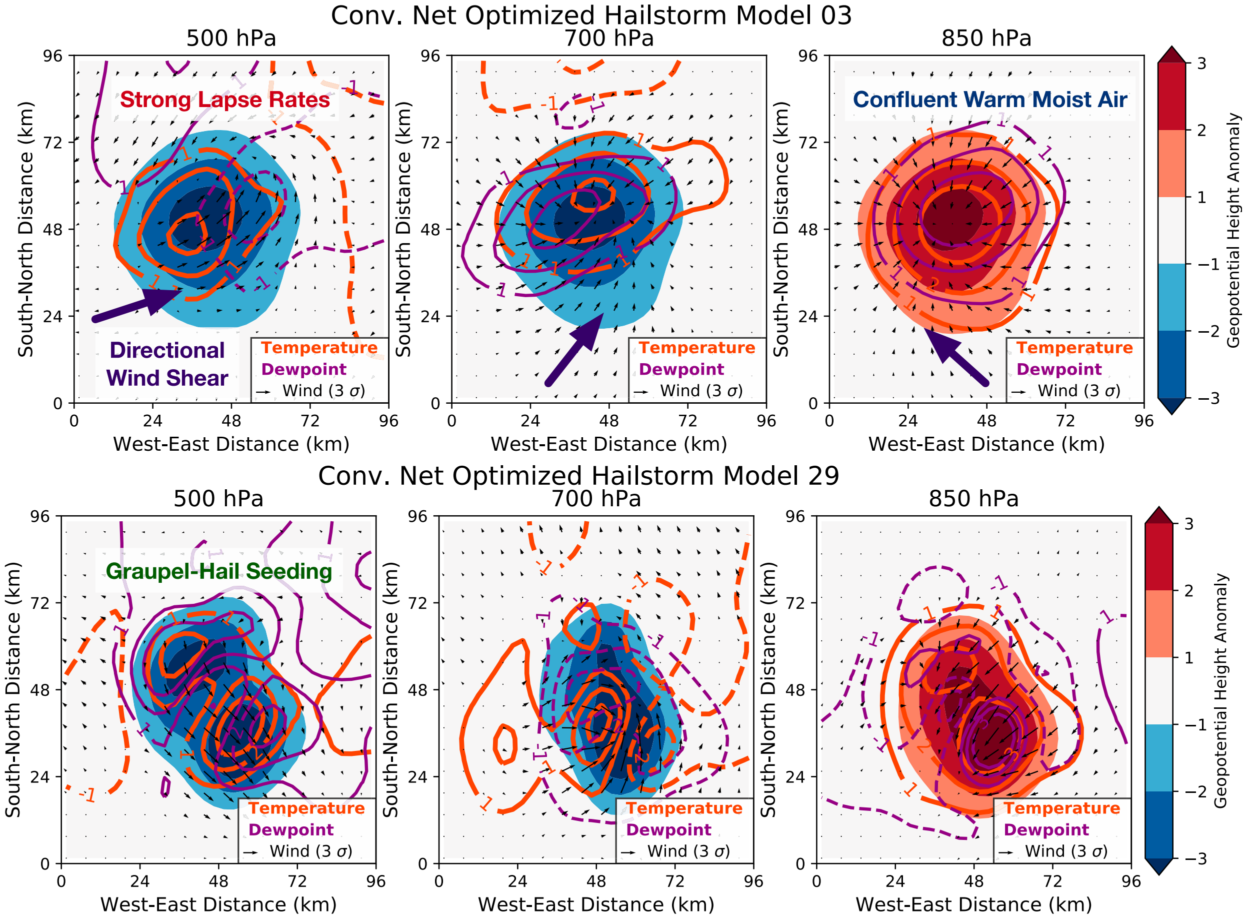

FIG. 10. A selection of input thermodynamic fields optimized by ConvNets to maximize the probability of severe hail. The filled contours indicate geopotential height (red is positive and blue is negative), the orange–red contours indicate temperature, the purple contours show dewpoint, and the arrows display the wind vectors. All contours are scaled in terms of normalized anomalies from the training data mean value of each input variable.

but the anomalies are all overlaid with each other, the winds are stronger near the center of the storm, and the wind field at 500 hPa is out of the west-southwest near the storm center and curves around north of the storm. The lapse rate information may be why the ConvNet and logistic mean models both featured high variable importance for the temperature and geopotential height variables. The rotational pattern seen in the 500-hPa wind field may be a precursor to the emergence of a rear flank downdraft, which has been shown to help large hail reach the surface by forcing cooler, drier air to the surface, which reduces hail melting (Rasmussen and Heymsfield 1987) and has been associated with the occurrence of large hail in hybrid supercell storms (Nelson 1987). The 500-hPa west–east wind vector orientation does match with the positive sensitivity to the wind shear direction found in Dennis and Kumjian (2017).

A few of the optimized hailstorms include two storms with wind fields linked at 500 hPa, as shown in the bottom panel of Fig. 10. This pattern may be associated with graupel and hail generated in one storm seeding the updraft of the downwind storm and encouraging the growth of large hail, which is a previously documented pattern of large hail formation (Heymsfield et al. 1980). The process should be resolvable in a convection-allowing model with Thompson microphysics because it relies on advection of graupel–hail mass, and an increase in graupel–hail mass should result in a sharp decrease in the intercept parameter and thus very large hail increasing in number concentration.

The input optimized features associated with activating individual neurons in the last convolutional layer provide information about the breadth of storm structures that the network associates with severe hail. Convolutional neurons were ranked based on how well their output values discriminate among hail and nonhail storms based on the area under the ROC curve. Figure 11 displays the optimized inputs

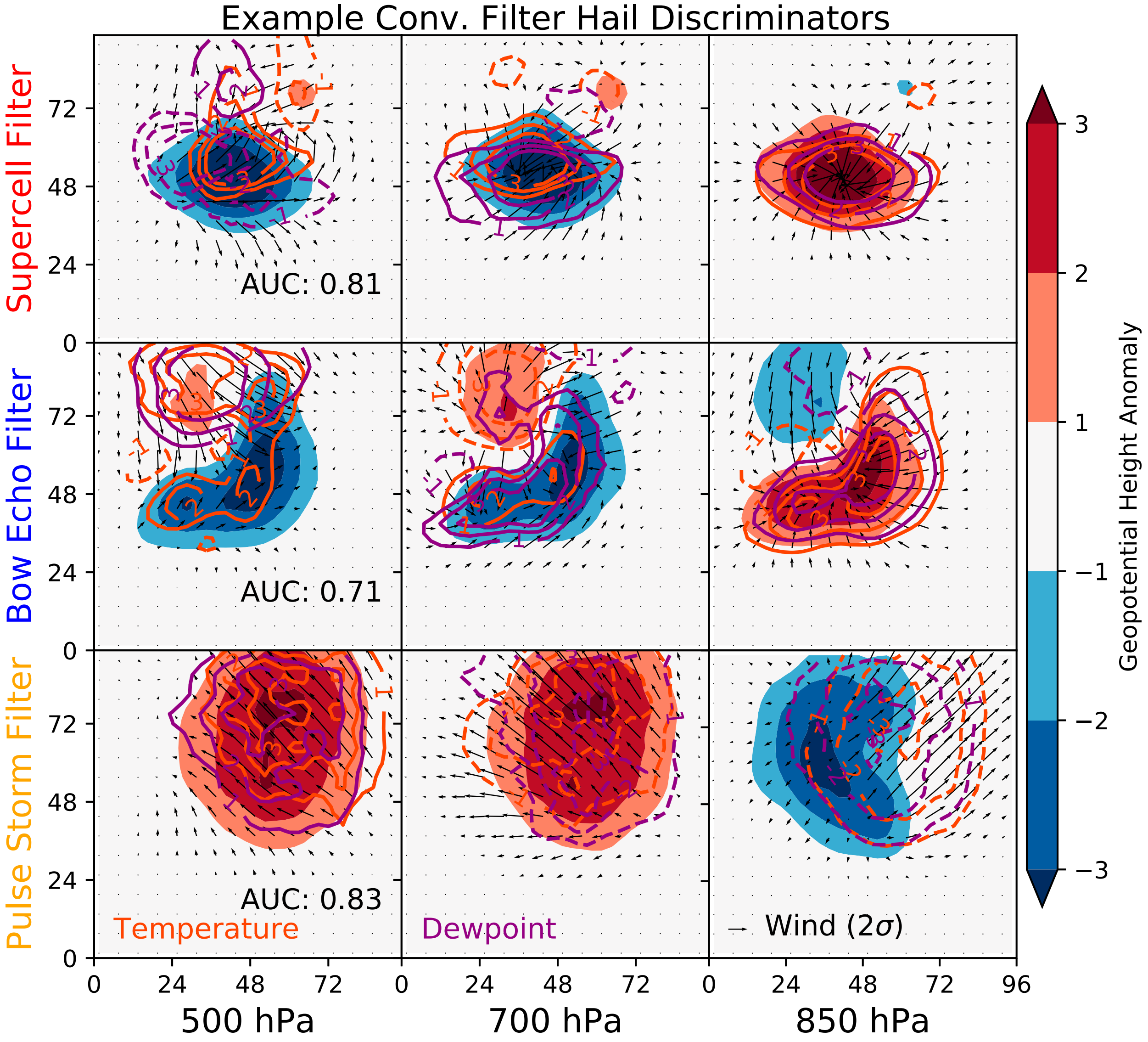

FIG. 11. Input fields that maximize the activation of neurons in the final convolution layer that strongly discriminate between severe and nonsevere hail based on AUC. The contours follow the same patterns as Figs. 9 and 10.

for three of the top discriminating neurons in the last convolutional layer. Each convolutional filter captures a different storm morphology. The first two filters are positively associated with large hail and resemble a supercell-like storm and a bow-echo- or bowing-line-like storm. Both of these feature the vertical geopotential height gradient in the main storm area. The bowing line also appears to have a strong rear flank downdraft with drier air at 700 hPa and cooler air at the surface. The third filter resembles a pulse-like storm and is negatively correlated with severe hail occurrence. The most notable differences between this storm and the others is the reversed geopotential height anomaly, negative temperature and moisture anomalies at the surface, and a lack of a clockwise hodograph or confluent winds at 850 and 700 hPa.

Once neurons have been associated with a particular storm morphology or other higher-level concept, those neurons can be used to filter a dataset into subsets for further analysis. The distributions of storms that activate the different filters in Fig. 11 are plotted in space (Fig. 12) and time (Fig. 13). The supercell storms are concentrated in the southern High Plains of the United States and tend to occur primarily in the late afternoon, which is expected for isolated supercell storms. The bow echo storms occur farther east and farther north with a temporal peak in the early evening but slightly later than

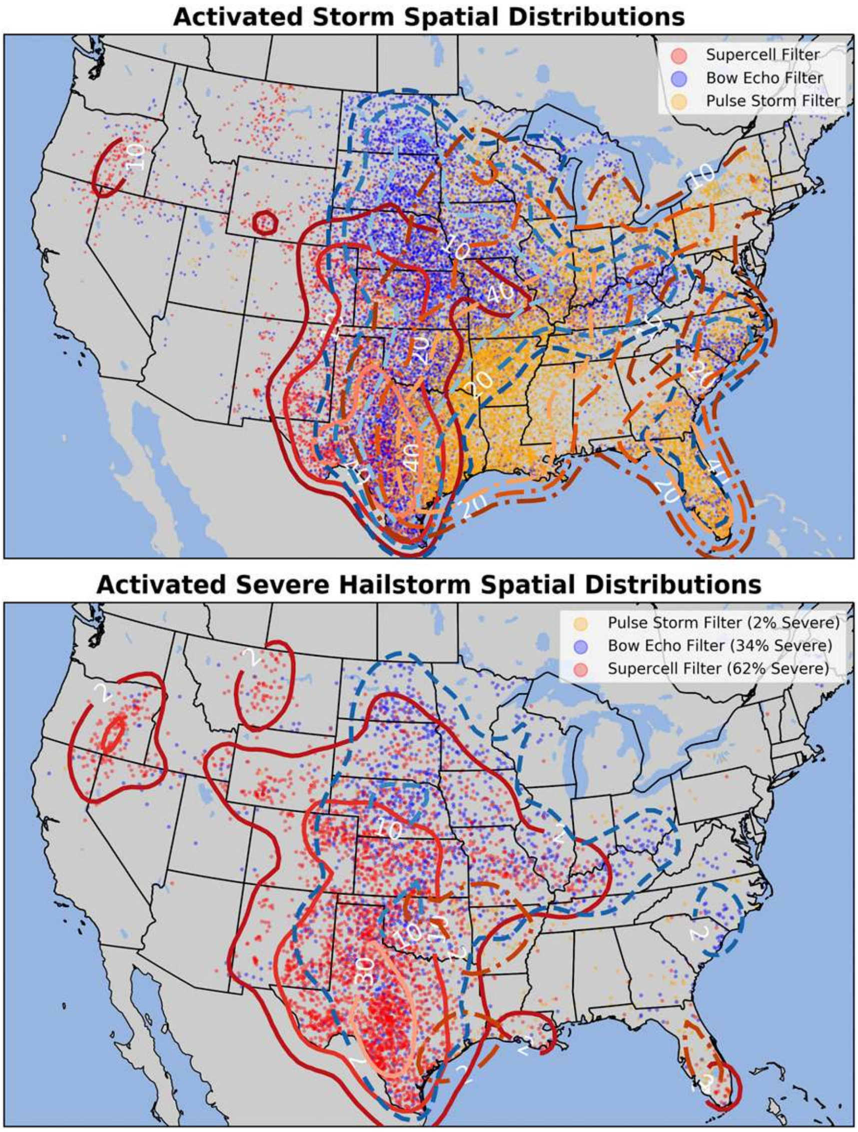
Activated Storm Spatial Distributions
● Supercell Filter
● Bow Echo Filter
● Pulse Storm Filter
10
40
20
Activated Severe Hailstorm Spatial Distributions
● Pulse Storm Filter (2% Severe)
● Bow Echo Filter (34% Severe)
● Supercell Filter (62% Severe)
2
10
30

FIG. 12. (top) Spatial kernel density estimates of the distributions of storms that activate one of the specified neurons from Fig. 11 with a magnitude of at least 0.5. The lighter red, blue, and orange contours indicate a higher spatial frequency of storms. The contour values are the number of storms per year at a given location. (bottom) The same spatial distribution by storm mode but only for the storms that produced severe hail.

the supercells. Unlike the other storm types, the pulse-like storms are found primarily in the lower Mississippi River valley, along the East Coast and the coasts of the Great Lakes, and peak in temporal frequency just after noon central standard time. The timing and location of these storms is consistent with a sea-breeze initiation mechanism (Byers and Rodebush 1948). The bottom panels of Figs. 12 and 13 show the spatial and temporal distributions of the storms that both activate each filter and produce severe hail. 62% of the supercell filter storms produce severe hail, but only 34% of the bow echo storms and 2% of the pulse storms produce severe hail. The combined areas covered by the different hailstorm modes closely match the extent of the hail climatologies in Cintineo et al. (2012) and Allen and Tippett (2015). The conditional spatial distribution of the supercells is largely unchanged. Severe hailstorms in May in the northern plains, Midwest, and the Carolinas tend to originate from bow echo storm types. The hail-producing pulse storms occur in regions around either Arkansas or Florida with none of the storms near the Great Lakes producing any severe hail.

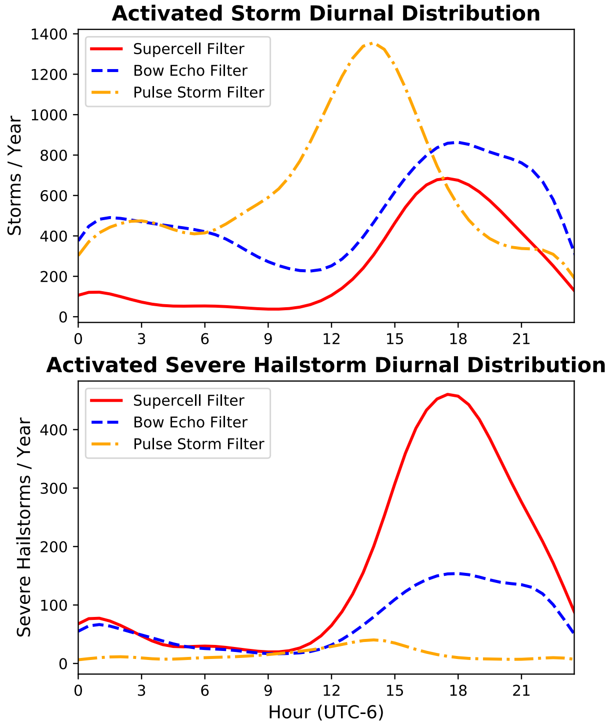

FIG. 13. Kernel density estimates of the diurnal distribution of the storms that activate each convolutional filter neuron from Fig. 11. Times are UTC − 6 h.

The prevalence of bow echo and pulse storms in the Southeast in May could partially explain why severe hail is less common there. Diurnally, the bow echo hailstorms tend to occur more frequently in the 1800 to midnight central standard time range than the full population of bow echo storms and less frequently between 0300 and 0900.

5. Discussion

The verification and interpretation results in this paper have shown that the more sophisticated spatial encoding process of the convolutional neural network provides increased skill over models using unsupervised feature dimensionality reduction and encodes multiple physical features that have been historically associated with severe hailstorms. These performance gains have been accomplished with relatively small convolutional neural networks compared with those being used in the domain of object recognition in images. Deeper networks have a larger representational capacity, but they also require more data and longer training periods for optimization. Because they require less training time and less data, smaller networks can be tuned to optimal parameter settings more quickly and generally have fewer parameters that require tuning. Also, compared with generic images, most gridded weather fields are often more constrained and less noisy. Many weather prediction problems can be framed in a way that requires the network to predict a small number of classes or quantities, compared with thousands of classes that are often found in image recognition datasets. Therefore, we recommend that atmospheric scientists interested in deep learning start by experimenting with smaller convolutional neural networks and only add more layers if skill is less than adequate for the problem.

The neural network feature visualization process in Fig. 11 and the spatiotemporal analysis of activations in Figs. 12 and 13 demonstrate the potential for deep learning interpretation to enable morphological analysis of complex spatial datasets without feature engineering or hand-labeling. Researchers could train their convolutional neural networks on storm data and identify which filters in the final convolutional layer best match the storm morphologies of interest. Climatological analysis of environmental parameters by storm morphology could provide greater insights to meteorologists about which storm modes are more likely with particular combinations of larger-scale environmental conditions. If the morphological feature identification is applied to real-time convection-allowing ensemble predictions, then forecasters could receive a probabilistic assessment of the timing and location of different storm modes and adjust their hazard assessments based on when supercells are expected to transition to a linear storm mode. The same deep learning and interpretation in this paper could also be applied to storms in gridded radar reflectivity mosaics to understand what radar features are linked with hail or tornado reports. This same interpretation approach could also be applied to a wide range of other weather and climate phenomena at different scales to perform similar kinds of analyses.

Although the logistic PCA model performed well in this evaluation, it did not provide any physical insights into the hail forecasting problem. The top components for each variable closely resembled the Buell (1979) patterns, likely due to varying locations and orientations of the storm in each patch. Further preprocessing of the data and recentering patches on the storm of interest rather than the track could potentially provide more of the signal in the PCs, but it would be difficult to capture the sophisticated multivariate patterns identified by the ConvNet. Interaction terms among the PCs could also be calculated to identify relationships among variables, but that would greatly expand the number of inputs that the logistic regression would have to optimize. Rotated principal components (Richman 1986) could provide more interpretability at the expense of the loss of orthogonality among components and may be investigated in future studies.

The results of this study have some limitations that should be considered when interpreting the results. First, the verification scores for this study should not be compared directly with other hail prediction evaluations (e.g., Gagne et al. 2017) based on observed data because the simulated storms within CAMs are often displaced in space, time, and intensity from the observed hailstorms, so the verification scores will generally be less skilled than the perfect model results shown here. Second, deep neural networks have many possible configurations and parameter settings, so better or worse results may be achieved under the ConvNet framework with other types of layers and other settings not explored in the limited grid search performed here. Third, different results and interpretations may be possible if the storm patches are centered on the storm at the beginning of the hour instead of on the center of the updraft track over the course of the hour. All methods may perform slightly better if the storm is centered, but given the variety of storm modes captured in this dataset and the already high skill scores, this change may not have a large impact on the results.

6. Conclusions

Deep learning models for encoding spatial weather data for analysis of severe hailstorms have been compared with traditional statistical approaches to determine the level of skill added by the deep learning encodings. Convolutional neural networks provide a statistically significant increase in multiple measures of prediction skill and result in sharper probabilistic predictions compared with logistic PCA. Through interpretation of the inputs that activate the hidden and output layers of the convolutional neural networks and logistic regressions, we discover that the machine learning models identify storm structures associated with severe hail in previous observational and modeling studies, including strong lapse rates, directional wind shear with a large west–east component, and seeding of graupel and hail from a weaker upstream storm. With these interpretation techniques, we can extract physical insights from machine learning models and compare the insights between models as well as with our conceptual understanding of the phenomenon. We also find that the neurons in convolutional neural networks can encode different storm morphologies, which enabled us to analyze our simulated hailstorm dataset and determine that supercell-like storms produce severe hail twice as frequently as bowing-line-like storms. In general, convolutional neural networks can serve as high-level feature detectors and enable semantic analysis of complex weather and climate datasets.

The machine learning and analysis software used in this paper can be accessed in the deepsky library available at https://github.com/djgagne/deepsky. Processed storm patch data, saved machine learning models, and other analysis data are available online (https://rda.ucar.edu/datasets/ds898.0/; Gagne 2019).

Acknowledgments. The authors thank Amy McGovern, Ryan Lagerquist, and Nathan Snook for discussions about deep learning interpretability. Ryan Sobash, Glen Romine, and Craig Schwartz assisted with access to the NCAR Ensemble output. Roy Rasmussen provided helpful insights into the hail growth processes encoded by the convolutional neural network. The authors thank Dorit Hammerling for a helpful internal review of the manuscript. Rich Loft, Ben Matthews, and Irfan Elahi provided access and support for GPU computing in the NCAR HPC Futures Lab, where machine learning training and evaluation was performed. The NCAR Ensemble output was processed on the Cheyenne supercomputer (Computational and Information Systems Laboratory 2017), provided by NCAR's Computational and Information Systems Laboratory. This research is funded by the NCAR Advanced Study Program Postdoctoral Fellowship. The National Center for Atmospheric Research is sponsored by the National Science Foundation. The authors also thank three anonymous reviewers whose comments resulted in an improved manuscript.

REFERENCES

Abadi, M., and Coauthors, 2016: Tensorflow: A system for large-scale machine learning. *Proc. 12th USENIX Symp. on Operating Systems Design and Implementation (OSDI'16)*, Savannah, GA, USENIX 265–283, https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf.

Adams-Selin, R. D., and C. L. Ziegler, 2016: Forecasting hail using a one-dimensional hail growth model within WRF. *Mon. Wea. Rev.*, **144**, 4919–4939, https://doi.org/10.1175/MWR-D-16-0027.1.

——, A. J. Clark, C. J. Melick, S. R. Dembek, I. L. Jirak, and C. L. Ziegler, 2019: Evolution of WRF-HAILCAST during the 2014–16 NOAA/Hazardous Weather Testbed spring forecasting experiments. *Wea. Forecasting*, **34**, 61–79, https://doi.org/10.1175/WAF-D-18-0024.1.

Allen, J. T., and M. K. Tippett, 2015: The characteristics of United States hail reports: 1955–2014. *Electron. J. Severe Storms Meteor.*, **10** (3), http://www.ejssm.org/ojs/index.php/ejssm/article/viewArticle/149.

Anderson-Frey, A. K., Y. P. Richardson, R. L. Thompson, and B. T. Smith, 2017: Self-organizing maps for the investigation of tornadic near-storm environments. *Wea. Forecasting*, **32**, 1467–1475, https://doi.org/10.1175/WAF-D-17-0034.1.

Breiman, L., 2001: Random forests. *Mach. Learn.*, **45**, 5–32, https://doi.org/10.1023/A:1010933404324.

Brimelow, J. C., G. W. Reuter, and E. R. Poolman, 2002: Modeling maximum hail size in Alberta thunderstorms. *Wea. Forecasting*, **17**, 1048–1062, https://doi.org/10.1175/1520-0434(2002)017<1048:MMHSIA>2.0.CO;2.

——, ——, R. Goodson, and T. W. Krauss, 2006: Spatial forecasts of maximum hail size using prognostic model soundings and hailcast. *Wea. Forecasting*, **21**, 206–219, https://doi.org/10.1175/WAF915.1.

Buell, C. E., 1975: The topography of empirical orthogonal functions. Preprints *Fourth Conf. on Probability and Statistics in Atmospheric Sciences*, Tallahassee, FL, Amer. Meteor. Soc., 188–193.

——, 1979: On the physical interpretation of empirical orthogonal functions. Preprints, *Sixth Conf. on Probability and Statistics in Atmospheric Sciences*, Banff, Alberta, Canada, Amer. Meteor. Soc., 112–117.

Byers, H. R., and H. R. Rodebush, 1948: Causes of thunderstorms of the Florida peninsula. *J. Meteor.*, **5**, 275–280, https://doi.org/10.1175/1520-0469(1948)005<0275:COTOTF>2.0.CO;2.

Chen, F., and J. Dudhia, 2001: Coupling an advanced land surface–hydrology model with the Penn State–NCAR MM5 modeling system. Part I: Model implementation and sensitivity. *Mon. Wea. Rev.*, **129**, 569–585, https://doi.org/10.1175/1520-0493(2001)129<0569:CAALSH>2.0.CO;2.

Chollet, F., and Coauthors, 2015: Keras: The Python Deep Learning Library. Accessed 31 August 2018, https://keras.io.

Cintineo, J. L., T. M. Smith, V. Lakshmanan, H. E. Brooks, and K. L. Ortega, 2012: An objective high-resolution hail climatology of the contiguous United States. *Wea. Forecasting*, **27**, 1235–1248, https://doi.org/10.1175/WAF-D-11-00151.1.

Computational and Information Systems Laboratory, 2017: Cheyenne: HPE/SGI ICE XA System (NCAR Community Computing). Tech. Rep., National Center for Atmospheric Research, Boulder, CO, accessed 1 September 2018, https://doi.org/10.5065/D6RX99HX.

Dennis, E. J., and M. R. Kumjian, 2017: The impact of vertical wind shear on hail growth in simulated supercells. *J. Atmos. Sci.*, **74**, 641–663, https://doi.org/10.1175/JAS-D-16-0066.1.

Edwards, R., and R. L. Thompson, 1998: Nationwide comparisons of hail size with WSR-88D vertically integrated liquid water and derived thermodynamic sounding data. *Wea. Forecasting*, **13**, 277–285, https://doi.org/10.1175/1520-0434(1998)013<0277:NCOHSW>2.0.CO;2.

Foote, G. B., 1984: A study of hail growth utilizing observed storm conditions. *J. Climate Appl. Meteor.*, **23**, 84–101, https://doi.org/10.1175/1520-0450(1984)023<0084:ASOHGU>2.0.CO;2.

Gagne, D. J., 2019: Interpretable deep learning for spatial analysis of severe hailstorms: Storm and analysis data. Research Data Archive at the National Center for Atmospheric Research, Computational and Information Systems Laboratory, https://doi.org/10.5065/6CJA-B154.

——, A. McGovern, S. E. Haupt, R. A. Sobash, J. K. Williams, and M. Xue, 2017: Storm-based probabilistic hail forecasting with machine learning applied to convection-allowing ensembles. *Wea. Forecasting*, **32**, 1819–1840, https://doi.org/10.1175/WAF-D-17-0010.1.

Goodfellow, I., Y. Bengio, and A. Courville, 2016: *Deep Learning*. MIT Press, 775 pp.

Grant, L. D., and S. C. van den Heever, 2014: Microphysical and dynamical characteristics of low-precipitation and classic supercells. *J. Atmos. Sci.*, **71**, 2604–2624, https://doi.org/10.1175/JAS-D-13-0261.1.

Greybush, S., S. Haupt, and G. Young, 2008: The regime dependence of optimally weighted ensemble model consensus forecasts of surface temperature. *Wea. Forecasting*, **23**, 1146–1161, https://doi.org/10.1175/2008WAF2007078.1.

Haberlie, A. M., and W. S. Ashley, 2018: A method for identifying midlatitude mesoscale convective systems in radar mosaics. Part I: Segmentation and classification. *J. Appl. Meteor. Climatol.*, **57**, 1575–1598, https://doi.org/10.1175/JAMC-D-17-0293.1.

Heymsfield, A. J., A. R. Jameson, and H. W. Frank, 1980: Hail growth mechanisms in a Colorado storm: Part II: Hail formation processes. *J. Atmos. Sci.*, **37**, 1779–1807, https://doi.org/10.1175/1520-0469(1980)037<1779:HGMIAC>2.0.CO;2.

Ioffe, S., and C. Szegedy, 2015: Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proc. 32nd Int. Conf. on Machine Learning*, Vol. 37, Lille, France, PMLR, 448–456, http://proceedings.mlr.press/v37/ioffe15.html.

Jewell, R., and J. Brimelow, 2009: Evaluation of Alberta hail growth model using severe hail proximity soundings from the United States. *Wea. Forecasting*, **24**, 1592–1609, https://doi.org/10.1175/2009WAF2222230.1.

Johnson, A. W., and K. E. Sugden, 2014: Evaluation of sounding-derived thermodynamic and wind-related parameters associated with large hail events. *Electron. J. Severe Storms Meteor.*, **9** (5), http://www.ejssm.org/ojs/index.php/ejssm/article/viewArticle/137.

Kingma, D. P., and J. Ba, 2015: Adam: A method for stochastic optimization. *Int. Conf. on Learning Representations*, San Diego, CA, https://arxiv.org/abs/1412.6980.

Kohonen, T., 1982: Self-organized formation of topologically correct feature maps. *Biol. Cybern.*, **43**, 59–69, https://doi.org/10.1007/BF00337288.

Krizhevsky, A., I. Sutskever, and G. Hinton, 2012: ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25* (*NIPS 2012*), F. Pereira et al. Eds., Neural Information Processing Systems Foundation, Inc., 1097–1105, http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

Lagerquist, R., A. McGovern, and T. Smith, 2017: Machine learning for real-time prediction of damaging straight-line convective wind. *Wea. Forecasting*, **32**, 2175–2193, https://doi.org/10.1175/WAF-D-17-0038.1.

Lakshmanan, V., and T. Smith, 2009: Data mining storm attributes from spatial grids. *J. Atmos. Oceanic Technol.*, **26**, 2353–2365, https://doi.org/10.1175/2009JTECHA1257.1.

——, K. Hondl, and R. Rabin, 2009: An efficient, general-purpose technique for identifying storm cells in geospatial images. *J. Atmos. Oceanic Technol.*, **26**, 523–537, https://doi.org/10.1175/2008JTECHA1153.1.

LeCun, Y., B. E. Boser, J. S. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, 1990: Handwritten digit recognition with a back-propagation network. *Advances in Neural Information Processing Systems*, D. S. Touretzky, Ed., Morgan Kaufmann Publishers, Inc., 396–404, http://papers.nips.cc/

paper/293-handwritten-digit-recognition-with-a-back-propagation-network.pdf.

——, Y. Bengio, and G. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444, https://doi.org/10.1038/nature14539.

Manzato, A., 2012: Hail in northeast Italy: Climatology and bivariate analysis with the sounding-derived indices. *J. Appl. Meteor. Climatol.*, **51**, 449–467, https://doi.org/10.1175/JAMC-D-10-05012.1.

Mason, I. B., 1982: A model for assessment of weather forecasts. *Aust. Meteor. Mag.*, **30**, 291–303.

McGovern, A., D. J. Gagne, J. K. Williams, R. A. Brown, and J. B. Basara, 2014: Enhancing understanding and improving prediction of severe weather through spatiotemporal relational learning. *Mach. Learn.*, **95**, 27–50, https://doi.org/10.1007/s10994-013-5343-x.

Mellor, G. L., and T. Yamada, 1982: Developement of turbulence closure model for geophysical fluid problems. *Rev. Geophys. Space Phys.*, **20**, 851–875, https://doi.org/10.1029/RG020i004p00851.

Mlawer, E. J., S. J. Taubman, P. D. Brown, M. J. Iacono, and S. A. Clough, 1997: Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave. *J. Geophys. Res.*, **102**, 16 663–16 682, https://doi.org/10.1029/97JD00237.

Morrison, H., and J. Milbrandt, 2011: Comparison of two-moment bulk microphysics schemes in idealized supercell thunderstorm simulations. *Mon. Wea. Rev.*, **139**, 1103–1130, https://doi.org/10.1175/2010MWR3433.1.

Murphy, A. H., 1973: A new vector partition of the probability score. *J. Appl. Meteor.*, **12**, 595–600, https://doi.org/10.1175/1520-0450(1973)012<0595:ANVPOT>2.0.CO;2.

Nelson, S. P., 1983: The influence of storm flow structure on hail growth. *J. Atmos. Sci.*, **40**, 1965–1983, https://doi.org/10.1175/1520-0469(1983)040<1965:TIOSFS>2.0.CO;2.

——, 1987: The hybrid multicellular-supercellular storm—An efficient hail producer. Part II: General characteristics and implications for hail growth. *J. Atmos. Sci.*, **44**, 2060–2073, https://doi.org/10.1175/1520-0469(1987)044<2060:THMSEH>2.0.CO;2.

Nowotarski, C. J., and A. A. Jensen, 2013: Classifying proximity soundings with self-organizing maps toward improving supercell and tornado forecasting. *Wea. Forecasting*, **28**, 783–801, https://doi.org/10.1175/WAF-D-12-00125.1.

Odena, A., V. Dumoulin, and C. Olah, 2016: Deconvolution and checkerboard artifacts. Distill, accessed 15 July 2018, https://distill.pub/2016/deconv-checkerboard/.

Olah, C., A. Mordvintsev, and L. Schubert, 2017: Feature visualization: How neural networks build up their understanding of images. Distill, accessed 15 July 2018, https://distill.pub/2017/feature-visualization/.

Pearson, K., 1901: On lines and planes of closest fit to systems of points in space. *Philos. Mag.*, **2**, 559–572, https://doi.org/10.1080/14786440109462720.

Pedregosa, F., and Coauthors, 2011: Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.

Púčik, T., P. Groenemeijer, D. Rýva, and M. Kolář, 2015: Proximity soundings of severe and nonsevere thunderstorms in central Europe. *Mon. Wea. Rev.*, **143**, 4805–4821, https://doi.org/10.1175/MWR-D-15-0104.1.

Rasmussen, R. M., and A. J. Heymsfield, 1987: Melting and shedding of graupel and hail. Part I: Model physics. *J. Atmos. Sci.*, **44**, 2754–2763, https://doi.org/10.1175/1520-0469(1987)044<2754:MASOGA>2.0.CO;2.

Richman, M. B., 1986: Rotation of principal components. *Int. J. Climatol.*, **6**, 293–335, https://doi.org/10.1002/joc.3370060305.

——, 1993: Comments on "The effect of domain shape on principal components analysis." *Int. J. Climatol.*, **13**, 203–218, https://doi.org/10.1002/joc.3370130206.

——, and P. J. Lamb, 1985: Climatic pattern analysis of three- and seven-day summer rainfall in the central United States: Some methodological considerations and a regionalization. *J. Climate Appl. Meteor.*, **24**, 1325–1343, https://doi.org/10.1175/1520-0450(1985)024<1325:CPAOTA>2.0.CO;2.

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

Schwartz, C. S., G. S. Romine, R. A. Sobash, K. Fossell, and M. L. Weisman, 2015: NCAR's experimental real-time convection-allowing ensemble prediction system. *Wea. Forecasting*, **30**, 1645–1654, https://doi.org/10.1175/WAF-D-15-0103.1.

Simoyan, K., A. Vedaldi, and A. Zisserman, 2014: Deep inside convolutional networks: Visualising image classification models and saliency maps. *Int. Conf. on Learning Representations Workshop*, Banff, Canada, https://arxiv.org/abs/1312.6034.

Skamarock, W. C., and J. B. Klemp, 2008: A time-split nonhydrostatic atmospheric model for weather research and forecasting applications. *J. Comput. Phys.*, **227**, 3465–3485, https://doi.org/10.1016/j.jcp.2007.01.037.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, 2014: Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**, 1929–1958.

Storm Prediction Center, 2019: Storm Prediction Center WCM Page. NOAA, accessed 1 February 2019, https://www.spc.noaa.gov/wcm/.

Thompson, G., and T. Eidhammer, 2014: A study of aerosol impacts on clouds and precipitation development in a large winter cyclone. *J. Atmos. Sci.*, **71**, 3636–3658, https://doi.org/10.1175/JAS-D-13-0305.1.

——, R. M. Rasmussen, and K. Manning, 2004: Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part I: Description and sensitivity analysis. *Mon. Wea. Rev.*, **132**, 519–542, https://doi.org/10.1175/1520-0493(2004)132<0519:EFOWPU>2.0.CO;2.

——, P. R. Field, R. M. Rasmussen, and W. D. Hall, 2008: Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part II: Implementation of a new snow parameterization. *Mon. Wea. Rev.*, **136**, 5095–5115, https://doi.org/10.1175/2008MWR2387.1.

Tibshirani, R., 1996: Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B*, **58**, 267–288.

Wilks, D. S., 2011: *Statistical Methods in the Atmospheric Sciences*. 3rd ed. International Geophysics Series, Vol. 100, Academic Press, 704 pp.