Transparent Machine Learning: Training and Refining an Explainable Boosting Machine to Identify Overshooting Tops in Satellite Imagery

Nathan Mitchell[a] , Lander Ver Hoef[b] , Imme Ebert-Uphoff [b,c] , Kristina Moen[d] , Kyle Hilburn[b] , Yoonjin Lee[e] , Emily J. King[d]

[a] *Department of Statistics, Colorado State University, Fort Collins, Colorado*

[b] *Cooperative Institute for Research in the Atmosphere, Colorado State University, Fort Collins, Colorado*

[c] *Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, Colorado*

[d] *Department of Mathematics, Colorado State University, Fort Collins, Colorado*

[e] *School of Earth and Environmental Sciences, Seoul National University, Seoul, South Korea*

Corresponding author: Nathan Mitchell, nathan.mitchell24@alumni.colostate.edu

ABSTRACT: An Explainable Boosting Machine (EBM) is an interpretable machine learning (ML) algorithm that has benefits in high risk applications but has not yet found much use in atmospheric science. The overall goal of this work is twofold: (1) explore the use of EBMs, in combination with feature engineering, to obtain interpretable, physics-based machine learning algorithms for meteorological applications; (2) illustrate these methods for the detection of overshooting top (OTs) in satellite imagery.

Specifically, we seek to simplify the process of OT detection by first using mathematical methods to extract key features, such as cloud texture using Gray-Level Co-occurrence Matrices, followed by applying an EBM. Our EBM focuses on the classification task of predicting OT regions, utilizing Channel 2 (visible imagery) and Channel 13 (infrared imagery) of the Advanced Baseline Imager sensor of the Geostationary Operational Environmental Satellite 16. Multi-Radar/Multi-Sensor system convection flags are used as labels to train the EBM model. Note, however, that detecting convection, while related, is different from detecting OTs.

Once trained, the EBM was examined and minimally altered to more closely match strategies used by domain scientists to identify OTs. The result of our efforts is a fully interpretable ML algorithm that was developed in a human-machine collaboration. While the final model does not reach the accuracy of more complex approaches, it performs well and represents a significant step toward building fully interpretable ML algorithms for this and other meteorological applications.

SIGNIFICANCE STATEMENT: The purpose of this work is to introduce the interpretable machine learning method of Explainable Boosting Machines (EBMs) to an atmospheric science audience by closely examining how they can be used to predict the location of overshooting cloud tops, which have been associated with the occurrence of severe weather, in satellite imagery. Interpretable machine learning methods are important in high-risk situations such as overshooting top detection as they allow forecasters to have a better understanding of how exactly a prediction is made. We walk through how to build, interpret, and modify the machine learning algorithm for use on this task, but the steps discussed can be generalized to any application.

1. Introduction

Explainable Boosting Machines (EBMs) are a simple and interpretable machine learning (ML) algorithm introduced by Nori et al. (2019) that have brought many advancements to medical, health care, and financial applications. EBMs are interpretable, meaning, by nature, their decision-making processes can be easily articulated and understood. In high-stakes fields understanding how a model arrived at its conclusions is paramount for avoiding harmful mistakes. ML models can suffer from problems such as encoded bias (Noiret et al. 2021; McGovern et al. 2024) and other faulty strategies (Lapuschkin et al. 2019; McGovern et al. 2022). Being able to observe and understand the underlying processes may help to alleviate such issues. This paper evaluates the use of EBMs, in combination with feature engineering, to obtain an interpretable, physics-based machine learning algorithm for a satellite imagery analysis application in meteorology.

a. Interpretable AI for Meteorological Applications

For high-stakes applications such as severe weather forecasting, forecasters need to trust and understand the models they use. A key part of building that trust is establishing their *failure modes*. Knowing in which situations a method might fail and what that failure might look like is an instrumental step in rolling out a model into operational use. While any ML model can be susceptible to adopting faulty strategies, it is very hard to detect these issues and understand the failure modes they lead to in typical complex, black-box ML models (Rudin 2019). Two broad methods for building understanding of ML models have been developed: Explainable AI (XAI) and Interpretable AI (Flora et al. 2023). XAI methods seek to create post-hoc explanations for

the behavior of a black-box model, and have been utilized in many atmospheric applications; see, e.g., McGovern et al. (2019); Toms et al. (2020); Bommer et al. (2024); Fan et al. (2024); Krell et al. (2024), and references therein. These methods, however, are generally limited to identifying strategies that a model employed for a *specific* sample, which makes establishing a broad understanding of all strategies and failure modes of a model extremely challenging.

An alternative to XAI is Interpretable AI, which entails utilizing ML pipelines that are inherently interpretable, i.e., models that make their strategies directly legible to users and as such do not require post-hoc explanations. This approach is advocated by Rudin (2019) and is the approach used here. Along the same guidelines, Hilburn (2023) attempted to build interpretability into their model by replacing a neural network with a combination of feature engineering (applying spatial derivatives and pooling operations to the inputs) and linear regression. This approach was able to reveal additional prediction strategies that were not obvious using XAI; however, the data preparation and model training were cumbersome using this approach.

In this research, we explore a different type of Interpretable AI method, namely the method of Explainable Boosting Machines (EBMs), inspired by a talk given by Caruana (2023) on the benefits of using EBMs. Motivated by this talk, we set out to explore whether EBMs can also be applied to meteorological applications. As a sample application, we chose to explore how EBMs can be used to predict the location of overshooting tops (OTs) in satellite imagery. This task has been taken on multiple times before (Bedka et al. 2010; Bedka and Khlopenkov 2016; Khlopenkov et al. 2021) including, more recently, by using complex, black-box machine learning models (Kim et al. 2017; Cooney et al. 2025). In this paper, we seek to leverage EBMs to develop a simplified approach. We first apply physics-based feature engineering to simplify the task, then apply EBMs using those features to gain a fully interpretable ML model.

By using such a simple ML method, we hope to achieve good performance—though perhaps not quite as good as that of complex ML methods—while gaining full interpretability, including the ability to identify the failure modes of the model. We emphasize that the goal of this study is not to deliver a new, state-of-the-art OT identification algorithm. Instead, our primary goal is to explore the ML technique of EBMs and its unique capabilities for meteorological applications. The application of OT identification is used only to demonstrate the step-by-step process of developing an EBM for such an application.

b. Sample Application - Detecting Overshooting Tops in Satellite Imagery

The importance of detecting and monitoring OTs is revealed through their connection with convective storms and, in general, the occurrence of severe weather. More specifically, satellite imagery has been used to connect OT signatures with the occurrence of severe weather events such as heavy rainfall, strong winds, hail, and tornadoes (Negri and Adler 1981; Heymsfield et al. 1991; Brunner et al. 2007; Dworak et al. 2012; Mikuš and Mahović 2013). Severe storms are an increasing contributor to billion-dollar disasters in the United States. In their original analysis, Smith and Katz (2013) found that of these billion-dollar disasters between 1980 and 2011, severe storms accounted for 10% of the cost and 32% of events. In a more recent analysis by NOAA National Centers for Environmental Information (NCEI) (2024), the contribution to the cost by severe storms has increased to 17% over the period 1980-2023, and severe storms are nearly 50% of the events with a strong increase in the severe storm count starting around 2006.

The most suitable way to observe OT location is via satellite. The latest generation of geostationary satellite imagers are ideally suited to this task. In this work, we utilize data provided by the Geostationary Operational Environmental Satellite (GOES) 16 satellite. The Advanced Baseline Imager (ABI) (Schmit et al. 2017) aboard GOES-16 has a spatial resolution ranging from 0.5 to 2 km and a temporal refresh rate ranging from 0.5 to 15 minutes. These scales are capable of observing features associated with OTs.

Reflectance data shown in Fig. 1a derived from visible (VIS) imagery shows cloud-top bubbling associated with OTs. Multi-spectral information enhances situational awareness associated with infrared (IR) imagery in Fig. 1b. Information from these two channels are combined by overlaying partially transparent, color-enhanced IR imagery at temperatures less than or equal to 250 K over VIS imagery to create a VIS/IR sandwich product, shown in Fig. 1d. At similar scales, shown in Fig. 1c, the "PrecipFlag" product from the Multi-Radar Multi-Sensor (MRMS) system can be used to provide insight into the location of convection.

c. Existing Approaches to Identify Overshooting Tops

Numerical thresholds are a classic scheme for isolating and identifying OTs. When using satellite data, thresholds derived from infrared imagery are most common as their usefulness is not limited by the time of day the image was taken (Kim et al. 2017). Fixed brightness temperature (Ai et al.

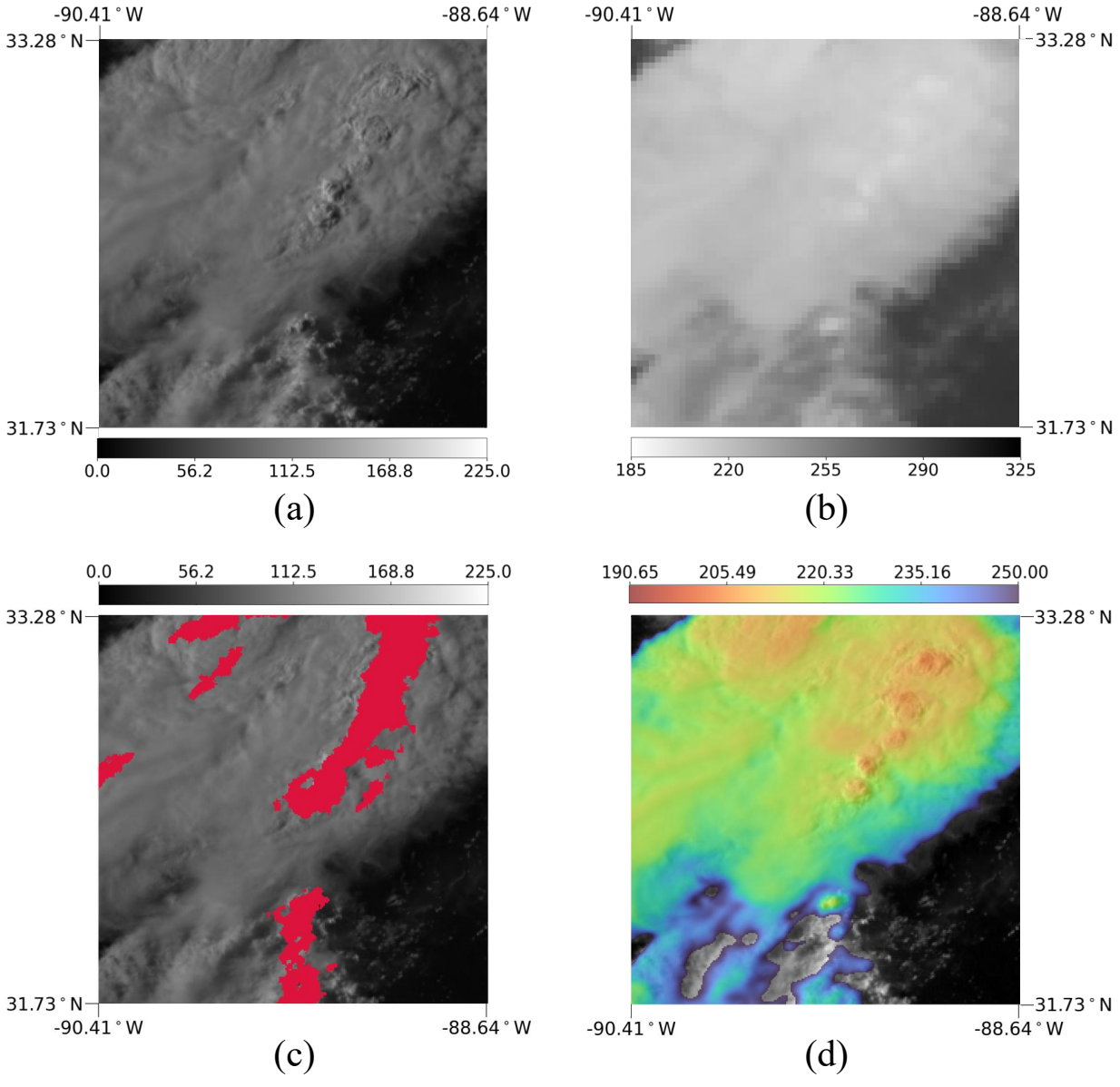


FIG. 1. Examples of (a) reflectance, (b) infrared, (c) VIS/IR sandwich product, and (d) MRMS scenes, each from 5 June 2024 at 21:45:00Z.

2017) and infrared window channel texture (Bedka et al. 2010) values, among others, can be used. The value, however, depends on the context (e.g., the season of interest), meaning there is not a one-size-fits-all value that can be chosen to reliably separate OTs from their surroundings in all cases.

Other approaches to OT identification have utilized ML techniques such as convolutional neural networks (CNNs), including work done by Kim et al. (2018). CNNs provide a natural choice for extracting information from satellite imagery because of their ability to learn complex statistical relationships and make use of the information content in gradients and, more generally, spatio-temporal patterns. More specialized CNNs designed for semantic segmentation, often using a U-net architecture, have been used in this domain. For example, Cooney et al. (2025) used the original U-net (Ronneberger et al. 2015), MultiResUnet (Ibtehaz and Rahman 2020), and AttentionUnet (Oktay et al. 2018) architectures to identify OTs within satellite imagery. These models, however, are black-box, and thus face the discussed limitations.

d. Proposed Strategies to Identify Overshooting Tops

Before building an interpretable ML model to automatically detect OTs from satellite-based imagery, we first consider how a human analyst, when presented with satellite imagery, might identify OTs. The product of strong atmospheric instability and subsequent vertical updrafts, OTs can be visually identified by considering the relationship they have with their surroundings. We present two primary strategies to identify OTs:

- Strategy 1 seeks to identify OTs by considering the elevation difference seen between them and the surrounding cirrus anvil clouds by numerically quantifying this displacement using a proxy such as cloud-top temperature.

- Strategy 2 involves numerically quantifying texture within regions (typically represented by small groups of pixels) of visible imagery to then isolate the bumpy, textured OTs from the surrounding flatter, less-textured anvil cloud.

In this work, we combine these two strategies by considering both cloud-top temperature derived from infrared imagery and texture derived from visible imagery. Additionally, we consider a measure of cloud-top brightness, making the observation that OTs are often visually brighter than the surrounding anvil from which they protrude. Considering brightness can also indicate the presence of a shadow cast by an OT, another easily identifiable feature of the phenomenon (Bedka et al. 2010).

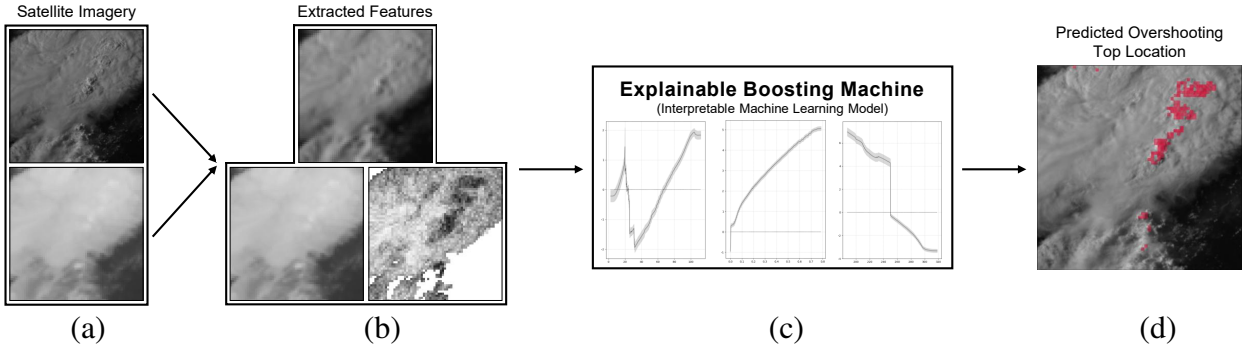


FIG. 2. Summarized approach displaying (a) the satellite imagery used (taken 5 June 2024 at 21:45:00Z), (b) the three extracted features, (c) the interpretable machine learning method used (EBM) and visualization of three of its learned strategies, and (d) a finalized prediction.

e. Toward an Interpretable Model to Identify OTs

We propose combining the steps outlined below—and illustrated in Fig. 2—to achieve an OT identification algorithm that is both simpler and fully interpretable:

1. Use GOES-16 ABI visible and infrared imagery (Fig. 2a),

2. Derive information-rich features using (a) standard image processing techniques and (b) the mathematical framework of Gray-level Co-occurrence Matrices (GLCMs) to extract texture information using the approach by Moen (2024) (Fig. 2b and Section 2c),

3. Combine extracted features using the interpretable ML algorithm Explainable Boosting Machines (Fig. 2c) to identify the location of OTs (Fig. 2d).

f. Introducing Explainable Boosting Machines

EBMs have found frequent use in the health sciences, see for example Morgan et al. (2021); Sarica et al. (2021); Hegselmann et al. (2022); Wang et al. (2022); Patel (2023); Körner et al. (2024); Arslan et al. (2024); Wang et al. (2024); Yagin et al. (2025) and many other studies. We, however, found only one set of studies that used EBMs for meteorological applications, in which they were used to estimate wind shear for aviation tasks (Khattak et al. 2023a,b). These wind shear studies focus primarily on EBM application, and as such do not provide a thorough tutorial on the theory and implementation of EBMs. Moreover, they do not make use of the ability to edit EBM

models after training, nor do they incorporate spatial context via feature engineering. Elsewhere in environmental science, EBMs have been used to anticipate landslides (Caleca et al. 2024), while in ecology they have been used to estimate the date of leaf unfolding (Gao et al. 2024) and in agriculture to predict crop yield (Celik et al. 2023; Pant et al. 2025) and pests (Nanushi et al. 2022).

Given the successful use of EBMs in a variety of fields, we now seek to motivate their use in atmospheric science by first explaining their core concepts and then applying them to the problem of detecting OTs. EBMs can be thought of as an extension of the classic linear regression scheme,

$$Y_i = \beta_0 + \sum \beta_i X_i + \epsilon_i, \tag{1}$$

where the response Y_i is governed by an intercept β_0; linear functions β_i's, each corresponding to a predictor X_i; and an error term ϵ_i. When multiple features are being considered, each feature gets its own β_i, i.e., its own slope. Thus, we can view the linear regression model as a sum of linear functions of the predictors. Similarly, EBMs can be viewed as the sum of functions. These functions, however, are not restricted to being linear. Instead of a slope, each feature is associated with a non-linear function that describes the relationship between the feature and the outcome of interest. These are known as feature functions and, within the EBM framework, are typically restricted to being either univariate (1D; in the case of a main effect) or bivariate (2D; in the case of a pairwise interaction) (Lou et al. 2013). As a result, EBMs are highly intelligible.

In contrast, many state-of-the-art black-box models are unintelligible; i.e., the ways in which the model arrived at its conclusions remain unknown to the users of the model. To illustrate how other models differ from EBMs in both the model development and model assessment processes, we present two workflows in Fig. 3, one for conventional ML models (such as CNNs or random forests), Fig. 3a, and one for EBMs, Fig. 3b. As Fig. 3a emphasizes, a common loop for other black-box ML applications involves evaluating and retraining models. Caruana et al. (2015) refer to this behavior as "repairing" the model. Hyperparameters may be adjusted or the data may be altered, but, in order to *change* the model, a new model must be trained. Even then, given these models are black-box, there is no way to guarantee alterations had the intended effect.

In contrast, because EBMs are glass-box (and, more specifically, modular), their strategies can be examined directly (Caruana et al. 2015). Even more importantly, the EBM's learned strategies themselves can be directly edited (Caruana et al. 2015) to correct any undesired behavior. This

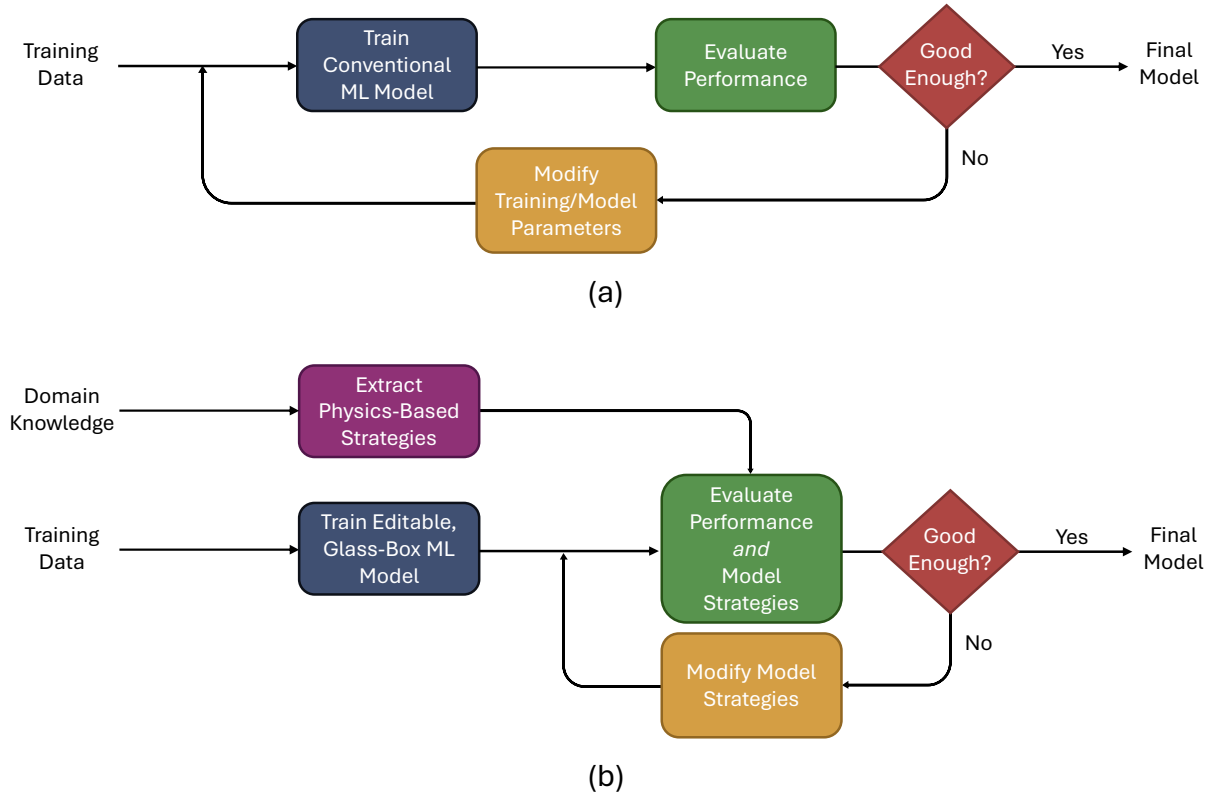


FIG. 3. Examples of two ML workflows—(a) for black-box ML models, such as neural networks, and (b) for editable glass-box ML models, such as EBMs. Note that in (b) the model's strategies are modified without re-training the model.

additional capability of EBMs sets the stage for a new loop, one that does not include retraining the model at all. In this new loop, shown in Fig. 3b, the model's strategies can be directly edited and the performance can be observed after each change with alterations occurring until a desired model—with acceptable strategies and adequate performance—has been constructed.

As such, the way in which the EBM can be updated is more direct than the ways in which conventional ML models are updated. In this work, we explore how domain knowledge, more specifically knowledge of the physics-based strategies related to detecting OTs, can be used to inform changes made to a fully trained model.

2. Data and Feature Extraction

The dataset used in this application was created in-house and was generated to be similar to the dataset used by Lee et al. (2021a). For our approach, it was separated into (1) a training set—to train the EBM, (2) a validation set—to examine how the model performed and to inform changes to be made to the learned strategies, and (3) a test set—to test the final model on unseen data. These sets were comprised of "scenes," each scene being 64×64 (4,096) pixels. In total, we had access to 10,404 scenes—5,206 for training, 2,579 for validation, and 2,619 for testing.

Data were collected over the central and eastern parts of the contiguous United States. As is common in earth science applications, our training, validation, and test datasets came from different years to avoid the impact of significant autocorrelation between samples in the three sets. The training set spans from May to August of both 2021 and 2022, the validation set from May to August of 2023, and the test set from May to August of 2024. These sets contain scenes that represent multiple time steps corresponding to several reported storm outbreaks. The training set corresponds to 242 distinct events, the validation set 118, and the test set 123. Given these data were collected only during the northern hemisphere's summer months, a limitation of the proposed algorithm is that it has not seen data from other time periods or locations. Performance may suffer if it encounters such data.

Satellite imagery was obtained from GOES-16's ABI, more specifically channel 2 (0.64 μm) in its native 0.5 km resolution and channel 13 (10.3 μm) in its native 2 km resolution. Convective flags from data provided by the MRMS system were used as labels to train the model. How information provided by the MRMS system was used to generate these labels is discussed in Section 2e.

Because our OT identifier uses visible imagery, it is more limited in scope than other state-of-the-art satellite-based identifiers as, in order for visible imagery to be useful, it must have been taken during the day. To further ensure the visible information was useful, any images taken when the solar zenith angle was over 65° were removed from the dataset following choices made by Lee et al. (2021a).

a. Developing Information-Rich Input Features for Overshooting Top Identification

EBMs have been designed to provide interpretable ML algorithms for limited tasks, usually tasks that involve scalar values as predictors. Thus, to use this approach for our application, we

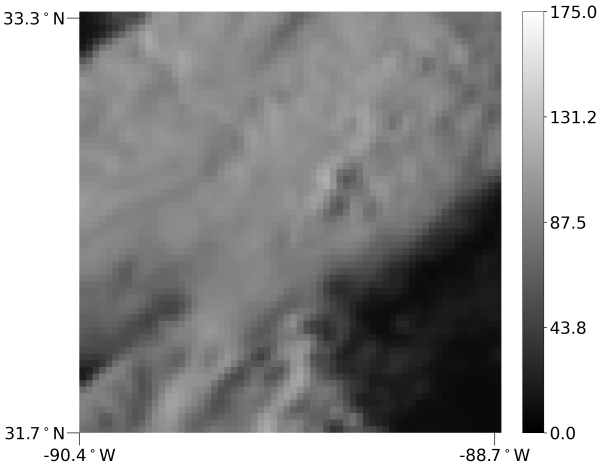


FIG. 4. Brightness feature example derived from channel 2 of GOES-16's ABI (taken 5 June 2024 at 21:45:00Z) which has a native resolution of 0.5 km and has been interpolated to a 2 km grid.

had to build a bridge, namely we had to first extract scalar features from the imagery that have high information contents regarding the desired task. In total, we extracted three features from the aforementioned satellite imagery. While temporal information from image *sequences* may be useful for detecting OTs (as Lee et al. (2021a) used temporal information to improve their convection detection), for this first study we only leveraged spatial features.

Below we define the features we extracted. Though we present these features using "scenes," we stress that the EBM algorithm does not treat these data as coming from any human-defined collection and only handles single-pixel "scalar" values.

b. Brightness Feature

The first feature, x_1, was derived from channel 2 imagery. Channel 2, also known as the "red" band, provides high-resolution visible imagery. Visible imagery provides a measure of the intensity of the light being reflected back to the satellite by the clouds and the surface being imaged. For this application, following choices made by Lee et al. (2021b), raw radiance values were converted into reflectance factor values using the κ factor then were normalized using the solar zenith angle to get reflectance values. An example of the reflectance data used can be seen in Fig. 1a. The high spatial resolution of the reflectance data allows for detailed textural information to be encoded. For x_1, the textural component was "removed" using a 9×9 box blur on every scene within each

dataset. The resulting data were then resized from their original resolution, 0.5 km, down to a coarser resolution, 2 km, to match the resolution of the subsequent three features. Nearest neighbor interpolation was used for this process. The result was a feature that captured overall brightness in larger regions. To delineate this information from the reflectance data, x_1 will be referred to as the "brightness" feature. An example of the brightness feature can be seen in Fig. 4. This scene is the "blurred" and resized version of the reflectance data seen in Fig. 1a.

c. Cool Contrast Tiles Feature

With advancements in the spatial and temporal resolution of data produced by state-of-the-art satellite systems, such as the GOES system, the use of visible imagery to (1) identify texture then (2) inform the identification of OTs is a logical sequence. As such, our next feature, x_2, was derived by extracting texture information from high-resolution visible We utilized the mathematical framework of Gray-Level Co-occurrence Matrices (GLCMs) for this process.

GLCMs capture the spatial relationship between pixel intensities in an image (Haralick et al. 1973). A GLCM is an $N \times N$ matrix where each entry $p(i, j)$ represents the number of times that a pixel with gray level i is adjacent to a pixel with gray level j. The number N is the range of gray levels in the image. Adjacency is defined by distance and direction—horizontal, vertical, or diagonal. We computed GLCMs for distance one (i.e., touching pixels) across all angles resulting in 8 neighboring pixels for every interior pixel. The matrices were then averaged and normalized. As one GLCM tile was calculated for every 4×4 pixel region, a reduction in resolution by a factor of four and a decrease in resolution from 0.5 km to 2 km was seen.

Once a GLCM has been calculated, various statistical measures can be derived. Following the work of Moen (2024), we determined the contrast statistic to be most relevant to our work. The contrast statistic sums the matrix entries weighted by the squared difference between gray levels.

Formally, we define the contrast statistic as:

$$f_{contrast} = \sum_{i,j=0}^{N-1} p(i, j)(i - j)^2 \tag{2}$$

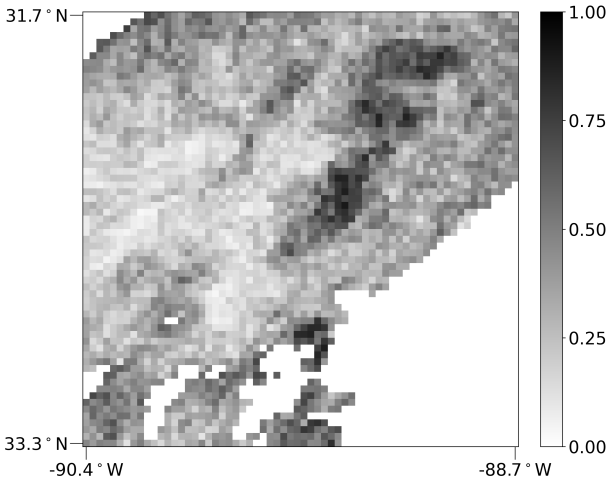


FIG. 5. Cool contrast tiles feature example derived from channel 2 of GOES-16's ABI (taken 5 June 2024 at 21:45:00Z) which has a native resolution of 0.5 km and has been interpolated to a 2 km grid. Darker values, those close to one, represent regions with the greatest amount of texture. Lighter values, those close to zero, represent regions with the least amount of texture. Values of exactly zero represent spatial regions corresponding to temperatures greater than 250 K.

In this statistic, the entries of the GLCM are weighted by $(i-j)^2$, which is larger the further i and j are from each other. Thus, when pixels are adjacent which have very different pixel values, $f_{contrast}$ take a larger value.

In particular, the diagonal entries (i.e., representing adjacent pixels with the same value) are assigned a weight of zero. Intuitively, higher contrast values indicate greater gray-level dissimilarity between adjacent pixels, highlighting areas of strong contrast in the image. Though these contrast values have a natural lower bound, 0, they do not have a natural upper bound. The solution to this problem was twofold. First, because the lack of an upper bound resulted in right-skewed tile values, the log of these tiles (plus one) was taken. Second, the tiles were normalized to the 0-1 range based on the largest contrast value found in the training dataset.

Finally, after a grid of contrast tiles had been calculated for each scene, local brightness temperature values provided by channel 13 of GOES-16's ABI were used to create the second feature, x_2, the "cool contrast tiles" feature, seen in Fig. 5. To create x_2, tiles that corresponded to brightness temperature values at or below 250 K, i.e., tiles that were relatively "cool," were retained and all other tiles, those that corresponded to "warm" temperatures, were set to 0. This modification acts

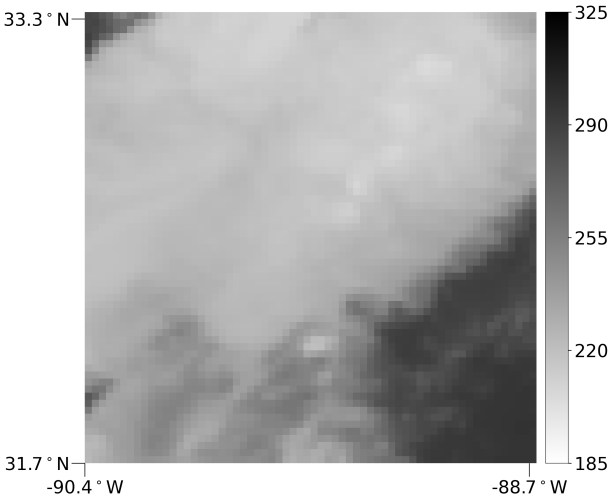


FIG. 6. Infrared feature example derived from channel 13 of GOES-16's ABI (taken 5 June 2024 at 21:45:00Z) which has a native resolution of 1 km and has been interpolated to a 2 km grid.

as a high-cloud mask. In the regions where there are no high clouds (i.e., regions of either low clouds or ground), the relationship between texture and OTs is already known and is thus not of interest. In each case, we do not expect to find an OT. Thus, the threshold limits the scope on which we draw inference to areas where we expect to find OTs want to understand the strategy. Similarly to Lee et al. (2021b), we chose a generous (quite warm) brightness temperature at which to threshold. Though OTs are not expected at such warm temperatures, such a low threshold allows for more flexibility during the model alteration process.

d. Infrared Feature

The final feature, x_3, was derived from channel 13, the "clean" IR longwave window band. Infrared imagery provides information related to surface and cloud-top temperature by measuring the intensity of emitted radiation and converting it into a temperature. In our work, for regions where clouds are present, we use this as a proxy for cloud top height, with higher clouds generally having lower cloud top temperatures and thus lower brightness temperatures. We consider temperature in Kelvin. A scene of the infrared feature can be seen in Fig. 6.

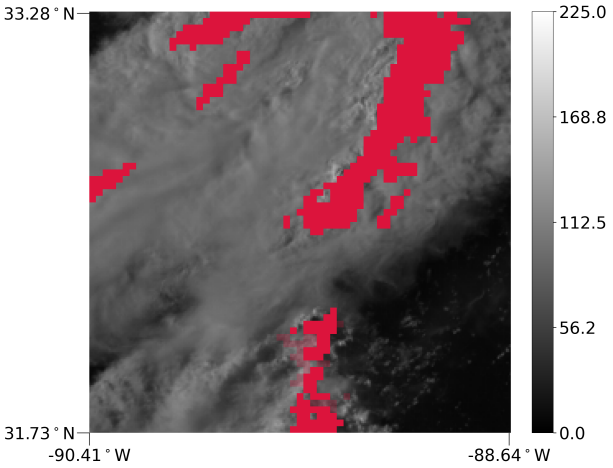


FIG. 7. MRMS system labels example on top of reflectance imagery, each taken 5 June 2024 at 21:45:00Z.

e. Multi-Radar Multi-Sensor Data as Convection Labels

To train a ML model to identify OTs, labels that depit their locations are desirable. One way to achieve this is by identifying echo tops, the highest areas at which precipitation can be detected above the tropopause, and using them as a stand-in for OTs. Not all OTs are able to penetrate the tropopause, however. When they do, echo tops do not always result in accurate OT outlines (Cooney et al. 2025). To get a more accurate sense of OT location, human analysts should be brought in and asked to label the scene by hand. As described by (Cooney et al. 2025), who chose this route, this process can be very strenuous. As such, we used labels that were available to us in-house but were not perfect for the task at hand, namely convection labels provided by the MRMS system, created following choices made by Lee et al. (2021b).

Developed by NOAA National Severe Storms Laboratory, the MRMS system integrates data from ground-based radars, surface and upper air observations, lightning detection system, satellite observations, and forecast models to generate high-resolution meteorological fields. It is designed to support real-time decision-making for severe weather monitoring, aviation operation, and hydrological forecasting. The system provides various fields such as radar reflectivity, precipitation rates, and hydrometeor classification at 1 km spatial resolution and two minute temporal resolution. In this study, the "PrecipFlag" product was used. This product categorizes surface precipitation type into seven categories: warm stratiform rain, snow, convection, hail, cool stratiform rain, tropical stratiform rain, and tropical convective rain. Among these categories, grid points labeled as convection, hail, and tropical convective rain were collectively defined as "convective" grid points.

Once rendered, the labels received a reduction in resolution from 1 km to 2 km. Nearest neighbor interpolation was used for this process. Additionally, a temperature threshold of 250 K according to the IR imagery was implemented to remove small, isolated regions of convection so the labels more closely matched the task at hand. Fig. 7 shows reflectance imagery with the corresponding convection labels overlaid. The tiles at 100% opacity correspond to detected convection in regions at or below 250 K while the tiles at 50% opacity correspond to detected convection above 250 K (and was thus set to 0 for model training).

f. Identifying Convection vs. Identifying Overshooting Tops

Despite the minor alterations, the labels are not perfect for identifying OTs. Though they typically cover the proper location of OTs, as convection is a necessary precursor for OTs, they also contains many "extra" indications corresponding to convective regions that did not result in an OT. Because OTs are a subsection of the overall "convective" class, however, the MRMS data contains useful information. This information can be leveraged to train a model but, because EBMs can be altered, does not need to be what the final model ends up predicting. This process can be thought of as another way to accomplish the overall objective of transfer learning. Instead of finding training labels for our task, we aimed to shift from identifying convection to identifying OTs during model development using the following three modifications:

1. **Feature engineering for OTs:** Selecting only features that focus on identifying OTs, in particular cloud texture, and processing them to emphasize properties that are unique to OTs;
2. **Modification of training data:** Applying the 250 K temperature threshold to alter the convection labels;
3. **Modification of trained model:** Altering the learned strategies of the trained EBM.

Note, however, that as the MRMS data is for convection, it is not immediately suitable to assess accuracy of our algorithm for OTs. See the discussion of this topic in Section 4.

3. Model Development

We approached the identification of OTs as a pixel-wise binary classification problem. When applied to image data, this can be interpreted as a form of image segmentation. We considered

how the interpretability and modularity of EBMs can be used to incorporate domain knowledge into the model. As outlined in Fig. 3b the development of an EBM model consists of first training a model, then visualizing and modifying its strategies as needed based on feedback from domain scientists. We explore that process in this section. The performance of the final model is discussed in Section 4.

a. Understanding Explainable Boosting Machines

The key to understanding our approach comes from an understanding of EBMs which, in turn, comes from a close examination of the formula that describes them. To start, however, we note that EBMs were built upon the framework of Generalized Additive Models (Hastie and Tibshirani 1987) and were designed to be an improvement upon them (Lou et al. 2013, 2012). As such, EBMs are additive models. Keeping this in mind, as outlined by Lou et al. (2013), EBMs take the form:

$$g(E[y]) = \beta_0 + \sum f_i(x_i) + \sum_{i,j \in I} f_{i_j}(x_i, x_j), \tag{3}$$

where $f_i(x_i)$ denotes a univariate (1D; "primary") feature function representing the contribution of feature x_i to the outcome of interest and $f_{i_j}(x_i, x_j)$ denotes a bivariate (2D; "interaction") feature function representing the contribution of a pair of interacting features, (x_i, x_j), to the outcome of interest. The GA^{2}M algorithm, which EBM is an implementation of (Nori et al. 2019), learns the main effect feature functions first using bagging and gradient boosted trees (Caruana et al. 2015). During the boosting process, only one feature function is trained at a time in a "round-robin" fashion to appropriately deal with features that are potentially linearly dependent and to ensure feature functions that show the appropriate modular relationship with the response are trained (Nori et al. 2019). Additionally, a low learning rate is used to ensure that the order in which the features are included and thus the order in which the functions are learned has no impact on the learned model (Nori et al. 2019).

After the main effect feature functions have been trained, the algorithm detects then ranks pairwise interactions in the residuals. Cross-validation is used to select the (limited) number of pairwise interactions to be included in the model (Caruana et al. 2015). More information on this algorithm can be found in Lou et al. (2013). Here, we represent this set with I. Though it is possible to

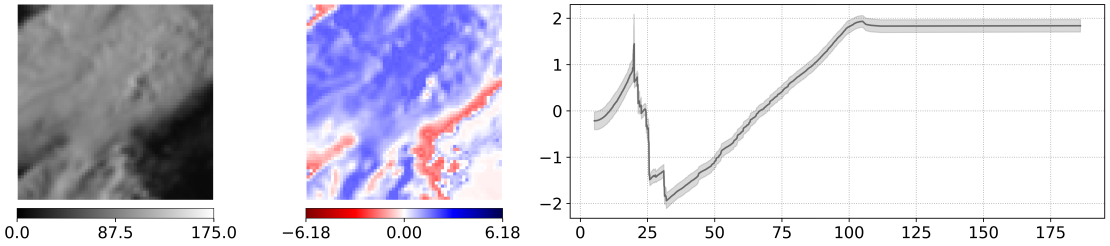


FIG. 8. (a) Brightness feature, (b) feature importance, and (c) corresponding feature function mapping pixel values from (a) to pixel values in (b) with shaded error bars to represent a measure of uncertainty.

manually include higher-order interactions within the model, they are not automatically included. We chose to not specify any high-order interactions to achieve a desired level of interpretability.

Finally, g represents the chosen link function which allows for tasks such as regression or classification to be performed (Nori et al. 2019), and β_0 represents the model's learned intercept. Though the intercept of an EBM can be altered, such an alteration was not explored in this work.

b. Visualizing EBM Strategies—Plotting Feature Functions and Feature Importance

By including three features—the brightness, cool contrast tiles, and infrared features—our model learned three primary feature functions. Only three pairwise interactions are possible between three features and all were automatically included as interaction feature functions.

Following the training of a model, the next step was to examine the learned strategies. To start, we examined the primary feature functions. To demonstrate this process, we examine, in detail, the brightness feature function as seen in Fig. 8c. The x values of the plot represent the input to the function (here, the values of the brightness feature found within the training set), and the y values represent the output of the function (here, something called the "score"). The score values represent the log-odds (Caruana et al. 2015) of any given brightness value being convective but can be thought of more intuitively as a way to quantify the strength of the association between the input and the output. Positive scores indicate a positive relationship between brightness and convection and negative scores a negative one. The magnitude of the score, then, indicates the strength of the relationship.

The actual learned relationship between brightness and convection can be seen in the dark line featured within Fig. 8c, the semi-transparent borders representing an automatically generated estimation of the "error" garnered during the EBM model fitting process Caruana et al. (2015).

Fig. 8 also displays a scene of our brightness feature (Fig. 8a). We use this scene to motivate the plot in Fig. 8b. This plot displays the score value associated with every pixel from 8a. In other words, 8b is the result of applying the feature feature function shown in 8c to the values of the scene shown in 8a. This plot is referred to as the "feature importance." The "red" portions represent areas of the scene negatively associated with convection and the "blue" portions areas positively associated with convection. The darker the value of the shade, the "stronger" the relationship.

By viewing all three scenes in Fig. 8 together, we observe that, in general, the model learned that low brightness levels are associated with non-convection and that high brightness levels are associated with convection.

c. Model Alteration—Primary Feature Functions

Once visualized and understood, the next step in our process was to consult domain scientists regarding the learned relationships to inform model alterations. In addition to displaying the functions themselves, satellite imagery and the derived features, as well as maps of the feature importance, were used to identify "faulty" strategies. To demonstrate this process, we once again turn to the brightness feature and feature function, noting that alterations were not inspired by this scene nor any other scene from the test set. Scenes from the validation set were used for this process.

To start, we examine the "spike" seen in the feature function at low brightness levels. Though we theorized this spike was due to the brightness values of shadows, further examination showed the spike actually corresponds to regions of low-level, small clouds that were blurred and anything darker, but, typically, not on-cloud shadows. To see this, Fig. 9c shows the spike highlighted in two different colors, one for negative score values (yellow) and one for positive score values (red). The corresponding brightness values are then highlighted in the same color in Fig. 9b. Fig. 9a acts as a reference.

After examining a similar map on many scenes, the decision was made to flatten out the left-hand side of the feature function to match the lowest score value seen. This alteration and the impact it

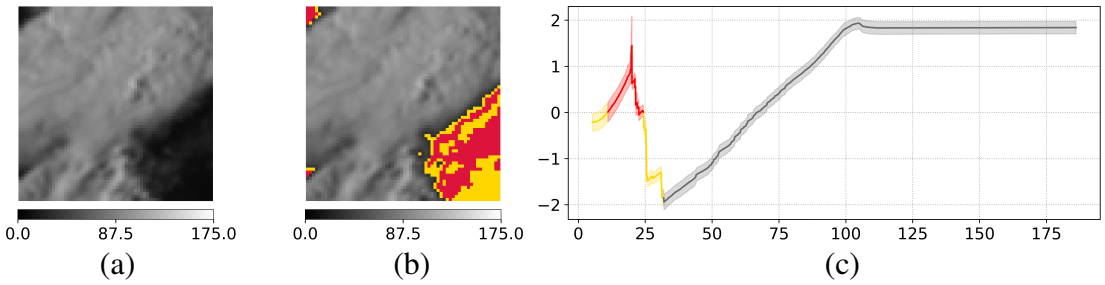


FIG. 9. (a) Brightness feature, (b) feature importance, and (c) corresponding feature function mapping pixel values from (a) to pixel values in (b) with shaded error bars to represent a measure of uncertainty. Brightness values between 11 and 24 have been highlighted in red on both (a) and (c).

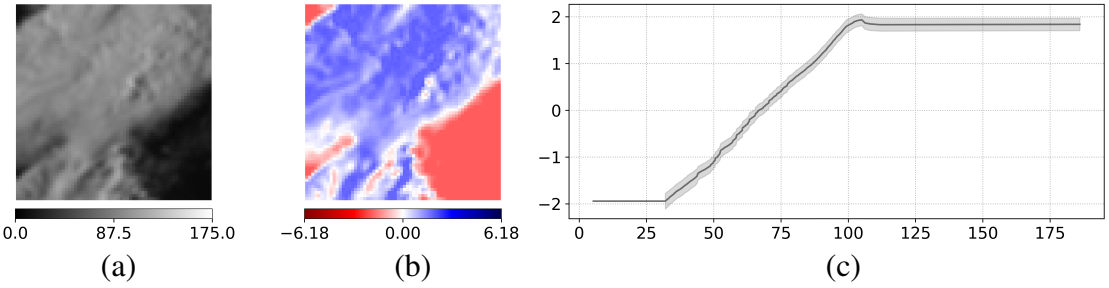


FIG. 10. (a) Brightness, (b) feature importance, and (c) corresponding (altered) feature function mapping pixel values from (a) to pixel values in (b) with shaded error bars to represent a measure of uncertainty. Error bars have been removed from regions of the feature function that were altered.

had on the map of feature importance can be seen in Fig. 10. Fig. 10 shows the same three plots as Fig. 8. In Fig. 10c, however, the error bars have been removed from where the feature function received alterations.

Next, we consider all three primary feature functions. The first row of Fig. 11 displays the original primary feature functions and the second row the edited primary feature functions. Of the three, we altered two—the brightness feature function, as described above, and the cool contrast tiles feature function. The cool contrast tiles feature function was scaled to allot more "importance" (potentially larger score values) to the texture within each scene and then shifted downward to ensure some smaller texture values received small, and even negative, score values. The effect the alteration had can most easily be seen by viewing a map of feature importance, which is shown in Fig. 12a.

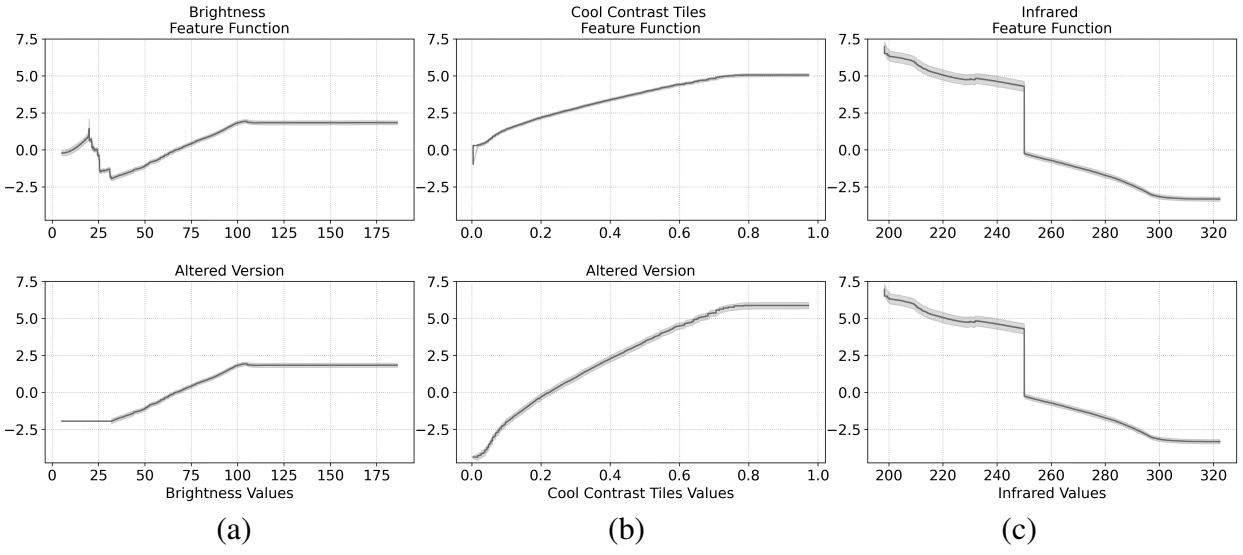


FIG. 11. Unedited and edited feature functions for all main effect features, (a) brightness, (b) cool contrast tiles, and (c) infrared—unedited functions in the first row and edited functions in the second. Only feature functions (a) and (b) were altered.

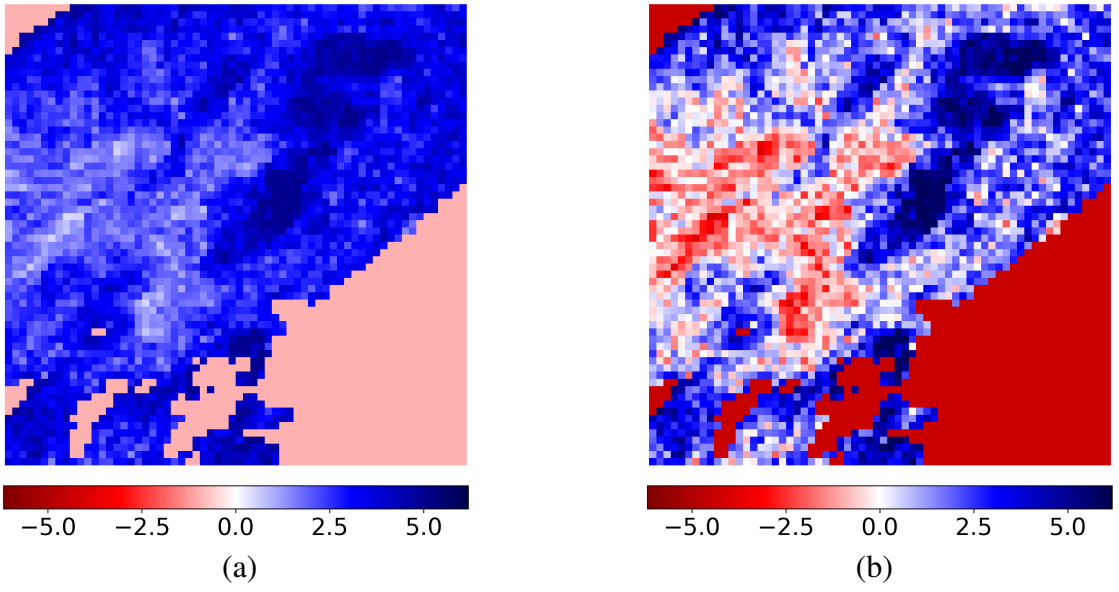


FIG. 12. Map of feature importance for the Cool Contrast Tiles (a) before the model was altered and (b) after the model was altered.

While the original feature function assigned positive score values to every texture value, the altered function takes a more nuanced approach. Smaller texture values receive a low, or even negative score, and large texture values receive larger, positive scores.

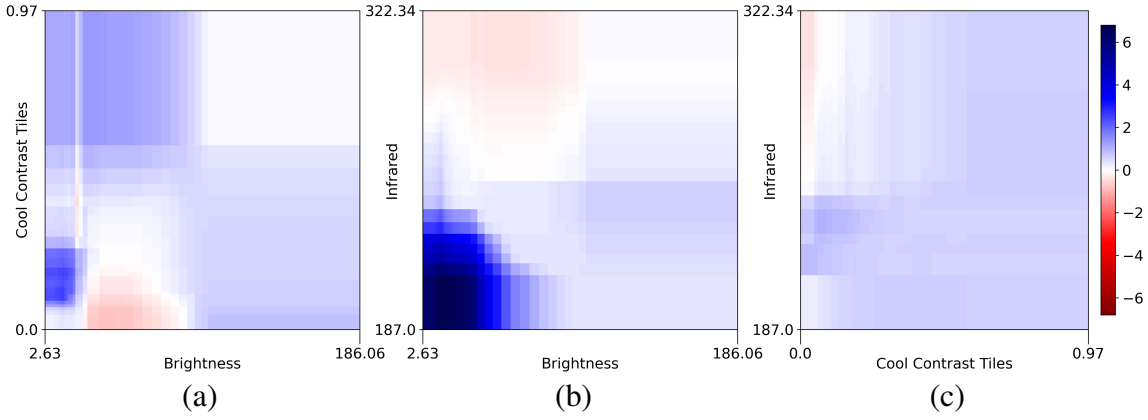


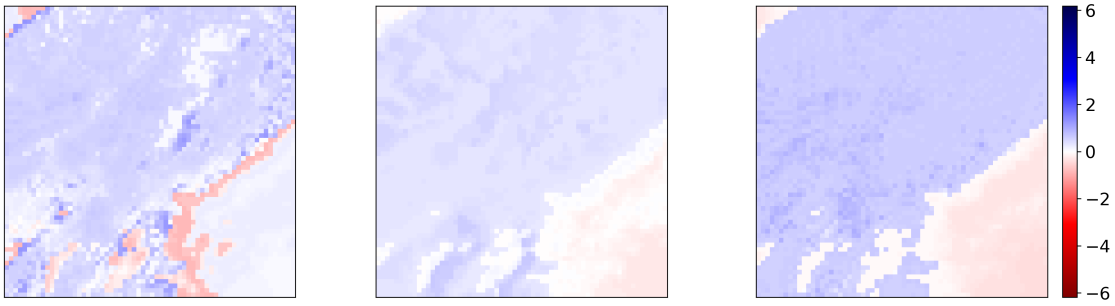
FIG. 13. Feature functions for the three included interactions between (a) the brightness and cool contrast tiles features, (b) the brightness and infrared features, and (c) the cool contrast tiles and infrared features.

d. Model Alteration—Interaction Feature Functions

Of the three possible pairwise interactions, all three were automatically included in the model. These were between the following pairs of features: brightness and cool contrast tiles, brightness and infrared, and cool contrast tiles and infrared. Each pairwise interaction feature function is represented as a heat map, where each (x,y) pair gives the score value for that combination of feature values. Fig. 13 displays the three interaction feature functions in the order outlined above. None of the three interaction feature functions were altered.

As outlined, the interaction feature functions are trained in the residuals and *after* the primary functions have been trained. Essentially, the role of the interaction feature functions is to "tie up" any potential lose ends. As such, there is less pressure for these functions to be as "neat" as the primary functions, so to speak. The result of this is that many of the strategies seen in these feature functions may not make as much "physical" sense as the strategies seen in the primary feature functions.

Despite this, some interesting information can be extracted by examining the maps of feature importance corresponding to each interaction feature function seen in Fig. 14. Maps of feature importance for pairwise interactions are similar to the maps of feature importance for the primary features. When considering a pairwise interaction map of feature importance, however, the score value seen comes from mapping one value from each of the two features that make up the interaction to a score as the combination of two inputs maps to a single score value output.



(a) (b) (c)

FIG. 14. Maps of feature importance for the interaction feature functions for the interactions between (a) the brightness and cool contrast tiles features, (b) the brightness and infrared features, and (c) the cool contrast tiles and infrared features.

In particular, the first (Fig. 13a) pairwise interaction seems to have attributes that, while seemingly counterintuitive, make the model more robust and help to aid predictions in areas where the primary features cannot.

When examining the interaction between the brightness and cool contrast tiles features, it can be seen that the upper-left corner is blue, corresponding to positive scores. This implies that regions of low brightness and high texture are associated with convection. We confirmed, using processes similar to those described in Section 3c, that these regions correspond to shadows. By comparing the visible imagery seen in Fig. 1a and the map of feature importance corresponding to this interaction feature function seen in Fig. 14a, the regions that appear the darkest blue near the middle of the scene in Fig. 14a correspond to the regions where OTs cast a shadow. It is important to note, however, that such a simple interaction cannot differentiate between shadows cast by and OT and any other shadow.

Looking to the interaction between the brightness and infrared features (Fig. 13b), a striking section of blue can be seen in the lower-left corner of the feature function. This implies that areas of low brightness and cold temperatures are receiving high, positive scores and indicate the presence of convection. Though this seemingly goes against what we have discussed, this section, too, can be attributed to helping to predict OT location in shadowed regions. An example of this can be found in the second case study presented in Section 4.

4. Results

This section focuses on the performance of our final (altered) EBM algorithm. The discussion is split into two parts—the first covers overall model performance metrics (and their limitations) and the second covers five case studies taken from the test set. Alternate model alterations for specific use cases that were developed but not included in the final model are explored as potential fixes for a class of identified failure modes.

a. Part One: Overall Model Results

Standard performance-based metrics using existing labels are typically an adequate way to assess how well a model performs on a given task. As discussed in Section 2f, however, our methodology is based on labels that mark convection rather than OT locations. Because of this, performance metrics using these labels illustrate how well the model is able to predict convection—a task it is not meant for—, rather than how well the model is able to predict OTs. Furthermore, predictions are being made on a fairly fine grid of 2 km, meaning some spatial mismatch is inevitable and not necessarily indicative of poor performance. Because of these limitations, metrics are presented for transparency but should not necessarily be used to judge model performance.

With 2,619 scenes used for testing, there are 10,727,424 pixels. When a prediction is made by the model, it is classified as one of the following: a "hit" if both the model and the convection labels indicated the presence of convection, a "correct rejection" if both the model and the convection labels indicated a lack of convection, a "false alarm" if the model indicated the presence of convection but the convection labels did not, and a "miss" if the model indicated a lack of convection but the convection labels did not.

The finalized model achieved 30,755 hits, 10,481,845 correct rejections, 32,424 false alarms, and 182,400 misses.

b. Part Two: Case Studies

We consider five cases and discuss them under the framework of OT identification. The first two cases presented represent scenes where we believe the model performed well at the given task. The third case represents a scene where the model performed well overall but still made some noticeable mistakes. The final two cases represent scenes where we believe the model performed

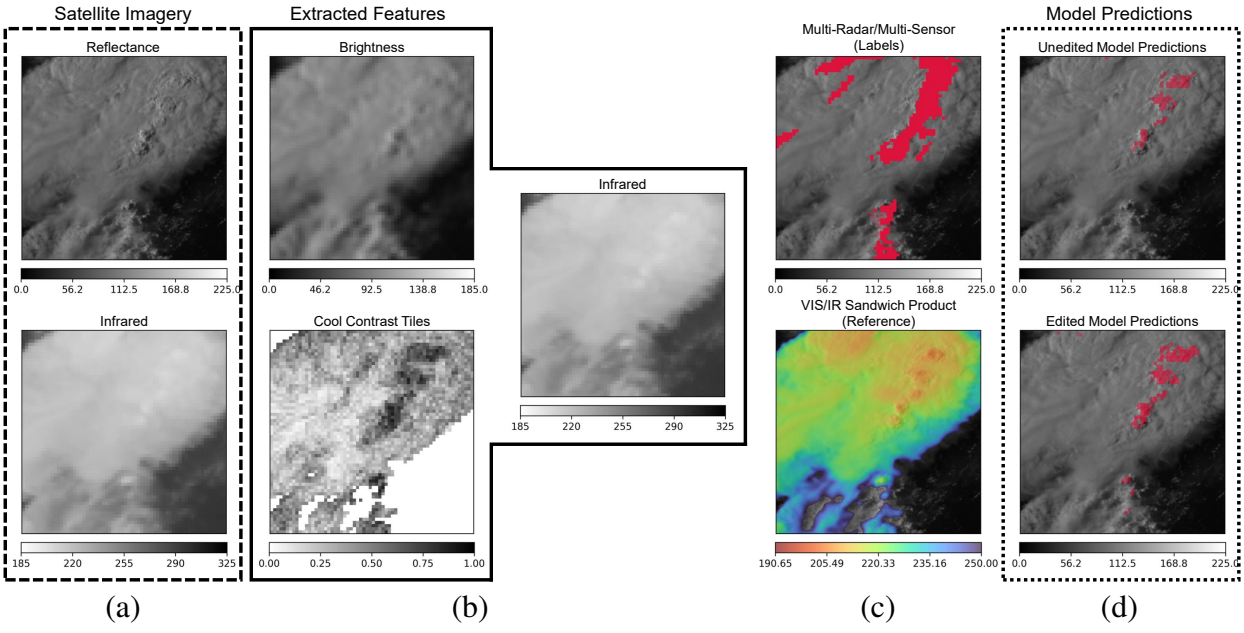


FIG. 15. EBM workflow stages for selected scene from the test dataset (taken 5 June 2024) including (a) satellite imagery (reflectance, infrared), (b) extracted features (smoothed brightness, cool contrast tiles, infrared), (c) MRMS labels and VIS/IR Sandwich reference, and (d) model predictions (from unedited and edited models).

poorly. The five cases were selected to illustrate interesting strategies of the final model, including its primary failure modes.

1) CASE I

The first case comes from imagery taken on 5 June 2024 at 21:45:00Z. Imagery from throughout this discussion has been combined and displayed in Fig. 15. Fig. 15a displays the satellite imagery used to derive the three features seen in Fig. 15b. The MRMS-derived convection labels can be seen in the first row of Fig. 15c. VIS/IR sandwich product imagery is displayed in the second row of Fig. 15c. This product is neither used for model development nor quantitative validation, and is used only to provide additional intuition about the scene. The predictions of the unedited model can be seen in the first row of Fig. 15d and the predictions of the edited model can be seen in the second.

Reflectance data shown in the first row of Fig. 15a shows five OTs in the upper-right corner. The VIS/IR sandwich product reflects this. Given the strong signal, the unedited model was able to

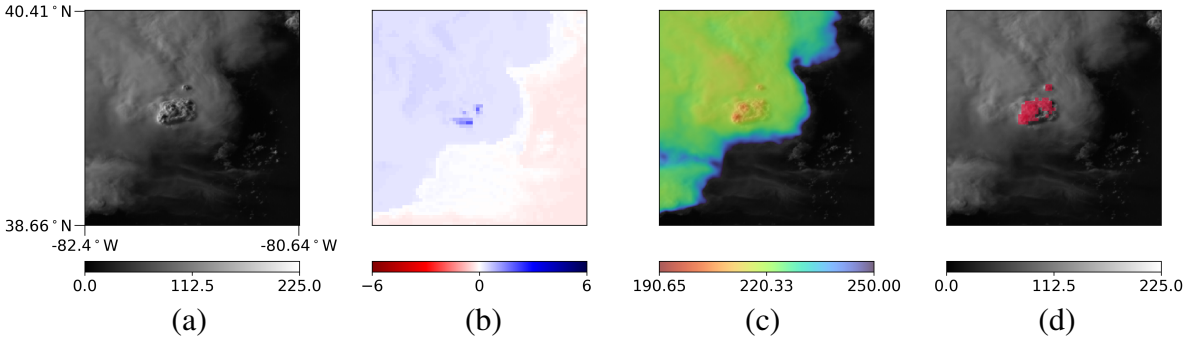


FIG. 16. Imagery taken 17 June 2024 including (a) VIS/IR sandwich product, (b) a map of feature importance corresponding to the interaction between the brightness and infrared features, and (c) predictions made by the edited model.

accurately detect their presence and predict their location. Alterations made to the model increased the strength of these predictions but also allowed for three potential OTs to be identified in the lower-middle portion of the scene. The signal these potential OTs provide is much weaker. Evidence that supports their status as OTs is not as strong in the VIS/IR sandwich product. Additionally, the model made two erroneous predictions in the upper-left corner of the scene where texture is minimal but temperatures are very cold.

2) CASE II

The second case comes from imagery taken on 17 June 2024 at 22:30:00Z. Fig. 16 displays the corresponding (a) reflectance, (b) map of feature importance corresponding to the interaction between the brightness and infrared features, (c) VIS/IR sandwich product, and (d) predictions made by the edited model. In Fig. 16a and c, the OTs can be seen almost directly in the center of the scene. They have a bubbly texture and are noticeably colder than their surroundings. The surrounding anvil is relatively flat and warm. As such, the EBM has no issue detecting their locations.

In this imagery, the direction of the sun causes much of the OT to be covered in shadow. Though the brightness feature function dictates dark values to not be associated with OTs, the EBM has no problem identifying OT locations. This is, in part, because of the interaction between the brightness and infrared features as seen in Fig. 13b. The associated feature function assigns large, positive scores to pixels that are both dark and cold, i.e., areas of shadow. This effect can be seen

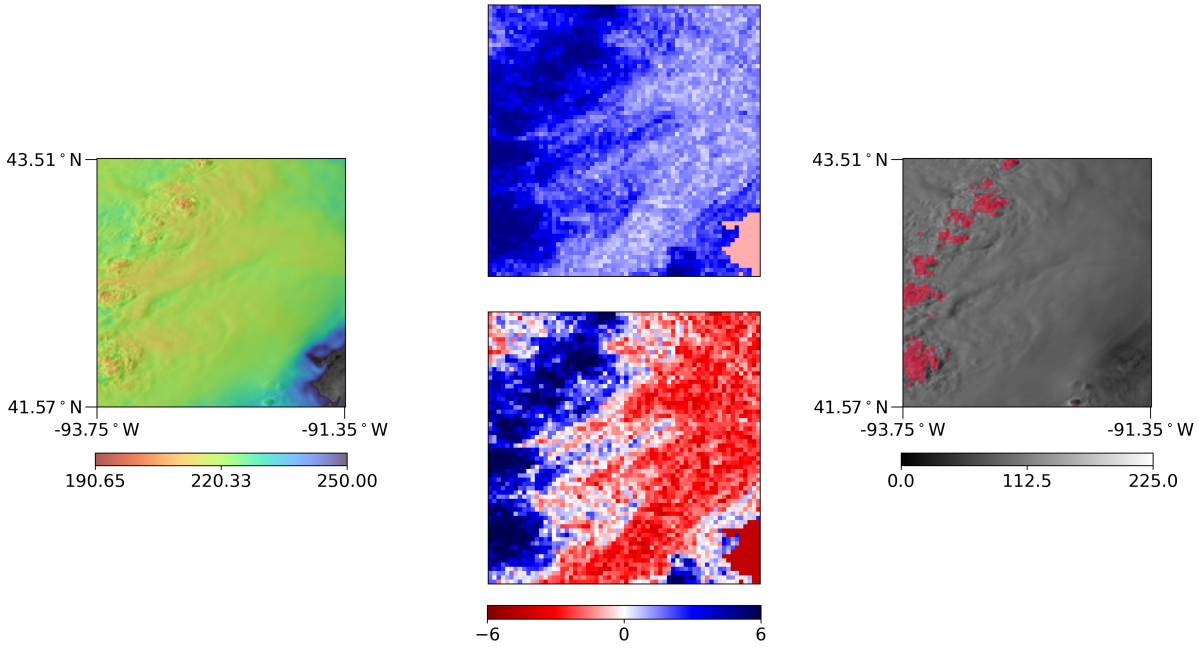


FIG. 17. Imagery taken 21 May 2024 including (a) VIS/IR sandwich product, (b) map of feature importance corresponding to the cool contrast tiles feature before (first row) and after (second row) the model was altered, and (c) predictions made by the altered model.

in the map of feature importance shown in Fig. 16b as the areas of shadow have been given large, positive scores.

3) CASE III

The third case comes from imagery taken on 21 May 2024 at 22:15:00Z. Fig. 17 displays the corresponding (a) VIS/IR sandwich product, (b) map of feature importance corresponding to the cool contrast tiles feature before (first row) and after (second row) the model was altered, and (c) predictions made by the altered model.

Around the center of each OT, the texture is notably "bubbly" and the temperature is cold. Though model alterations were effective in highlighting the regions featuring the most texture, as seen by comparing the first and second rows of Fig. 17b, such regions were not localized enough for acute OT detection, as seen in Fig. 17c. The predictions cover the locations of the OTs, but there are many false positives due to regions of cold temperature and high texture that do not correspond

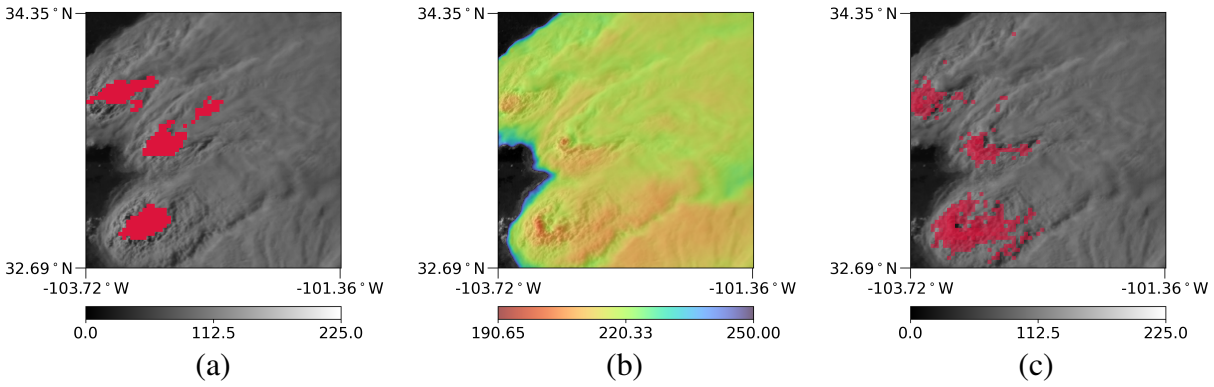


FIG. 18. Imagery taken 30 May 2024 including (a) MRMS labels, (b) VIS/IR sandwich product, and (c) predictions made by the edited model.

to OTs. The model's inability to distinguish between when this signal does and does not lead to an OT is the most common failure mode.

4) CASE IV

The fourth case comes from imagery taken on 30 May 2024 at 23:15:00Z. Fig. 18 displays the corresponding (a) MRMS labels, (b) VIS/IR sandwich product, and (c) predictions made by the edited model. This scene displays three potential OTs as seen in the VIS/IR sandwich product imagery. Surrounding each OT are features consistent with cold U/V shapes. Though associated with severe weather (Adler et al. 1985), detecting these shapes is not of interest for this application. Despite this, because such features are both cold and highly textured, the altered EBM classified them as OTs.

Because the EBM has only six scalar features, it is not possible to encode a way to differentiate between cold/textured regions that do and do not correspond to OTs. Attempts to dampen or heighten this signal result in the model failing to capture OT locations in other scenes. Despite model failure in this scene, the ability to visualize and alter the model's strategies allowed insight into the cause of the problem, allowing us to acutely identify and understand this failure mode.

5) CASE V

The fifth case comes from imagery taken on 3 June 2024 at 22:30:00Z. Fig. 19 displays the corresponding (a) VIS/IR sandwich product, (b) feature importance corresponding to the cool

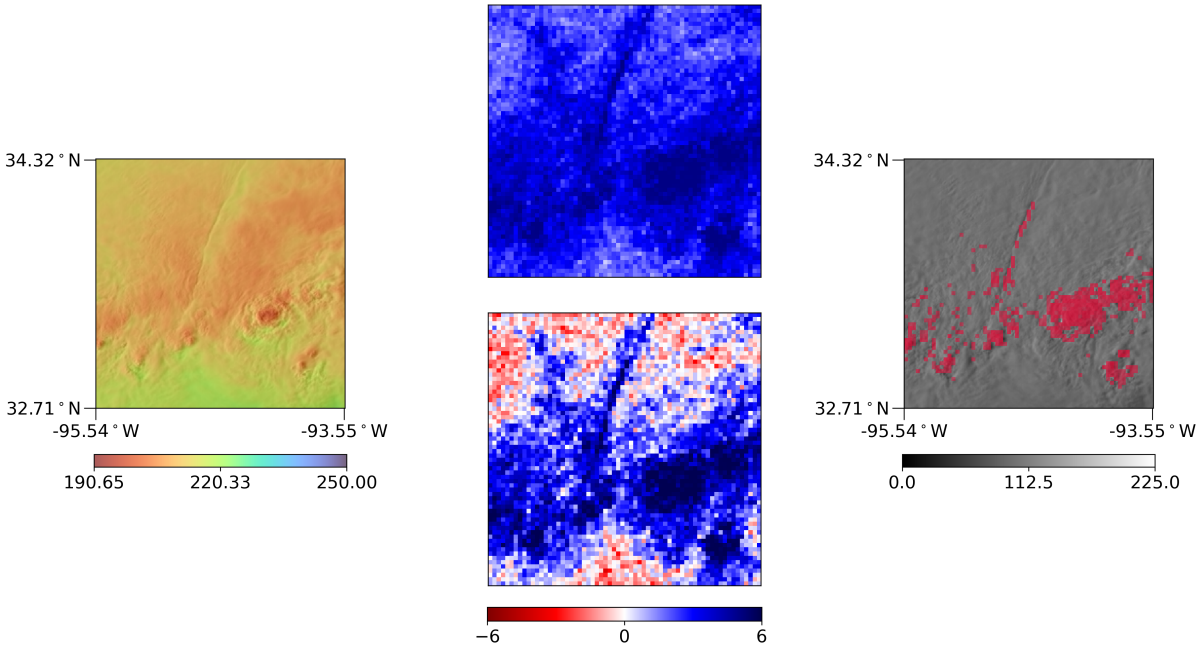


FIG. 19. Imagery taken 21 May 2024 including (a) VIS/IR sandwich product, (b) feature importance corresponding to the cool contrast tiles feature before (first row) and after (second row) the model was altered, and (c) predictions made by the edited model.

contrast tiles feature before (first row) and after (second row) the model was altered, and (c) predictions made by the edited model.

As seen in Case IV, the model struggles to make concise predictions. Once again, this issue is caused by extremely cold temperatures and highly textured regions that do not belong to OTs. Altering the cool contrast tiles feature functions helped to isolate the relevant texture, but the cold temperatures work to overpower the effect, leading to many false positives.

This scene also features a "straight line" of predictions corresponding to a "crease" seen in the middle of the scene. This represents another common failure mode of the model. In this case, regions of high contrast caused by the crease lead to large texture values despite the texture not being "bubbly." This is due to how the texture values are calculated, meaning our current methodology does not offer a way to correct for this (see future work suggested below).

In most other scenes, cloud boundaries also lead to high texture values. The 250 K threshold, however, typically takes care of this issue. In other scenes where there are regions of high texture on the cloud, the temperature is typically not cool enough for false predictions to be made.

5. Conclusions and Future Work

In this work, we make the case for the use of EBMs in combination with physics-informed feature engineering to yield interpretable ML algorithms for certain meteorological applications. This approach has several advantages, including, but not limited to (1) the ability to fully understand the strategies used by the ML algorithm when making predictions, exposing potential failure modes; (2) the opportunity to adjust its strategies to more closely match the strategies expected based on domain knowledge; and (3) the ability to develop a generalizable model from just a few data samples, or, if data for a similar task is available, utilizing those instead in a way analogous to transfer learning.

We have illustrated how these advantageous aspects of the EBM framework can aid in the approach of detecting OT locations from satellite imagery. We emphasize, however, that this application of EBMs was only possible due to feature engineering that first simplified the task at hand. Nevertheless, we believe that this method has the potential to be used in a wide variety of meteorological applications.

At first sight, the identification and tuning of EBM model strategies, which is the part of the EBM development process illustrated in Section 3, may appear to be a lot of extra work, especially when compared to the hands-off training procedure of a comparable neural network model. One should keep in mind, however, that for a neural network model, the identification of strategies should come as a separate step *after* its training is completed, e.g., using XAI methods, but that step is often neglected, since it is nearly impossible to detect most of its strategies anyway. EBMs should thus not be dismissed for enabling and, in fact, requiring this important step during their development process. In other words, this step is simply the price to pay to obtain an interpretable model.

For the application of identifying OTs, the next step in this research should be the creation of a large hand-labeled data set that identifies OTs in GOES visible imagery. Creating such a labeled data set is a larger effort, but it is needed to fully evaluate how well the EBM model matches human labeling. More generally, much work remains to further explore the use of EBMs in the field of meteorology in terms of both identifying the most suitable applications and developing a larger range of engineered features—endeavors we hope will serve to further improve the performance of these models.

Acknowledgments. This material is based upon work supported by the National Science Foundation under AI institute Grant No. 2019758 and CAIG grant No. 2425923; and by the Machine Learning Strategic Initiative at the Cooperative Institute for Research in the Atmosphere.

Data availability statement. The dataset and python code used to train, validate, and test the EBM model will be made publicly available before publication.

References

Adler, R. F., M. J. Markus, and D. D. Fenn, 1985: Detection of severe midwest thunderstorms using geosynchronous satellite data. *Monthly Weather Review*, **113 (5)**, 769 – 781, https://doi.org/10.1175/1520-0493(1985)113⟨0769:DOSMTU⟩2.0.CO;2.

Ai, Y., J. Li, W. Shi, T. J. Schmit, C. Cao, and W. Li, 2017: Deep convective cloud characterizations from both broadband imager and hyperspectral infrared sounder measurements. *Journal of Geophysical Research: Atmospheres*, **122 (3)**, 1700–1712, https://doi.org/https://doi.org/10.1002/2016JD025408, https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2016JD025408.

Arslan, A. K., F. H. Yagin, A. Algarni, F. Al-Hashem, and L. P. Ardigò, 2024: Combining the strengths of the explainable boosting machine and metabolomics approaches for biomarker discovery in acute myocardial infarction. *Diagnostics*, **14 (13)**, 1353.

Bedka, K., J. Brunner, R. Dworak, W. Feltz, J. Otkin, and T. Greenwald, 2010: Objective satellite-based detection of overshooting tops using infrared window channel brightness temperature gradients. *Journal of Applied Meteorology and Climatology*, **49 (2)**, 181 – 202, https://doi.org/10.1175/2009JAMC2286.1.

Bedka, K. M., and K. Khlopenkov, 2016: A probabilistic multispectral pattern recognition method for detection of overshooting cloud tops using passive satellite imager observations. *Journal of Applied Meteorology and Climatology*, **55 (9)**, 1983 – 2005, https://doi.org/10.1175/JAMC-D-15-0249.1.

Bommer, P. L., M. Kretschmer, A. Hedström, D. Bareeva, and M. M.-C. Höhne, 2024: Finding the right xai method—a guide for the evaluation and ranking of explainable ai methods in climate science. *Artificial Intelligence for the Earth Systems*, **3 (3)**, e230 074.

Brunner, J. C., S. A. Ackerman, A. S. Bachmeier, and R. M. Rabin, 2007: A quantitative analysis of the enhanced-v feature in relation to severe weather. *Weather and Forecasting*, **22 (4)**, 853 – 872, https://doi.org/10.1175/WAF1022.1.

Caleca, F., P. Confuorto, F. Raspini, S. Segoni, V. Tofani, N. Casagli, and S. Moretti, 2024: Shifting from traditional landslide occurrence modeling to scenario estimation with a "glass-box" machine learning. *Science of the total environment*, **950**, 175 277.

Caruana, R., 2023: Friends don't let friends deploy black-box models: The importance of intelligibility in machine learning. AI2ES Side-wide meeting, Recording available at https://drive.google.com/file/d/1L1RGG_R0v-NGjGakfqLKhIt4qC63eUKy/view.

Caruana, R., Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, 2015: Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, NY, USA, 1721–1730, KDD '15, https://doi.org/10.1145/2783258.2788613, URL https://doi.org/10.1145/2783258.2788613.

Celik, M. F., M. S. Isik, E. Erten, and G. Taskin, 2023: Informative earth observation variables for cotton yield prediction using explainable boosting machine. *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*, IEEE, 3542–3545.

Cooney, J. W., K. M. Bedka, C. A. Liles, and C. R. Homeyer, 2025: Automated detection of overshooting tops and above-anvil cirrus plumes within geostationary imagery using deep learning. *Artificial Intelligence for the Earth Systems*, **4 (2)**, e240 037, https://doi.org/10.1175/AIES-D-24-0037.1.

Dworak, R., K. Bedka, J. Brunner, and W. Feltz, 2012: Comparison between goes-12 overshooting-top detections, wsr-88d radar reflectivity, and severe storm reports. *Weather and Forecasting*, **27 (3)**, 684 – 699, https://doi.org/10.1175/WAF-D-11-00070.1.

Fan, D., S. J. Greybush, E. E. Clothiaux, and D. J. Gagne, 2024: Physically explainable deep learning for convective initiation nowcasting using goes-16 satellite observations. *Artificial Intelligence for the Earth Systems*, **3 (3)**, e230 098.

Flora, M., C. Potvin, A. Mcgovern, and S. Handler, 2023: A machine learning explainability tutorial for atmospheric sciences. *Artificial Intelligence for the Earth Systems*, **3**, https://doi.org/10.1175/AIES-D-23-0018.1.

Gao, C., H. Wang, Q. Ge, and J. Dai, 2024: Interpreting the influences of multiple factors on forcing requirements of leaf unfolding date by explainable machine learning algorithms. *Ecological Indicators*, **166**, 112 402.

Haralick, R., K. Shanmugan, and I. Dinstein, 1973: Texture features for image classification. *IEEE Trans. Systems Man Cybernet.*, **SMC-3 (6)**, 610–621.

Hastie, T., and R. Tibshirani, 1987: Generalized additive models: Some applications. *Journal of the American Statistical Association*, **82 (398)**, 371–386.

Hegselmann, S., C. Ertmer, T. Volkert, A. Gottschalk, M. Dugas, and J. Varghese, 2022: Development and validation of an interpretable 3 day intensive care unit readmission prediction model using explainable boosting machines. *Frontiers in Medicine*, **9**, 960 296.

Heymsfield, G. M., R. Fulton, and J. D. Spinhirne, 1991: Aircraft overflight measurements of midwest severe storms: Implications an geosynchronous satellite interpretations. *Monthly Weather Review*, **119 (2)**, 436 – 456, https://doi.org/10.1175/1520-0493(1991)119⟨0436:AOMOMS⟩2.0.CO;2.

Hilburn, K. A., 2023: Understanding spatial context in convolutional neural networks using explainable methods: Application to interpretable gremlin. *Artificial Intelligence for the Earth Systems*, **2 (3)**, 220 093, https://doi.org/10.1175/AIES-D-22-0093.1.

Ibtehaz, N., and M. S. Rahman, 2020: Multiresunet : Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural Networks*, **121**, 74–87, https://doi.org/https://doi.org/10.1016/j.neunet.2019.08.025.

Khattak, A., P.-w. Chan, F. Chen, and H. Peng, 2023a: Assessing wind field characteristics along the airport runway glide slope: An explainable boosting machine-assisted wind tunnel study. *Scientific Reports*, **13 (1)**, 10 939.

Khattak, A., J. Zhang, P.-W. Chan, F. Chen, and H. Almujibah, 2023b: Explainable boosting machine: A contemporary glass-box strategy for the assessment of wind shear severity in the runway vicinity based on the doppler light detection and ranging data. *Atmosphere*, **15 (1)**, 20.

Khlopenkov, K. V., K. M. Bedka, J. W. Cooney, and K. Itterly, 2021: Recent advances in detection of overshooting cloud tops from longwave infrared satellite imagery. *Journal of Geophysical Research: Atmospheres*, **126 (14)**, e2020JD034 359, https://doi.org/https://doi.org/10.1029/2020JD034359, https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2020JD034359.

Kim, M., J. Im, H. Park, S. Park, M.-I. Lee, and M.-H. Ahn, 2017: Detection of tropical overshooting cloud tops using himawari-8 imagery. *Remote Sensing*, **9 (7)**, https://doi.org/10.3390/rs9070685.

Kim, M., J. Lee, and J. I. and, 2018: Deep learning-based monitoring of overshooting cloud tops from geostationary satellite data. *GIScience & Remote Sensing*, **55 (5)**, 763–792, https://doi.org/10.1080/15481603.2018.1457201, https://doi.org/10.1080/15481603.2018.1457201.

Körner, A., B. Sailer, S. Sari-Yavuz, H. A. Haeberle, V. Mirakaj, A. Bernard, P. Rosenberger, and M. Koeppen, 2024: Explainable boosting machine approach identifies risk factors for acute renal failure. *Intensive Care Medicine Experimental*, **12 (1)**, 55.

Krell, E., P. Tissot, A. Mamalakis, W. Collins, I. Ebert-Uphoff, and S. King, 2024: Using grouped features to improve explainable ai results for atmospheric ai models that use gridded spatial data and complex machine learning technique. *Authorea Preprints*.

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)**, 1096.

Lee, Y., C. D. Kummerow, and I. Ebert-Uphoff, 2021a: Applying machine learning methods to detect convection using geostationary operational environmental satellite-16 (goes-16) advanced baseline imager (abi) data. *Atmospheric Measurement Techniques*, **14 (4)**, 2699–2716, https://doi.org/10.5194/amt-14-2699-2021.

Lee, Y., C. D. Kummerow, and M. Zupanski, 2021b: A simplified method for the detection of convection using high-resolution imagery from goes-16. *Atmospheric Measurement Techniques*, **14 (5)**, 3755–3771, https://doi.org/10.5194/amt-14-3755-2021.

Lou, Y., R. Caruana, and J. Gehrke, 2012: Intelligible models for classification and regression. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, NY, USA, 150–158, KDD '12, https://doi.org/10.1145/2339530.2339556, URL https://doi.org/10.1145/2339530.2339556.

Lou, Y., R. Caruana, J. Gehrke, and G. Hooker, 2013: Accurate intelligible models with pairwise interactions. *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, NY, USA, 623–631, KDD '13, https://doi.org/10.1145/2487575.2487579, URL https://doi.org/10.1145/2487575.2487579.

McGovern, A., A. Bostrom, M. McGraw, R. J. Chase, D. J. Gagne, I. Ebert-Uphoff, K. D. Musgrave, and A. Schumacher, 2024: Identifying and categorizing bias in ai/ml for earth sciences. *Bulletin of the American Meteorological Society*, **105 (3)**, E567–E583.

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.

McGovern, A., R. Lagerquist, D. John Gagne, 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. *Bulletin of the American Meteorological Society*, **100 (11)**, 2175–2199.

Mikuš, P., and N. S. Mahović, 2013: Satellite-based overshooting top detection methods and an analysis of correlated weather conditions. *Atmospheric Research*, **123**, 268–280, https://doi.org/https://doi.org/10.1016/j.atmosres.2012.09.001.

Moen, K., 2024: Gray level co-occurrence matrix and its application to weather satellite imagery, URL https://www.math.colostate.edu/~king/Moen_MS_Project.pdf, Master's project, Department of Mathematics, Colorado State University.

Morgan, H., K. Wang, M. Dohopolski, X. Liang, M. R. Folkert, D. Sher, and J. Wang, 2021: Explainable boosting machine model with a parallel ensemble design predicts local failure for head and neck cancer with clinical, ct, and delta cbct radiomic features. *International journal of radiation oncology, biology, physics*, **111 (3)**, e115–e116.

Nanushi, O., V. Sitokonstantinou, I. Tsoumas, and C. Kontoes, 2022: Pest presence prediction using interpretable machine learning. *2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, IEEE, 1–5.

Negri, A. J., and R. F. Adler, 1981: Relation of satellite-based thunderstorm intensity to radar-estimated rainfall. *Journal of Applied Meteorology and Climatology*, **20 (3)**, 288 – 300, https://doi.org/10.1175/1520-0450(1981)020⟨0288:ROSBTI⟩2.0.CO;2.

NOAA National Centers for Environmental Information (NCEI), 2024: U.s. billion-dollar weather and climate disasters. "NOAA National Centers for Environmental Information (NCEI)", URL https://www.ncei.noaa.gov/access/billions/, https://doi.org/10.25921/stkw-7w73.

Noiret, S., J. Lumetzberger, and M. Kampel, 2021: Bias and fairness in computer vision applications of the criminal justice system. *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–8, https://doi.org/10.1109/SSCI50451.2021.9660177.

Nori, H., S. Jenkins, P. Koch, and R. Caruana, 2019: Interpretml: A unified framework for machine learning interpretability. *ArXiv*, **abs/1909.09223**.

Oktay, O., and Coauthors, 2018: Attention u-net: Learning where to look for the pancreas. https://doi.org/10.48550/arXiv.1804.03999.

Pant, H., G. Joshi, B. Rawat, H. R. Goyal, Y. Joshi, and C. S. Bohra, 2025: Comparative study of crop yield prediction using explainable ai and interpretable machine learning techniques. *2025 Fifth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, IEEE, 1–7.

Patel, S. S., 2023: Explainable machine learning models to analyse maternal health. *Data & Knowledge Engineering*, **146**, 102 198.

Ronneberger, O., P. Fischer, and T. Brox, 2015: U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI*

2015, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds., Springer International Publishing, Cham, 234–241.

Rudin, C., 2019: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, **1 (5)**, 206–215.

Sarica, A., A. Quattrone, and A. Quattrone, 2021: Explainable boosting machine for predicting alzheimer's disease from mri hippocampal subfields. *International Conference on Brain Informatics*, Springer, 341–350.

Schmit, T. J., P. Griffith, M. M. Gunshor, J. M. Daniels, S. J. Goodman, and W. J. Lebair, 2017: A closer look at the abi on the goes-r series. *Bulletin of the American Meteorological Society*, **98 (4)**, 681 – 698, https://doi.org/10.1175/BAMS-D-15-00230.1.

Smith, A. B., and R. W. Katz, 2013: Us billion-dollar weather and climate disasters: data sources, trends, accuracy and biases. *Nat Hazards*, **67**, 387–410, https://doi.org/10.1007/s11069-013-0566-5.

Toms, B. A., E. A. Barnes, and I. Ebert-Uphoff, 2020: Physically interpretable neural networks for the geosciences: Applications to earth system variability. *Journal of Advances in Modeling Earth Systems*, **12 (9)**, e2019MS002 002.

Wang, B., and Coauthors, 2024: Interpretable predictive value of including hdl-2b and hdl-3 in an explainable boosting machine model for multiclass classification of coronary artery stenosis severity in acute myocardial infarction patients. *European Heart Journal-Digital Health*, ztae100.

Wang, D., S. H. Lee, H. Geng, H. Zhong, J. Plastaras, A. Wojcieszynski, R. Caruana, and Y. Xiao, 2022: Interpretable machine learning for predicting pathologic complete response in patients treated with chemoradiation therapy for rectal adenocarcinoma. *Frontiers in Artificial Intelligence*, **5**, 1059 033.

Yagin, F. H., C. Colak, A. Algarni, A. Algarni, F. Al-Hashem, and L. P. Ardigò, 2025: Explainable boosting machines identify key metabolomic biomarkers in rheumatoid arthritis. *Medicina*, **61 (5)**, 833.