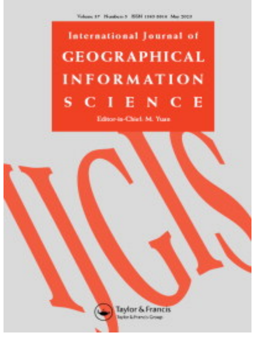

Taylor & Francis
Taylor & Francis Group

International Journal of Geographical Information Science

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tgis20

Explainable GeoAI: can saliency maps help interpret artificial intelligence's learning process? An empirical study on natural feature detection

Chia-Yu Hsu & Wenwen Li

To cite this article: Chia-Yu Hsu & Wenwen Li (2023) Explainable GeoAI: can saliency maps help interpret artificial intelligence's learning process? An empirical study on natural feature detection, International Journal of Geographical Information Science, 37:5, 963-987, DOI: 10.1080/13658816.2023.2191256

To link to this article: https://doi.org/10.1080/13658816.2023.2191256

Published online: 24 Mar 2023.

INTERNATIONAL JOURNAL OF GEOGRAPHICAL INFORMATION SCIENCE
2023, VOL. 37, NO. 5, 963–987
https://doi.org/10.1080/13658816.2023.2191256

RESEARCH ARTICLE

Explainable GeoAI: can saliency maps help interpret artificial intelligence's learning process? An empirical study on natural feature detection

Chia-Yu Hsu and Wenwen Li

School of Geographical Sciences and Urban Planning, Arizona State University, Tempe, AZ, USA

ABSTRACT

Improving the interpretability of geospatial artificial intelligence (GeoAI) models has become critically important to open the 'black box' of complex AI models, such as deep learning. This paper compares popular saliency map generation techniques and their strengths and weaknesses in interpreting GeoAI and deep learning models' reasoning behaviors, particularly when applied to geospatial analysis and image processing tasks. We surveyed two broad classes of model explanation methods: perturbation-based and gradient-based methods. The former identifies important image areas, which help machines make predictions by modifying a localized area of the input image. The latter evaluates the contribution of every single pixel of the input image to the model's prediction results through gradient backpropagation. In this study, three algorithms—the occlusion method, the integrated gradients method, and the class activation map method—are examined for a natural feature detection task using deep learning. The algorithms' strengths and weaknesses are discussed, and the consistency between model-learned and human-understandable concepts for object recognition is also compared. The experiments used two GeoAI-ready datasets to demonstrate the generalizability of the research findings.

ARTICLE HISTORY
Received 8 September 2022
Accepted 12 March 2023

KEYWORDS
XAI; artificial intelligence; deep learning; visualization; GeoAI

1. Introduction

Geospatial artificial intelligence (GeoAI) is an exciting and rapidly growing transdisciplinary research area that fuses AI with geographical laws and principles for solving geospatial problems in a data-driven manner (Li 2020). Recent breakthroughs in deep learning technologies (e.g. convolutional neural networks [CNNs], recurrent neural networks, transformers, and deep reinforcement learning) have led to a flourishing of their applications in geographic information science (GIScience), such as terrain feature detection (Buscombe and Ritchie 2018, Helber *et al.* 2019, Li and Hsu 2020, Wang and Li 2021, Hsu *et al.* 2021), weather forecasting and nowcasting (Zhang *et al.* 2019), extreme climate event detection (Kurth *et al.* 2018), and neighborhood property

CONTACT Wenwen Li ✉ wenwen@asu.edu

quantification and change monitoring (Gebru *et al*. 2017, Koo *et al*. 2022, Li 2022a). Although a wide variety of methods have been developed to support image analysis and machine vision, CNNs remain the most widely used type of deep learning model, achieving state-of-the-art performance (Li and Hsu 2022, Li *et al*. 2022a). By stacking multiple convolutional layers together, a CNN has the ability to hierarchically extract the prominent features of the target objects within an image scene for classification, detection, or segmentation purposes.

Although CNNs have a well-defined model structure and reproducible parameters, their reasoning process remains a black box and is difficult to interpret due to the complex, non-linear nature of the model. The opaque decision-making process raises concerns about the scientific trustworthiness of the model's prediction results (Li and Arundel 2022, Kedron *et al*. 2021). The lack of model explainability may further hinder the replicability of research using GeoAI (Goodchild and Li 2021). Hence, besides aiming to achieve outstanding performance in big data analytics in relation to aspects such as detection and prediction, it is also important to develop algorithms that increase our understanding of the knowledge-derivation process of an AI machine. To address this issue, the AI and GeoAI communities have been developing model explanation methods and tools to support the visual examination of model behavior and to connect this to the cognitive concepts used by humans when making decisions. For instance, feature attribution is a commonly used approach to explain a model's predictions by attributing a decision to the input data. Based on different implementation mechanisms, the attribution can be identified at the pixel level, feature level, instance level, and concept level (Lundberg and Lee 2017). As the deep CNN models normally employ more complicated interaction strategies among the neural network layers, the most recent feature attribution methods have focused on identifying attribution at the first three levels (Simonyan *et al*. 2014, Zeiler and Fergus 2014, Fong *et al*. 2019, Selvaraju *et al*. 2020).

In the field of geography, efforts have also been made to enhance the interpretability and explainability of GeoAI. For instance, Li *et al.* (2022b) employs an explainable AI (XAI) package, called SHapley Additive exPlanations (SHAP) to compare the spatial effects extracted by a machine learning-based regression model (i.e. XGBoost) with those of more traditional statistical approaches, such as the spatial lag model and multi-scale geographically weighted regression (MGWR) model. The research found that XGBoost has the ability to excerpt local spatial effects similar to those of classic models. This research shed new lights on using machine learning techniques for modelling spatial processes. Xing and Sieber (2021) also used SHAP, within a land-use classification case, to understand which areas in the input data positively or negatively influenced a CNN model's prediction result. Two example images were provided and the importance of feature maps at different stages of the convolution process were visualized. The authors argued that although the XAI tool can provide some level of explanation, it is difficult to link this explanation to a semantic or geographic concept. They therefore called for a deeper integration of XAI and GeoAI to understand how location and geographic attributes (e.g. those adhering to Tobler's First Law of Geography) play a role in GeoAI modelling and decision making (Li *et al*. 2021). Another interesting work was published by Duckham *et al*. (2022), who described an explainable spatiotemporal reasoning software framework for GeoAI applications. Different from the above works, this research adopted a top-down approach, using ontology and rule-based reasoning to derive new information.

To further advance the field of explainable geospatial artificial intelligence (GeoAI), particularly as it relates to making deep learning models (e.g. CNNs) more explainable, this paper surveys and compares popular model explanation approaches from computer vision. It uses a GeoAI-ready natural feature dataset used in Li and Hsu (2020) as an example to discuss the strengths and weaknesses of each method, catalog the feature types each method finds, and examine whether they agree with human-understandable concepts for characterizing a natural feature. To demonstrate our conclusions' generalizability, the experiments were also conducted on GeoNat v1.0—another natural feature training dataset (Arundel *et al*. 2020). The remainder of this article is organized as follows: Section 2 presents a review of the literature. Section 3 introduces the feature attribution methods for XAI. Section 4 presents a series of experiments that identify the characteristics of each method and uses them to compare the model-learned features and human-understandable concepts for the image classification of natural features. Section 5 summarizes the findings and discusses the generalizability of the research findings. Section 6 concludes the paper and identifies future research directions.

2. Literature review

Recently, deep learning techniques have shown outstanding predictive performance in image analysis and computer vision. However, they yield a lower level of explainability than other AI methods due to model complexity (Gunning and Aha 2019, Li and Arundel 2022). An initial attempt to observe and explain a deep learning model's decision-making process was through visualizing various model components, such as filters and feature maps, but the information gained from this exercise can hardly be mapped to meaningful concepts (Li *et al*. 2017). Recent XAI developments for deep learning models have shown two trends: developing global and per-decision explainable AI algorithms (Phillips *et al*. 2020). A global, explainable AI algorithm treats a deep learning model as a black box that can be queried and develops an algorithm approximation to explain the model. Testing with concept activation vectors (TCAV; Kim *et al*. 2018) is one such global algorithm. It tests a model's decision sensitivity to various pre-defined concepts, such as color, texture, and some specific patterns, from which it draws a conclusion about important decision factors for the model. Conversely, a per-decision explainable AI algorithm aims to determine why the model made a particular decision. Unlike the global algorithm, a per-decision algorithm does not require predefining or generating hypotheses about which concepts are important to the model. Instead, it distills evidence from each individual decision, allowing for the discovery of potentially unknown patterns and concepts to improve the understanding of a model's learning process. We will mainly examine algorithms belonging to the second category in our research.

A very popular technique for deriving per-decision explanation is through the generation of a saliency map that visualizes the importance of different regions in the input image that factored into the final decision (i.e. classification or object detection). As known, deep CNN models stack multiple convolutional layers together to perform feature extraction from the original input image. During this process, downsampling is often applied to the input image and the feature maps generated from each convolutional layer. After the prominent image features are extracted, the feature maps are sent to fully connected layers for image classification. The model training process involves learning a set of model parameters, also called weights, to make a prediction as close to the ground truth as possible. The learning process is iterative by nature and contains forward propagation (i.e. forward pass) and backward propagation (i.e. backward pass). During the forward pass, the classification result is generated and compared with the ground-truth to calculate the loss. The loss is propagated back to each weight through backpropagation, which is a way to compute the partial derivative of the loss with respect to the weights such that a model learns how to adjust each weight and improve its predictability. The set of values capturing partial derivatives is called the *gradient* and the weight adjustment process is called the *gradient descent* because the weights are adjusted to *minimize* the loss.

The saliency map generation techniques leveraged to explain CNN model behaviors visually and quantitatively can be categorized into two broad classes: perturbation-based methods (Zeiler and Fergus 2014, Ribeiro *et al*. 2016, Fong and Vedaldi 2017, Sundararajan *et al*. 2017, Petsiuk *et al*. 2018, Hesse *et al*. 2021) and gradient-based methods (Simonyan *et al*. 2014, Zeiler and Fergus 2014, Springenberg *et al*. 2015, Zhou *et al*. 2016, Selvaraju *et al*. 2020).

Perturbation-based methods systematically modify different portions of the input image and analyze the output sensitivity. They reveal which image regions contribute more and are likely more important to the model's prediction results. Zeiler and Fergus (2014) developed an approach to perturb images with a gray patch sliding across the images and monitor result changes in both the classifier's output and the feature maps. We call this an *occlusion approach*. Perturbation-based methods that involve strategies other than occluding images by raster scanning have also been introduced, such as one involving random masking using Monte Carlo sampling (Petsiuk *et al*. 2018) and another using superpixel masking (Ribeiro *et al*. 2016). Here superpixels represent a continuous patch of pixels with similar intensities. Fong and Vedaldi (2017) further proposed an optimization approach to identify a mask that minimizes the prediction score of a certain class; this mask would contain much of the information responsible for a classifier's decision about an object class. In these perturbation-based techniques, the masks may consist of constant values, noise, or a blurring effect. Although perturbation-based methods have been proven to be effective in removing key information from the original image (Fong and Vedaldi 2017), there is concern about whether these methods would also introduce spurious structures in the input image and therefore produce unexpected outputs (Nguyen *et al*. 2015, Kurth *et al*. 2018).

Another family of methods for generating saliency maps is gradient-based methods. Unlike perturbation-based methods, which measure the influence of local areas on the prediction results, gradient-based methods measure the contribution of each individual pixel. These methods also answer the question of how much the prediction results would change if variation were introduced in a single given pixel. If pixels are viewed as variables and the output as a function of them, these methods are essentially computing the partial derivative of the output with respect to a given pixel. Backpropagation facilitates the calculation of partial derivatives, and the results can also be called the gradient, which can then be visualized as a *saliency map*. It is for this reason that these types of CNN visualizations are called *gradient-based methods*. Some recent methods calculate gradients at the input space (Simonyan *et al*. 2014, Zeiler and Fergus 2014, Springenberg *et al*. 2015), while others need only gradients at the middle layers (Zhou *et al*. 2016, Selvaraju *et al*. 2020). For methods calculating gradients at the input space, the major difference is the way they handle backpropagation through the non-linear layers (e.g. Rectified Linear Unit [ReLU]). Different treatments will result in different saliency maps. Simonyan *et al*. (2014) adopted vanilla backpropagation by which, at the non-linear layers, locations with negative values during the forward pass are recorded while gradients at those locations during the backward pass are suppressed. The deconvolution network (DeconvNet; Zeiler and Fergus 2014) only propagates positive gradients and suppresses negative gradients. Finally, guided backpropagation (Springenberg *et al*. 2015) combines both the DeconvNet and the vanilla backpropagation, only propagating positive gradients at the locations with positive values during the forward pass.

For the methods that need only gradients at a given middle convolutional layer, the gradients can be used as weights to combine feature maps at different channels and generate a saliency map. Because the feature maps are downsampled from the input image, this saliency map is considered as a downsampled version of the saliency map at the input space. To improve the visual effect, the saliency map is then upsampled to the same dimension as the input space. Zhou *et al*. (2016) developed a Class Activation Map (CAM) by applying global average pooling (GAP; Lin *et al*. 2014) on the last convolutional layer and identifying the weights of feature maps from the following fully connected layer. Selvaraju *et al*. (2020) proposed a generalization of a CAM called gradient-weighted CAM (Grad-CAM), which is applicable to different CNN models without using a GAP layer. The idea is to use gradient signals at the target layer as weights and prove them to be mathematically equal to the weights in CAM. In this way, Grad-CAM can derive the saliency map without needing to modify and retrain the CNN models. Methods that have followed up Grad-CAM include Grad-CAM++ (Chattopadhay *et al*. 2018) for the localization of multi-objects belonging to the same class and Score-CAM (Wang *et al*. 2020), which obtains weights for corresponding feature maps without gradient calculation.

These gradient-based methods require only one forward-backward pass and are computationally efficient compared to the perturbation-based methods. However, the nonlinearity of the fully connected layers may cause gradient saturation or undesirable artifacts during backpropagation (Shrikumar *et al*. 2017). Recent works have developed new strategies to address this issue. For example, Sundararajan *et al*. (2017) developed an integrated gradient method to aggregate the gradients over an entire image as it goes through continuous modifications, to avoid gradient saturation and unexpected artifacts. The entire image is gradually altered from the original image to an all-black image (i.e. baseline). The result is an aggregation of all the intermediate results after each alternation. This method can be considered as a combination of perturbation-based and gradient-based methods and has shown promising results in many tasks (Sundararajan *et al.* 2017). However, as inferred from the algorithm strategy, the computation of integrated gradients requires repeated calculation over multiple iterations. To reduce computational cost, Hesse *et al.* (2021) proposed a special class of CNNs to compute the result of integrated gradients with only one forward-backward pass. As a result, the saliency map itself can be used as an effective tool; for example, it can be integrated into regular training as priors (Erion *et al.* 2021, Rieger *et al.* 2020).

Each of the discussed methods has its own strengths and weaknesses. For example, the perturbation-based method may underestimate the importance of features because other features already saturate the output (Shrikumar *et al.* 2017). Gradient-based methods may generate the same results using an untrained model as those from a trained model, indicating that the method demonstrates only general model characteristics (Adebayo *et al.* 2018). Although CAM methods can generate a saliency map by passing through fewer non-linear layers and potentially suffer from fewer issues, they may predict inaccurate object locations due to the coarse resolution of the saliency map. In this paper, we explore the characteristics of different 'feature attribution-based' model explanation approaches and use two GeoAI-ready natural feature datasets in an image classification task to compare model-learned features with human-understandable features. In addition to interpreting a model's reasoning process, we compare the strengths and weaknesses of these popular approaches and discuss ways to improve them within the growing field of explainable GeoAI.

3. Methods

Suppose we have a function $F : R^{w\times h} \rightarrow R^{m}$, which represents a CNN network for image classification. The input is an image $x \in R^{w\times h}$ with width w and height h. The output is a vector of length m, representing the probability distribution P over m classes. P^{c} stands for the classification probability of the class c to the image x. For any given class c, we could generate an image y where $y_{i,j}$ (i.e. pixel *[i, j]* in image y) is the contribution of $x_{i,j}$ (i.e. pixel *[i, j]* in image x) to the prediction of the image class c. We call this image y a saliency map. In this section, we describe in greater detail the three different methodologies adopted to generate a saliency map.

3.1. Perturbation-based saliency map generation (i.e. occlusion method)

To generate a perturbation-based saliency map, we adopt the occlusion method from the work by Zeiler and Fergus (2014). Given the function F and an input image x, we calculate the output probability distribution P. P^{c} represents the output probability for the class c. Next, we create a black patch at a given size. The patch slides through the image x in a row-prime scan order with a stride step of 1. At each step, a new image $x'(i,j)$ is generated by replacing a portion of the image x with the black patch. The center of the patch is located at the pixel (i,j). The new image $x'(i,j)$ is then fed into the model F, and the new probability distribution $P'(i,j)$ is calculated. $P'c(i,j)$

represents the new probability of the class c. The difference between the two probabilities, P^c and $P'c(i,j)$, shows the importance of the information in the black patch to the model's prediction of the class c:

$$\text{Occlusion}^c_{i,j} = P^c - P'c(i,j) \tag{1}$$

Finally, by calculating $\text{Occlusion}^c_{i,j}$ over all pixels, we can generate the saliency map indicating which pixels or pixel regions are more or less important for predicting the existence of the class c in image x.

3.2. CAM-series saliency map generation with Grad-CAM

The idea of Grad-CAM (Selvaraju *et al.* 2020) is to calculate the gradient at the last convolutional layer in the network from a given class-prediction score, together with the feature maps at this layer, to generate a coarse saliency map (i.e. a CAM) highlighting important areas for predicting the class. Grad-CAM can be directly applied to a trained CNN network to derive a saliency map mathematically. It allows for the generalization of a regular CAM (Zhou *et al.* 2016) without the need to add a GAP (Global Average Pooling) layer to the network and retrain the CNN model. The process of generating a saliency map with a regular CAM is as follows. Given the function F and an input image x, $A^k_{i,j}$ can be used to represent the value at location (i,j) of the kth feature map resulting from the last convolutional layer. Further, g^k is the output of GAP, the function of which can be written as:

$$g^k = \frac{1}{Z}\sum_{i,j} A^k_{i,j} \tag{2}$$

where Z is the total number of pixels in feature map A^k.

Next, let S^c represent the prediction score of the class c. S^c can be derived by calculating the composited value from g^k, as follows:

$$S^c = \sum_k w^{ck} g^k \tag{3}$$

where w^{ck} is the relative weight of g^k to S^c. Accordingly, w^{ck} can indicate the importance of g^k to the prediction of the class c because the bigger w^{ck} is, the more weight that g^k requires to derive the class c. The probability of an image being the class c, P^c, is derived by applying softmax to score S over all classes. Softmax is a normalization function to transfer the prediction scores to a probability distribution over the predicted classes. The higher S^c (i.e. class prediction score) is, the higher P^c will be. Next, by substituting g^k in Equation 3 with its definition in Equation 2, we can derive the following:

$$S^c = \sum_k w^{ck} \frac{1}{Z}\sum_{i,j} A^k_{i,j} = \frac{1}{Z}\sum_{i,j}\sum_k w^{ck} A^k_{i,j} \tag{4}$$

Further, by making

$$M^c_{i,j} = \sum_k w^{ck} A^k_{i,j} \tag{5}$$

we have

$$S^c = \frac{1}{Z}\sum_{i,j} M^c_{i,j} \tag{6}$$

Because S^c is the prediction score for the class c, $M^c_{i,j}$ directly indicates the importance of the location (i,j) in the last feature map (i.e. by aggregating information from all channels) to the class c. Therefore, if we calculate $M^c_{i,j}$ over all locations, we can derive a saliency map indicating the important areas for making predictions about the class c.

However, to generate $M^c_{i,j}$ (in Equation (5)) over all locations, we need both $A^k_{i,j}$ and weight w^{ck} at the pixel level. In a regular CAM approach (Zhou *et al*. 2016), w^{ck} is generated through GAP, which requires changes to the CNN model architecture. In Grad-CAM (Selvaraju *et al*. 2020), however, a generalization of the CAM is achieved through the mathematical transformation of the gradient calculations. The gradients are the partial derivatives of the target class score S^c with respect to the feature map k). Mathematically, it can be written as follows. First, by taking partial derivatives of S^c with respect to g^k from Equation (3) and combining information from Equation (2), we can derive

$$\frac{\partial S^c}{\partial g^k} = w^{ck} = \frac{\partial S^c}{\partial A^k_{i,j}} \cdot \frac{\partial A^k_{i,j}}{\partial g^k} = \frac{\partial S^c}{\partial A^k_{i,j}} \cdot Z \tag{7}$$

For the weight w^{ck}, its relation to a given pixel at (i,j) in feature map A^k can be identified using

$$w^{ck} = Z \cdot \frac{\partial S^c}{\partial A^k_{i,j}} \tag{8}$$

If we calculate the summation of w^{ck} overall pixels, it can be expressed as

$$\sum_{i,j} w^{ck} = \sum_{i,j} Z \cdot \frac{\partial S^c}{\partial A^k_{i,j}} = Z \cdot \sum_{i,j} \frac{\partial S^c}{\partial A^k_{i,j}} \tag{9}$$

At the same time, the weight is not a function of the location (i,j), so the summation can also be expressed as

$$\sum_{i,j} w^{ck} = Z \cdot w^{ck} \tag{10}$$

Combining Equations (9) and (10), the weight w^{ck} can be mathematically derived from the partial derivatives of S^c with respect to all pixels at feature map A^k, as follows:

$$w^{ck} = \sum_{i,j} \frac{\partial S^c}{\partial A^k_{i,j}} \tag{11}$$

Equation (11) indicates a way to calculate w^{ck} without using a GAP layer. Therefore, we can replace w^{ck} in Equation (5) with Equation (11), giving

$$\mathrm{GradCAM}^c_{i,j} = M^c_{i,j} = \sum_k \sum_{i,j} \frac{\partial S^c}{\partial A^k_{i,j}} \cdot A^k_{i,j} \tag{12}$$

which provides the importance of a pixel at a given location (i,j) to the prediction of the class c based on the class prediction score S^c and the final feature map A at each channel k. This way, no architecture changes to the model are required. Because the generated saliency map is a coarse map with a resolution smaller than the original input image, it can also be upsampled to obtain a better visual effect.

3.3. Gradient-based saliency map generation with integrated gradients

The motivation of integrated gradients (Sundararajan *et al*. 2017) is to solve the challenge of separating actual errors from misbehavior of the model and errors caused by the model explanation method. The authors identified two axioms in order to evaluate the validity of different attribution methods: sensitivity and implementation invariance. The sensitivity axiom indicates that a pixel should be given a non-zero contribution if two input images have different predictions but only differ in that pixel. The implementation invariance axiom indicates that the saliency maps are always identical for two functionally equivalent models. *Functionally equivalent* models are those yielding the same output given any input, despite their different implementations. By satisfying the two axioms, errors introduced by the saliency map generation method can be disregarded. Using the two axioms, a new method—the integrated gradients method—was developed (Sundararajan *et al*. 2017). One key concept of the integrated gradients method is the use of a *baseline* image. The idea of a *baseline* image is implicitly used in previously derived methods because, when assigning credit to a pixel or a subregion of an image, we actually consider the image without the pixel or the subregion as a baseline image with which to compare the difference in outputs. In the integrated gradients method, however, the baseline image is explicitly used, taking the form of a black image (i.e. all zeros) for a classification task. Given the classification function F, an input image x, and a baseline image x', the integrated gradients method defines the contribution of a pixel $x_{i,j}$ to the prediction of a class c as

$$\text{Integrated Gradients}_{i,j}^{c} = (x_{i,j} - x'_{i,j}) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \cdot (x - x'))^{c}}{\partial x_{i,j}} d\alpha \qquad (13)$$

Equation (13) demonstrates why this method is named after *integrated gradients*, as it computes and integrates gradients of the final output with respect to pixel $x_{i,j}$ from the baseline image x' to the original input image x. In other words, it identifies the ‘path integrals’ of gradients along the linear path from x' to x. Within the fundamental theorem of path integrals, if the function F is differentiable almost everywhere, the summation of Equation (13) over all pixels can be written as

$$\sum_{i,j} \text{Integrated Gradients}_{i,j}^{c} = F(x)^{c} - F(x')^{c} \qquad (14)$$

In our experiment, we assume the black baseline image would make the prediction score near zero—that is, $F(x')^c \approx 0$. Therefore, the prediction of image x being the class c ($F(x)^c$) equals the summation of the contributions from all pixels. This confirms the accuracy of Equation (13) in calculating an individual pixel’s contribution to the final prediction. The integrated gradients method also satisfies the aforementioned axioms. In terms of sensitivity, assuming that only one given pixel is different between the input image x and the baseline x', from Equation (13), we know that all of the integrated gradients would be zero except for those of the given pixel. From Equation (14), the contribution of this given pixel becomes the difference between the outputs of $F(x)^c$ and $F(x')^c$. Therefore, if $F(x)^c$ and $F(x')^c$ are different, the contribution is non-zero. In terms of implementation invariance, because Equation (13) only considers the gradients of the model, it is invariant to different architectures. To compute integrated gradients more efficiently, instead of calculating the integrals from Equation (13), we can also add the gradients at sufficiently small intervals from the baseline image x' to the original image x. An approximation via summation can be achieved as

$$\text{Integrated Gradients}^{c}_{i,j}{}' = (x_{i,j} - x'_{i,j}) \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \cdot (x - x'))^c}{\partial x_{i,j}} \cdot \frac{1}{m} \qquad (15)$$

where m is the number of intervals. The authors (Sundararajan *et al.* 2017) suggested that a number between 20 and 300 steps is sufficient to approximate the integral (at a 95% accuracy).

4. Experimental results

To understand how the CNN models make decisions in image analysis tasks specifically when inspecting the different types of terrain features in an image, we adopted the same dataset used in the work by Li and Hsu (2020). The terrain dataset contained 826 natural features of eight types (i.e. crater, meander, river, hill, lake, volcano, iceberg tongue, and sand dune). Each category had 100 to 108 images. Figure 1 shows examples of training images containing features of each type. Briefly, a crater is ‘a circular-shaped depression on the surface of the land’ (McEwen *et al.* 1983). A river is ‘a natural flowing watercourse, usually freshwater, flowing towards an ocean, sea, or

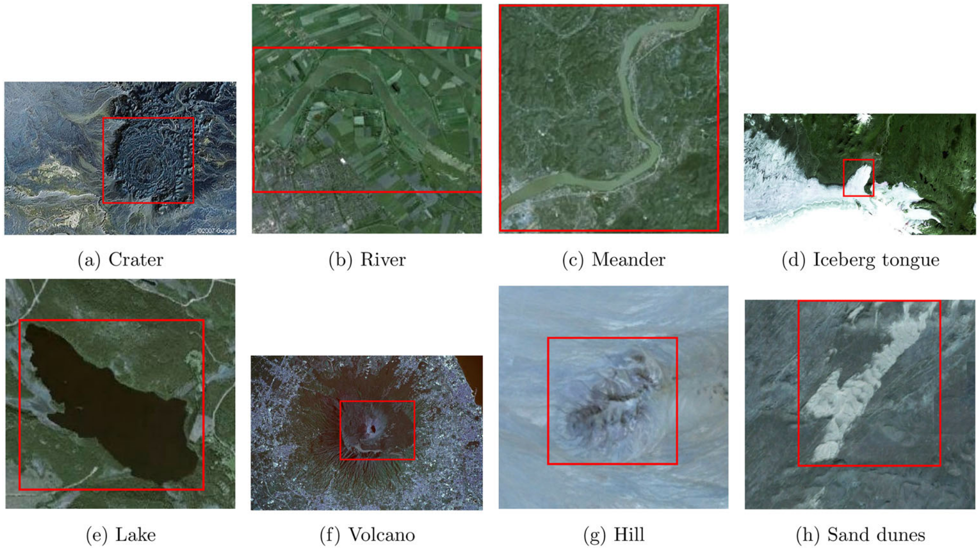
(a) Crater (b) River (c) Meander (d) Iceberg tongue
(e) Lake (f) Volcano (g) Hill (h) Sand dunes

Figure 1. Sample dataset images. The bounding boxes are labels indicating object locations. (a) Crater. (b) River. (c) Meander. (d) Iceberg tongue. (e) Lake. (f) Volcano. (g) Hill. (h) Sand dunes.

another river' (Wikipedia 2022b). A meander is 'a winding curve or bend in a river' (Wikipedia 2022a). An iceberg tongue is 'a seaward extension of a glacier tongue, which consists of floating glacier ice that is still connected [to the glacier] and extends to the sea or ocean from a glacier' (Herzfeld 2004). A lake is 'an area of land that is filled with water' (Purcell 2018). A volcano is 'a rupture in the crust of a planetary-mass object, such as Earth, that allows hot lava, volcanic ash, and gases to escape from a magma chamber below the surface' (Wikipedia 2022c). A hill is 'a naturally raised area of land, not as high or craggy as a mountain' (National Geographic 2022). A sand dune is 'a small ridge of hill of sand found in a desert or on top of a beach' (EarthEclipse 2022).

The dataset was originally used for object detection tasks, and here, we adopted it for image-level classification using only the class labels (i.e. without the bounding-box labels) for model training and prediction. The dataset was randomly separated into 70% for training and 30% for testing. The classification model was trained using a pre-trained VGG16 CNN (Simonyan and Zisserman 2015). The final training accuracy was 99.83%, and the testing accuracy was 88.05%. All experiments were conducted on the Amazon EC2 platform. The g4dn.xlarge instance, with a NVIDIA T4 graphics processing unit (GPU) that had a 16GB memory, was used to run the experiments.

In each experiment, we applied three model explanation techniques—including occlusion sensitivity by Zeiler and Fergus (2014), Grad-CAM by Selvaraju *et al*. (2020), and the integrated gradients method by Sundararajan *et al*. (2017)—to explain the model's decision-making process. The three methods were all applied to a trained model, and the generation of the saliency maps is detailed in Section 3. By comparing the results, we aimed to examine the characteristics of each method, cross-validate their findings, and, most importantly, use them jointly to explain the model's classification results in image analysis and natural feature recognition. Figures 2–7, below, provide the results. In these figures, the first column displays the original image, with the targeted feature(s) highlighted in a rectangle. The ground-truth label of the object class and the model's prediction results, with a confidence score, are listed below the image. The second column lists the saliency maps generated from the three model explanation methods. The last column shows the blending result of the saliency map on top of the original image to make observing highlighted regions easier. For each saliency map, higher values (indicated in red) represent the pixels or image areas that are more important to the prediction results. After cross-validating the saliency maps, we found that the findings of each method showed distinct characteristics due to the methods' unique model interpretation strategies.

4.1. Ability to identify multiple objects of the same type

Figure 2 presents an image with multiple occurrences of objects belonging to the same class (i.e. volcanoes). According to the generated saliency maps, both Grad-CAM and integrated gradients were able to highlight multiple objects belonging to the same class, while the occlusion method highlighted only a single object. This is due to the limitations of the occlusion method in generating accurate visual results. Occlusion continuously replaces part of the input image with a black patch and

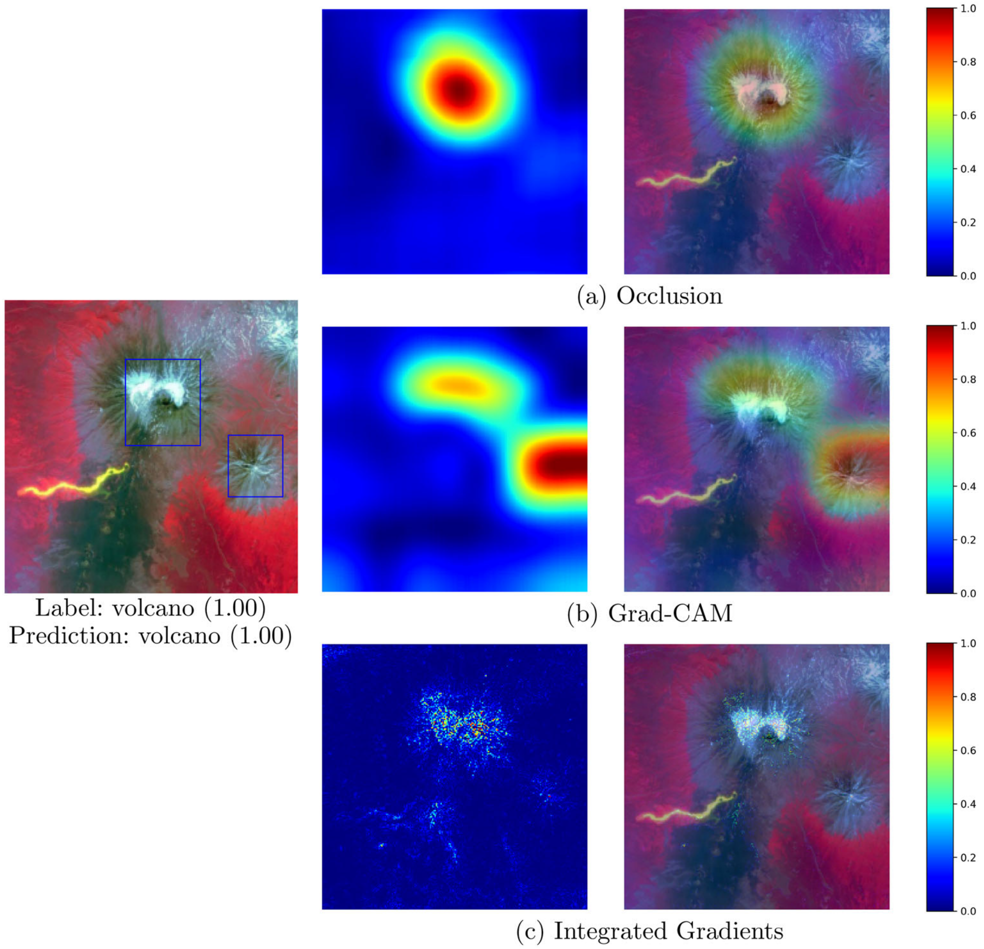

Figure 2. Saliency maps showing important image areas identified by the deep learning model. First column: original image with ground-truth and model prediction results. Second column: saliency map generated from (a) the occlusion method, (b) Grad-CAM, and (c) integrated gradients (red/dark color: high value, blue/light color: low value). Third column: saliency map overlaid on the original image.

calculates the change of the output probability. However, if there are multiple objects belonging to the same class in the image and one of them already saturates the output, the blocking of other recessive objects will not change the output probability. This result shows that the occlusion method cannot fully reflect how the model processes images containing multiple objects of the same class. Compared to the occlusion method, Grad-CAM combines multiple feature maps, wherein each map could contain features from different objects. The integrated gradients method adopts pixel-wise evaluation and integrates all results from the baseline image to the input image to avoid output saturation by a certain object. The results also illustrate the importance and value of using joint evaluation instead of a single approach.

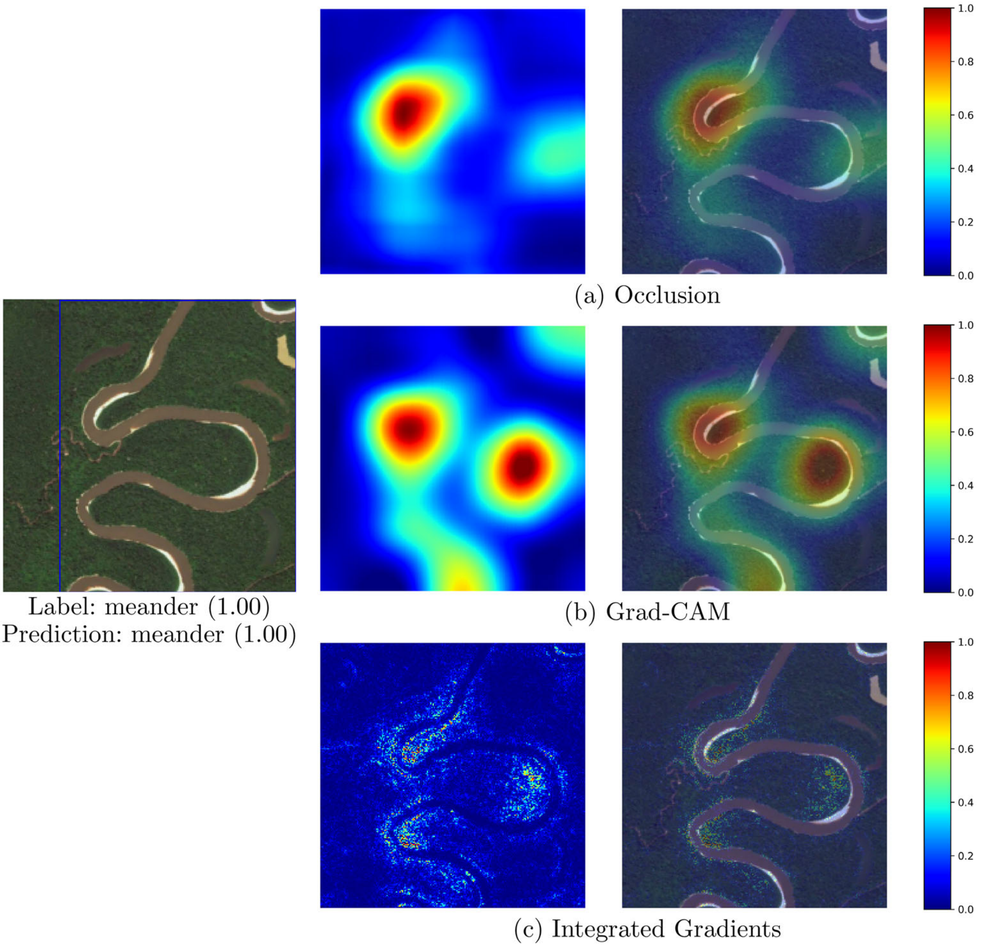

Figure 3. Ability of the saliency maps to identify multiple prominent features of the same object. Figure layout is the same as that in Figure 2. (a) Occlusion. (b) Grad-CAM. (c) Integrated gradients. Label: meander (1.00). Prediction: meander (1.00).

4.2. Ability to identify multiple prominent features of the same object

Figure 3 shows the generated saliency maps for detecting meanders in images. A *meander* is defined as 'a river [that] flows back and forth across the landscape to form a series of sinuous curves' (Charlton 2007). Hence, meanders are objects with multiple prominent features (i.e. curving parts). Similar to the detection of multiple objects (Figure 2), the results related to the detection of multiple prominent features of the same object showed that Grad-CAM (Figure 3(b)) and integrated gradients (Figure 3(c)) could highlight multiple prominent features that characterize a meander. In contrast, the results from the occlusion method showed that the classification depends mainly on the detection of a single curve. Even though the occlusion method detected two curving parts of the meander (Figure 3(a)), the curve near the center of the image received the most attention and garnered much more importance than did

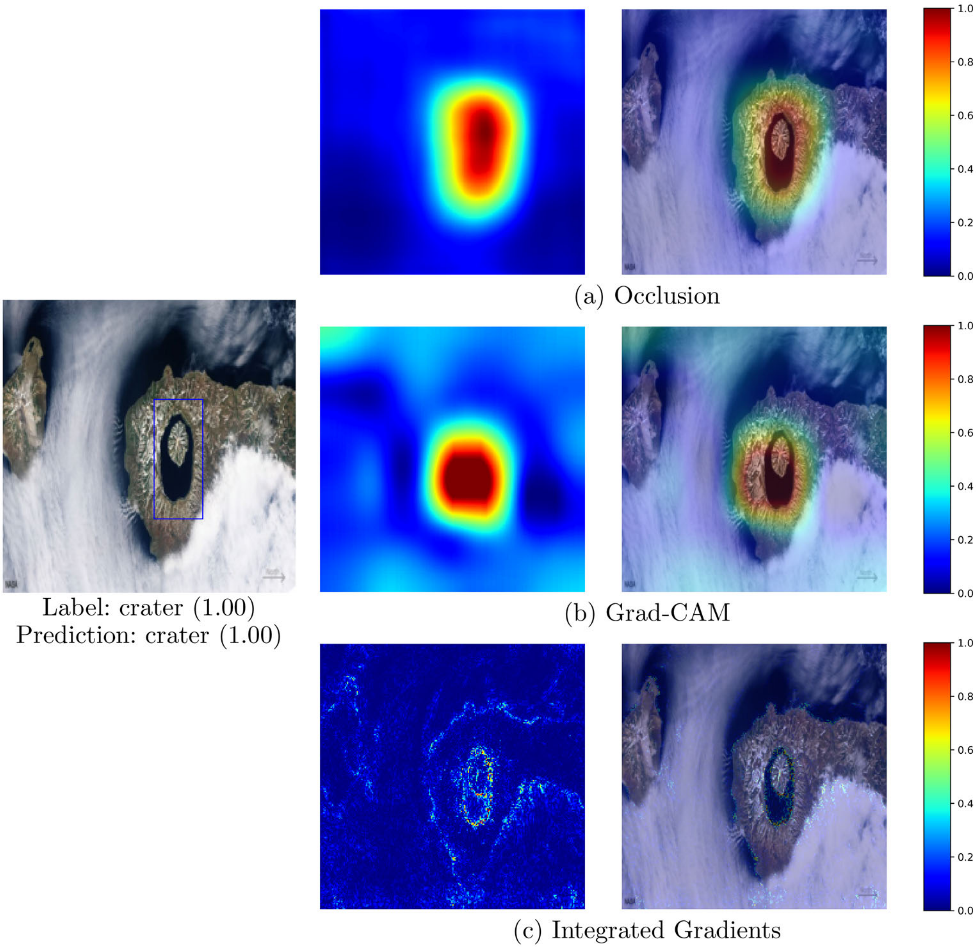

Figure 4. Accuracy of the saliency maps in highlighting the shape of the target objects. Figure layout is the same as that in Figure 2. (a) Occlusion. (b) Grad-CAM. (c) Integrated gradients. Label: crater (1.00). Prediction: crater (1.00).

the curve near the right-side edge of the image (Figure 3(a)). This is because, in the occlusion method, one prominent part of the object dominantly influences the result and saturates the output probability.

4.3. Ability to accurately capture object shape

Figure 4 shows a comparison of the different methods in correctly highlighting the shape of an oval shaped crater. Both the occlusion (Figure 4(a)) and the integrated gradients (Figure 4(c)) methods can generate highlights that match the actual shapes of the target (i.e. a crater with an oval shape, shown in Figure 4(a)) because they both calculate the importance of image regions at the pixel-level of the input image. As pixel values near the edges of the objects often have shape changes, the pixel-based saliency map generation process can relatively accurately capture the border pixels

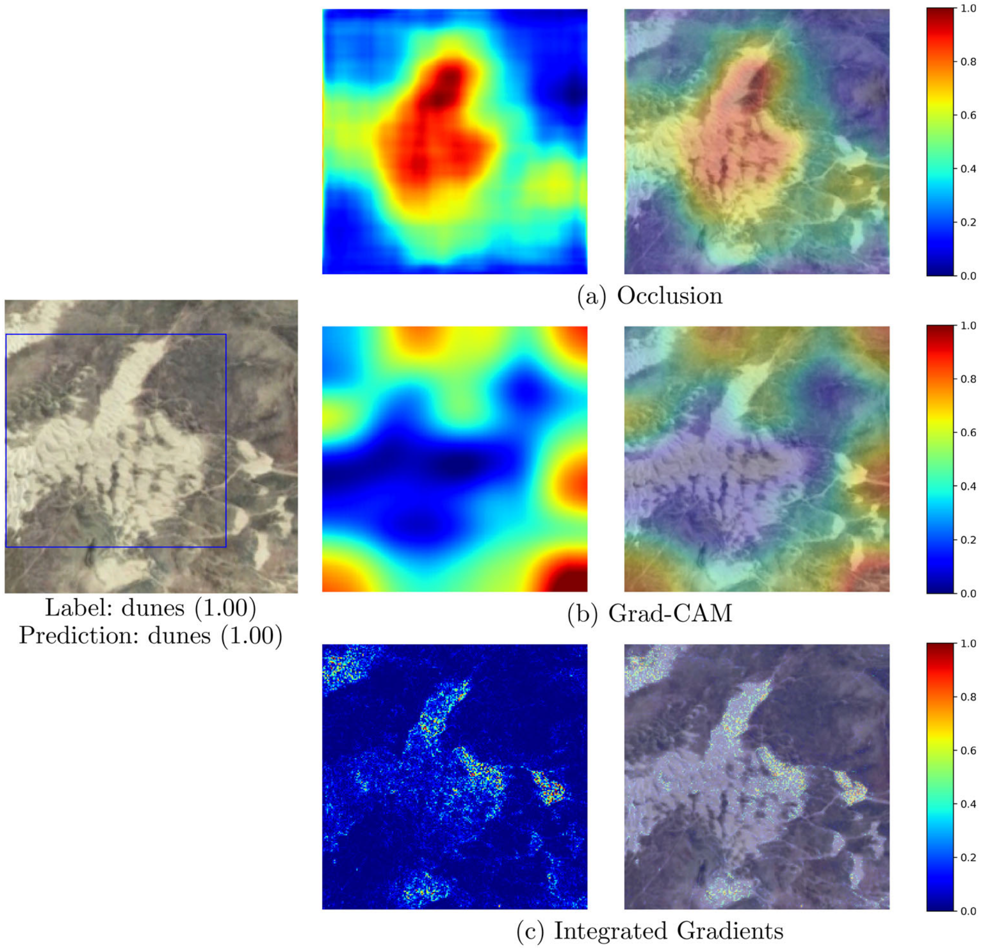

Figure 5. Accuracy of the saliency maps in highlighting important image areas containing the target objects. Figure layout is the same as that in Figure 2. (a) Occlusion. (b) Grad-CAM. (c) Integrated gradients. Label: dunes (1.00). Prediction: dunes (1.00).

and, hence, the shape of the target. Grad-CAM, however, may generate mismatching shapes between the highlighted image areas and the target shape in its visualization. As shown in Figure 4(b), a circular region was generated by Grad-CAM when interpreting the oval-shaped crater. This is because Grad-CAM combines feature maps from the last convolutional layer—which has a much lower resolution than does the original image—to generate the saliency map, the resolution of which is therefore lower than that generated by the other two methods. This coarse saliency map is often upsampled to the same size as the input image to gain a better visual effect. However, the upsampling may cause information loss, and some subtle differences (e.g. between a circular shaped area vs. an oval-shaped area) at a coarse resolution may become more substantial after upsampling, resulting in location and shape mismatches.

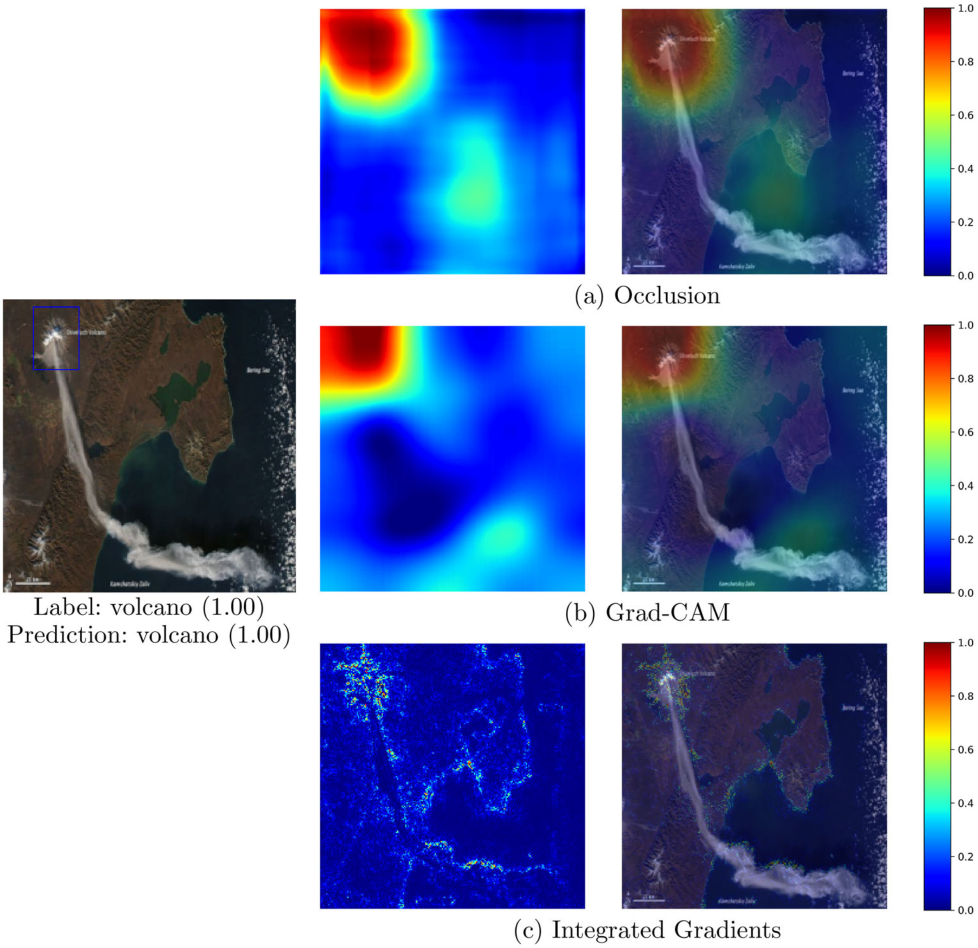

Figure 6. Comparison of the saliency maps in generating clear attention when multiple high-contrast areas exist. Figure layout is the same as that in Figure 2. (a) Occlusion. (b) Grad-CAM. (c) Integrated gradients. Label: volcano (1.00). Prediction: volcano (1.00).

4.4. Ability to correctly highlight important image regions

Figure 5 shows a comparison of the three methods in terms of their ability to correctly highlight important image regions in the resultant saliency maps. For this test, the object to detect was sand dunes. Results indicate that both the occlusion and the integrated gradients methods were able to highlight the targets correctly, and their results were similar. However, Grad-CAM highlighted surrounding areas of the target instead of the target itself. Analyzing the algorithm behaviors, we note that the large differences between the results generated by Grad-CAM and the other two methods may be caused by the issue of gradient discontinuity, a non-linear function used in the Grad-CAM model. As does ReLU, it introduces discontinuous gradients when they are non-differentiable at some locations of the function curves. Grad-CAM computes partial derivatives of the classification score with respect to each pixel at the last convolutional layer, and the discontinuity may transfer to some artifacts in the saliency

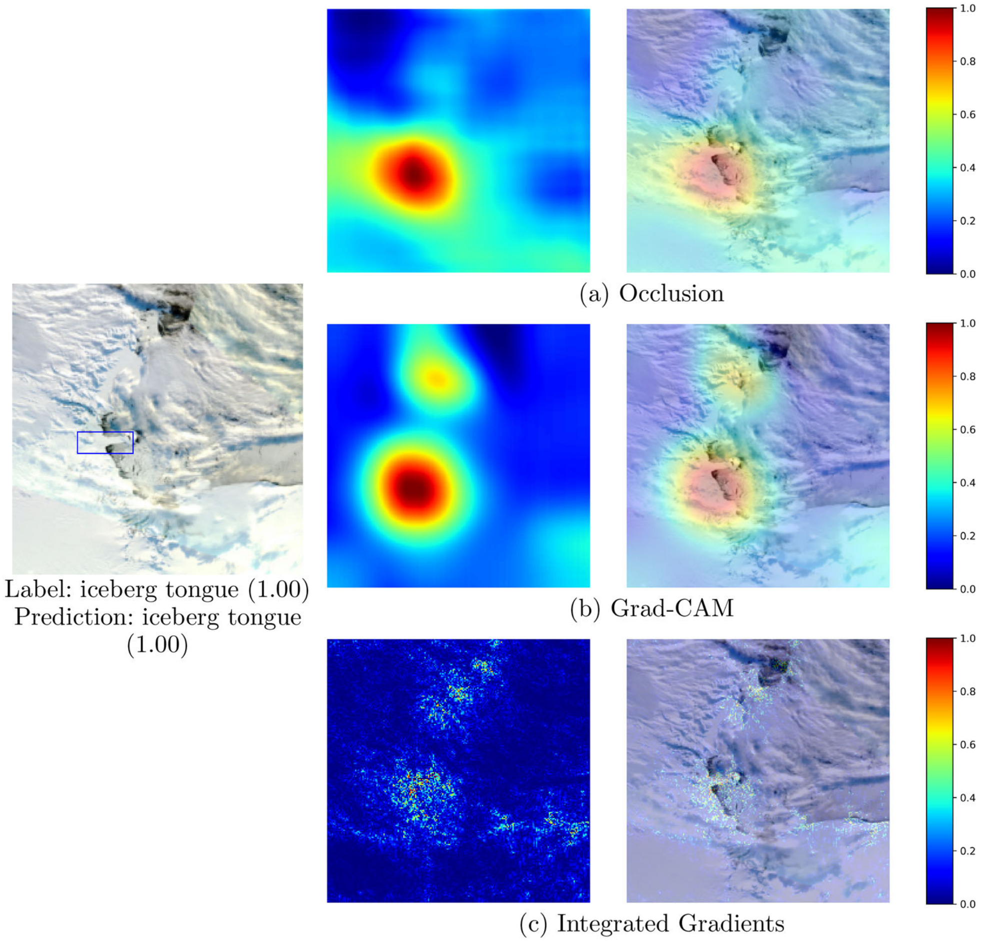

Figure 7. Comparison of the saliency maps in generating clear attention when the contrast between the foreground and background is low. Figure layout is the same as that in Figure 2. (a) Occlusion. (b) Grad-CAM. (c) Integrated gradients. Label: iceberg tongue (1.00). Prediction: iceberg tongue (1.00).

map. Unlike Grad-CAM, the occlusion and integrated gradients methods do not suffer from this issue. The occlusion method takes an image as the input and observes the output probability change in the forward pass, while the integrated gradients method computes and integrates all gradients from the baseline image to the input image to avoid the effect of discontinuous gradients. The results in Figure 5 further illustrate the advantages of using a joint analysis approach to avoid the possible pitfalls of using a single method.

4.5. Ability to generate clear areas of attention due to the existence of multiple high-contrast areas

Figure 6 shows the saliency maps generated by different methods when there are multiple high-contrast areas presented in an image. In this figure, an erupting volcano was the target feature to detect. In addition to the volcano, the image contained erupted gases forming a unique linear shape, which had a high level of contrast with the background. Meanwhile, the coastline in the image also presented a high level of contrast, forming the division between the water and the land. Because the integrated gradients approach uses per-pixel computation to generate pixel-wise highlights, it can more easily distinguish edges and even thin linear features, such as ridges or shorelines. For the case presented in Figure 6, the integrated gradients method highlighted multiple areas with such characteristics (Figure 6(c)). However, this result makes it difficult to examine which are the most distinctive image regions that guided the model to detect the target of interest (i.e. volcano). The occlusion and Grad-CAM approaches, in comparison, can more confidently and clearly highlight the most important image regions that the model relies on to make a correct prediction (i.e. the mouth of the volcano). This issue of the integrated gradients approach is also found in processing images when there is a low level of contrast between the target and the background, such as in the detection of an iceberg tongue in an image of a glacier.

4.6. Ability to generate clear attention due to low contrast between the foreground and background

Figure 7 further illustrates the contrast-related issues of the integrated gradients method, which results in unclear patterns when the contrast between the foreground (i.e. the target) and the background of an image is low. As discussed, the integrated gradients approach can generate a saliency map at the pixel level, providing a fine-grained view of the important pixels and image subareas identified by a model when making decisions. However, under some conditions, the highlighted pixels could spread across the image, making the explanation of the model's decision process difficult to achieve. For instance, Figure 7 presents an iceberg tongue feature within a glacier region. Because an iceberg tongue is often a part of a glacier, and they are both white, there tends to be a low level of contrast between the two features. The integrated gradients approach generates pixel-level highlights based on the existence of shades, which may be randomly distributed in the image. Accordingly, even though the iceberg tongue area has been highlighted in the result (Figure 7(c)), other areas are also highlighted, making it difficult to understand and explain the model's learning process. In comparison, the occlusion and the Grad-CAM methods create saliency maps with region-based patterns containing less noise, providing a clearer view on the important feature and contextual information that helps the model with its prediction.

5. Research findings

5.1. A Summary analysis of CNN model explanation methods

Table 1 summarizes the capabilities of the model explanation methods. We also generate the statistics (Table 2) based on the entire training set by counting the number of saliency maps generated by these methods that can meet an explanation goal. The

Table 1. Summary analysis of the capabilities of the CNN model explanation methods.

Model explanation goals	Occlusion approach	Grad-CAM	Integrated gradients
Ability to identify multiple objects of the same type	Often no	Often yes	Often yes
Ability to identify multiple prominent features of a single object	Often no	Often yes	Often yes
Ability to accurately detect object shape	Often yes	Sometimes no	Often yes
Ability to correctly highlight important image regions	Often yes	Sometimes no	Often yes
Ability to generate clear attentions for an image with multiple high-contrast areas	Often yes	Often yes	Sometimes no
Ability to generate clear attentions for an image when the contrast between the target and the background is low	Often yes	Often yes	Sometimes no
Computational efficiency	Low	High	Low
Need to retrain the model	No	No	No

Table 2. Statistics on percentage of saliency maps generated by different methods that can meet an explanation goal.

Model explanation goals	Occlusion approach (%)	Grad-CAM (%)	Integrated gradients (%)	Total
Ability to identify multiple objects of the same type	30	71	79	73
Ability to identify multiple prominent features of a single object	31	81	89	214
Ability to accurately detect object shape	76	44	79	826
Ability to correctly highlight important image regions	81	58	87	826
Ability to generate clear attentions for an image with multiple high-contrast areas	86	87	53	207
Ability to generate clear attentions for an image when the contrast between the target and the background is low	82	78	50	152

'Total' means number of images in the dataset that supports the corresponding criterion listed in each row.

cells in Table 1 are labelled as 'Often Yes' when a method (e.g. Grad-CAM) meets the goal in most cases; they are labelled as 'Often No' when a method does not meet an explanation goal in most testing cases. A 100% 'Yes' case does not exist due to the inherent uncertainties in the model explanation process for calculating attributed features and the inconsistency between model perception and human perception.

In summary, Grad-CAM can highlight multiples of the same type of object and multiple prominent features of a single object. It can also generate clear attention by highlighting important image regions when there are multiple high-contrast areas in an image or when the contrast between the foreground (i.e. the target) and background is low. However, it sometimes cannot accurately reveal the object shape due to the saliency map's low resolution. Comparatively, the occlusion and integrated gradient approaches can better reveal an object's actual shape as they generate the saliency map at the input images' resolutions. The pixel-based computation adopted in the integrated gradient approach makes it capable to identify multiple objects or multiple prominent features of an object. However, when an image has multiple high-contrast areas, this method may highlight all of them, generating noise in the saliency map. Although the occlusion approach avoids highlighting multiple high-contrast areas if they belong to different object classes, the gradient saturation issue makes it favor only one of multiple objects of the same class or one prominent feature of an object.

The Grad-CAM method has the advantage of greater computational efficiency because it requires only a single forward and partial backward computation per image. In contrast, the occlusion and the integrated gradients methods need multiple forward and backward calculations. Their computation efforts depend on corresponding hyper-parameters, which incur a trade-off between saliency map quality and computation time. Compared to methods such as CAM (Zhou *et al.* 2016), none of the methods studied in this paper require network architecture changes; therefore, there is no need to retrain the deep learning models.

5.2. Results generalizability

To demonstrate the generalizability of our results, we adopted another AI-ready natural feature dataset—GeoNat v1.0 (Arundel *et al.* 2020). This dataset contains 10 types of natural features (basin, bay, bend, crater, gap, gut, island, lake, ridge, and valley), and we selected five (bay, bend, crater, island, and lake) features and a total of 540 images for the experiments. Some features were not selected because they are uncommon (e.g. gap and gut), and some (e.g. basin, valley, and ridge) are extremely difficult to be visually inspected using optical remote sensing images alone due to their limited color representations. This five-category GeoNat dataset is trained using the same experimental setting as the first dataset. The saliency maps are generated on a trained VGG model with 98.89% training accuracy and 82.52% testing accuracy. The results reported in Appendix Table A1 show that despite small differences, the findings about the GeoAI models' characteristics using the two datasets are quite consistent. This further supports the conclusion drawn in Table 1.

6. Conclusion

As AI becomes an important tool for high-stakes decision-making, its explainability—the articulation of an AI's algorithm's rationale in deriving an answer—has become a critical research topic because it can help open the black box and help users gain the confidence and trust to adopt AI in real-world decision-making processes (Phillips *et al.* 2020). However, little research has assessed the capabilities of existing AI model explanation methods and their applicability in geospatial applications. This paper fills this knowledge gap by providing an in-depth analysis of existing methods' mechanisms, especially those aiming to explain a deep learning model by capturing the 'visual attention' of a machine in a saliency map. Multiple methods, including occlusion, Grad-CAM, and integrated gradients, were implemented and applied to a deep learning model for image classification tasks. By examining the decisions the model made for every single image of the training dataset (indicated in the saliency maps), we derived and summarized a number of model explanation goals (e.g. the ability to identify multiple objects of the same type) and assessed each method's ability to meet such goals. The experiments used two natural feature datasets, and the results derived from these datasets are highly consistent, demonstrating good generalizability of the research findings in natural feature analysis.

The results in Table 1 show that no single AI explanation method can achieve all the explanation goals in a geospatial task. Therefore, simultaneously applying multiple methods when attempting to explain a GeoAI model is important so that the results can be cross-validated and uncertainties can be removed. The term 'uncertainty' is used to describe situations where a saliency map contains artifacts that do not reflect the model's learned knowledge but are instead caused by the inherent limitations of the model explanation methods (e.g. gradient saturation). Furthermore, while two datasets were used in the experiments to demonstrate the generalizability of our research findings, we cannot conclusively establish their universal generalizability without conducting more systematic experiments. This limitation underscores the need for further improvement in this area of research.

In the future, in addition to increasing the results accuracy of the GeoAI explanation methods, we will work to combine the global and per-decision model explanation algorithms and develop new strategies to support the automatic extraction of semantically meaningful concepts from saliency maps to further enhance the human understanding of machine learning processes. We will compare the consistency between human-understandable concepts defined in domain knowledge graphs ((Li *et al*. 2023, Janowicz *et al*. 2022, Li *et al*. 2012)) and the machine-learned features to further improve the explainability of the machine learning process. We will also use these methods to analyze the failure cases and understand why a GeoAI model makes a wrong decision. As discussed, detection of some natural features—especially those existing in hilly terrains, such as ridges and valleys—is often difficult to achieve using satellite images alone. This task can benefit from the use of additional data sources, such as digital elevation models (DEMs) data (Wang and Li 2021, Li *et al*. 2022b). Hence, our future work will also include extending the application of deep learning model explanation algorithms to a multisource learning framework. Finally, more experiments will be conducted to further verify the generalizability of our research findings using additional, more diverse datasets, including both natural and artificial features.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work is supported in part by the National Science Foundation under [awards 2120943, 2230034, 1853864].

Notes on contributors

Chia-Yu Hsu is a research professional at Arizona State University. His research interests include artificial intelligence, computer vision, spatiotemporal data analysis, and their applications in climate change and terrain research.

Wenwen Li is a professor in geographic information science at Arizona State University (ASU). Her research interests are cyberinfrastructure, big data, GeoAI and their applications in data-

and computation-intensive environmental and social sciences. At ASU, she directs the Cyberinfrastructure and Computational Intelligence Lab (http://cici.lab.asu.edu/) and serves as the Research Director for the Spatial Analysis Research Center.

Data and codes availability statement

The data and codes that support the findings of this study are available at https://github.com/ASUcicilab/explainable-geoai. Instructions on how to use the data and codes are provided in the README file.

References

Adebayo, J., *et al.*, 2018. Sanity checks for saliency maps. *Advances in Neural Information Processing Systems*, 31, 9525–9536.

Arundel, S.T., Li, W., and Wang, S., 2020. Geonat v1. 0: A dataset for natural feature mapping with artificial intelligence and supervised learning. *Transactions in GIS*, 24 (3), 556–572.

Buscombe, D., and Ritchie, A.C., 2018. Landscape classification with deep neural networks. *Geosciences*, 8 (7), 244.

Charlton, R., 2007. *Fundamentals of fluvial geomorphology*. New York, NY: Routledge.

Chattopadhay, A., *et al.*, 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. *In*: A. Hoogs, S. McCloskey and G. Medioni, eds. *2018 IEEE winter conference on applications of computer vision* (WACV). New York, NY: IEEE, 839–847. doi: 10.1109/WACV.2018.00097.

Duckham, M., *et al.*, 2022. Explainable spatiotemporal reasoning for geospatial intelligence applications. *Transactions in GIS*, 26 (6), 2455–2479.

EarthEclipse. 2022. *What is a sand dune: formation and types of sand dunes*. Available from: https://eartheclipse.com/science/geology/sand-dune-formation-types.html.

Erion, G., *et al.*, 2021. Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, 3 (7), 620–631.

Fong, R., and Vedaldi, A., 2017. Interpretable explanations of Black Boxes by meaningful perturbation. *In*: K. M. Lee, *et al.*, eds. 2017 IEEE *international conference on computer vision (ICCV)*. New York, NY, 3449–3457.

Fong, R., Patrick, M., and Vedaldi, A., 2019. Understanding deep networks via extremal perturbations and smooth masks. In: 2019 IEEE/CVF *international conference on computer vision* (ICCV), October. Seoul, Korea (South): IEEE, 2950–2958.

Gebru, T., *et al.*, 2017. Using deep learning and Google Street View to estimate the demographic makeup of neighborhoods across the United States. *Proceedings of the National Academy of Sciences*, 114 (50), 13108–13113.

Goodchild, M.F., and Li, W., 2021. Replication across space and time must be weak in the social and environmental sciences. *Proceedings of the National Academy of Sciences*, 118 (35), e2015759118.

Gunning, D., and Aha, D.A., 2019. Darpa's explainable artificial intelligence program. *AI Magazine*, 40 (2), 44.

Helber, P., *et al.*, 2019. Eurosat: a novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12 (7), 2217–2226.

Herzfeld, U.C., 2004. *Atlas of antarctica: Topographic maps from geostatistical analysis of satellite radar altimeter data: with 169 figures*. Heidelberg, Germany: Springer Science & Business Media.

Hesse, R., Schaub-Meyer, S., and Roth, S., 2021. Fast axiomatic attribution for neural networks. *Advances in Neural Information Processing Systems*, 34 (2021), 19513–19524.

Hsu, C.Y., Li, W., and Wang, S., 2021. Knowledge-driven GeoAI: Integrating spatial knowledge into multi-scale deep learning for mars crater detection. *Remote Sensing*, 13 (11), 2116.

Janowicz, K., *et al.*, 2022. Know, know where, knowwheregraph: a densely connected, cross-domain knowledge graph and geo-enrichment service stack for applications in environmental intelligence. *AI Magazine*, 43 (1), 30–39.

Kedron, P., *et al.*, 2021. Reproducibility and replicability: opportunities and challenges for geospatial research. *International Journal of Geographical Information Science*, 35 (3), 427–445.

Kim, B., *et al.*, 2018. Interpretability beyond feature attribution: quantitative testing with concept activation vectors (TCAV). *In*: J. Dy and A. Krause, eds. *International conference on machine learning*. Stockholm, Sweden: PMLR, 2668–2677.

Koo, B.W., Guhathakurta, S., and Botchwey, N., 2022. How are neighborhood and street-level walkability factors associated with walking behaviors? A big data approach using street view images. *Environment and Behavior*, 54 (1), 211–241.

Kurth, T., *et al.*, 2018. Exascale deep learning for climate analytics. *In*: R.A. McEldowney. T. Damkroger and M. Taufer, eds. *SC18: International conference for high performance computing, networking, storage and analysis*. Dallas, TX: IEEE, 649–660.

Li, W., 2020. GeoAI: Where machine learning and big data converge in GIScience. *Journal of Spatial Information Science*, 2020 (20), 71–77.

Li, W., 2022a. GeoAI in social science. *In*: S. Rey and R. Franklin, eds. *Handbook of spatial analysis in the social sciences*. Cheltenham, UK: Edward Elgar, 291–304.

Li, W., *et al.*, 2017. Recognizing terrain features on terrestrial surface using a deep learning model: An example with crater detection. *In*: H. Mao, *et al.*, eds. *Proceedings of the 1st workshop on artificial intelligence and deep learning for geographic knowledge discovery*. New York, NY: ACM, 33–36.

Li, W., *et al.*, 2022a. Real-time GeoAI for high-resolution mapping and segmentation of arctic permafrost features: the case of ice-wedge polygons. *In*: D. Lunga and S. Newsam, eds. *Proceedings of the 5th ACM SIGSPATIAL international workshop on AI for geographic knowledge discovery*. New York, NY: ACM, 62–65.

Li, W., *et al.*, 2022b. GeoImageNet: a multi-source natural feature benchmark dataset for GeoAI and supervised machine learning. *GeoInformatica*, 2022, 1–22.

Li, W., *et al.*, 2023. Geographvis: a knowledge graph and geovisualization empowered cyberinfrastructure to support disaster response and humanitarian aid. *ISPRS International Journal of Geo-Information*, 12 (3), 112.

Li, Z., 2022b. Extracting spatial effects from machine learning model using local interpretation method: an example of SHAP and XGBoost. *Computers, Environment and Urban Systems*, 96, 101845.

Li, W., and Arundel, S.T., 2022. GeoAI and the future of spatial analytics. *In*: B. Li, *et al.*, eds. *New thinking in giscience*. Singapore: Springer, 151–158.

Li, W., and Hsu, C.Y., 2020. Automated terrain feature identification from remote sensing imagery: a deep learning approach. *International Journal of Geographical Information Science*, 34 (4), 637–660.

Li, W., and Hsu, C.Y., 2022. GeoAI for large-scale image analysis and machine vision: recent progress of artificial intelligence in geography. *ISPRS International Journal of Geo-Information*, 11 (7), 385.

Li, W., Hsu, C.Y., and Hu, M., 2021. Tobler's first law in GeoAI: a spatially explicit deep learning model for terrain feature detection under weak supervision. *Annals of the American Association of Geographers*, 111 (7), 1887–1905.

Li, W., Raskin, R., and Goodchild, M.F., 2012. Semantic similarity measurement based on knowledge mining: an artificial neural net approach. *International Journal of Geographical Information Science*, 26 (8), 1415–1435.

Lin, M., Chen, Q., and Yan, S., 2014. Network in network. *In*: Y. Bengio and Y. LeCun, eds. *2nd International conference on learning representations, ICLR 2014, conference track proceedings*, April 14–16, 2014. Banff, AB, Canada. https://sites.google.com/site/representationlearning2014/program-details/publication-model

Lundberg, S.M., and Lee, S.I., 2017. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30 (2017), 4768–4777.

McEwen, R.B., Witmer, R.E., and Ramey, B.S., 1983. *USGS digital cartographic data standards*. vol. 2. Rolla, MO: US Department of the Interior, Geological Survey.

National Geographic. 2022. *Hill*. Available from: https://education.nationalgeographic.org/resource/hill.

Nguyen, A., Yosinski, J., and Clune, J., 2015. Deep neural networks are easily fooled: high confidence predictions for unrecognizable images. *In*: 2015 IEEE *conference on computer vision and pattern recognition* (CVPR), June. Boston, MA, USA: IEEE, 427–436.

Petsiuk, V., Das, A., and Saenko, K., 2018. RISE: randomized input sampling for explanation of black-box models. In: L. Shao, H.P.H. Shum and T. Hospedales, eds. *British machine vision conference* 2018, BMVC 2018*, September 3*–6*, 2018*. Newcastle, UK, 151. http://bmvc2018.org/index.html

Phillips, P.J., *et al.*, 2020. *Four principles of explainable artificial intelligence*. Gaithersburg, Maryland: NIST.

Purcell, A., 2018. *Basic biology: An introduction*. Cambridge, New Zealand: Basic Biology Ltd.

Ribeiro, M.T., Singh, S., and Guestrin, C., 2016. "Why should I trust you?": explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD *international conference on knowledge discovery and data mining*, August. San Francisco, California, USA: ACM, 1135–1144.

Rieger, L., *et al.*, 2020. Interpretations are useful: penalizing explanations to align neural networks with prior knowledge. *In*: L. Rieger, *et al.*, eds. *International conference on machine learning*. PMLR, 8116–8126. http://proceedings.mlr.press/v119/rieger20a.html

Selvaraju, R.R., *et al.*, 2020. Grad-CAM: visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128 (2), 336–359.

Shrikumar, A., Greenside, P., and Kundaje, A., 2017. Learning important features through propagating activation differences. *In*: D. Precup and Y. W. Teh, eds. *Proceedings of the 34th international conference on machine learning*, July. PMLR, 3145–3153. https://proceedings.mlr.press/v70/

Simonyan, K., and Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. *In*: Y. Bengio and Y. LeCun, eds. *International conference on learning representations*. San Diego, CA, USA. https://iclr.cc/archive/www/doku.php%3Fid=iclr2015:accepted-main.html

Simonyan, K., Vedaldi, A., and Zisserman, A., 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. *In*: A. Couville, R. Fergus and B. Kingsbury, eds. *Workshop at international conference on learning representations*. Citeseer. https://sites.google.com/site/representationlearning2014/workshop-proceedings

Springenberg, J.T., *et al.*, 2015. Striving for simplicity: the all convolutional net. *In*: Y. Bengio and Y. LeCun, eds. 3rd International *conference on learning representations, ICLR 2015, workshop track proceedings,* May 7–9, 2015. San Diego, CA, USA. https://dblp.org/db/conf/iclr/iclr2015.htm

Sundararajan, M., Taly, A., and Yan, Q., 2017. Axiomatic attribution for deep networks. *In*: D. Precup and Y.W. Teh, eds. *Proceedings of the 34th international conference on machine* learning, July. PMLR, 3319–3328. https://proceedings.mlr.press/v70/

Wang, H., *et al.*, 2020. Score-CAM: score-weighted visual explanations for convolutional neural networks. *In*: T. Boult, G. Medioni and R. Zabih, eds. Proceedings of the IEEE/CVF *conference on computer vision and pattern recognition workshops*. New York, NY: IEEE, 24–25.

Wang, S., and Li, W., 2021. GeoAI in terrain analysis: enabling multi-source deep learning and data fusion for natural feature detection. *Computers, Environment and Urban Systems*, 90, 101715.

Wikipedia, 2022a. *Meander—Wikipedia, the free encyclopedia*. http://en.wikipedia.org/w/index.php?title=Meander&oldid=1121852800 [Online; accessed 22 Nov 2022].

Wikipedia, 2022b. *River—Wikipedia, the free encyclopedia*. http://en.wikipedia.org/w/index.php?title=River&oldid=1123171885 [Online; accessed 22 Nov 2022].

Wikipedia, 2022c. *Volcano—Wikipedia, the free encyclopedia*. http://en.wikipedia.org/w/index.php?title=Volcano&oldid=1121909395 [Online; accessed 22 Nov 2022].

Xing, J., and Sieber, R., 2021. Integrating XAI and GeoAI. *In*: K. Janowicz and J.A. Verstegen, eds. *GIScience 2021 short paper proceedings*. 11th International *conference on geographic information science*, September 27–30, 2021. Poznań, Poland. Wadern, Germany: ACM.

Zeiler, M.D., and Fergus, R., 2014. Visualizing and understanding convolutional networks. *In*: D. Fleet, *et al*., *European conference on computer vision*. New York, NY: Springer, 818–833.

Zhang, W., *et al*., 2019. Application of multi-channel 3D-cube successive convolution network for convective storm nowcasting. *In*: R. Barga and C. Zaniolo, eds. *2019 IEEE international conference on big data (big data*). New York, NY: IEEE, 1705–1710.

Zhou, B., *et al*., 2016. Learning deep features for discriminative localization. In: 2016 IEEE *conference on computer vision and pattern recognition* (CVPR), June. Las Vegas, NV, USA: IEEE, 2921–2929.

Appendix

Table A1. Statistics on percentage of saliency maps generated by different methods that can meet an explanation goal.

Model explanation goals	Occlusion approach (%)	Grad-CAM (%)	Integrated gradients (%)	Total
Ability to identify multiple objects of the same type	29	74	81	119
Ability to identify multiple prominent features of a single object	32	86	94	142
Ability to accurately detect object shape	78	48	81	540
Ability to correctly highlight important image regions	84	60	89	540
Ability to generate clear attentions for an image with multiple high-contrast areas	89	82	52	132
Ability to generate clear attentions for an image when the contrast between the target and the background is low	84	81	52	204.

'Total' means number of images in the GeoNat v1.0 dateset (Arundel et al. 2020) that supports the corresponding criterion listed in each row.