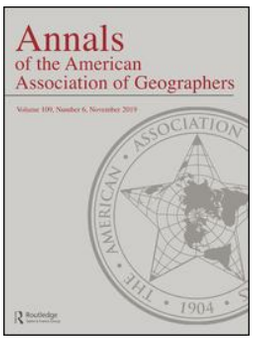

Annals of the American Association of Geographers

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/raag21

Tobler's First Law in GeoAI: A Spatially Explicit Deep Learning Model for Terrain Feature Detection under Weak Supervision

Wenwen Li, Chia-Yu Hsu & Maosheng Hu

To cite this article: Wenwen Li, Chia-Yu Hsu & Maosheng Hu (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, DOI: 10.1080/24694452.2021.1877527

To link to this article: https://doi.org/10.1080/24694452.2021.1877527

Published online: 23 Apr 2021.

Tobler's First Law in GeoAI: A Spatially Explicit Deep Learning Model for Terrain Feature Detection under Weak Supervision

Wenwen Li,[*] Chia-Yu Hsu,[*] and Maosheng Hu[†]

[*]*School of Geographical Sciences and Urban Planning, Arizona State University*
[†]*School of Geography and Information Engineering, China University of Geosciences*

Recent interest in geospatial artificial intelligence (GeoAI) has fostered a wide range of applications using artificial intelligence (AI), especially deep learning for geospatial problem solving. Major challenges, however, such as a lack of training data and ignorance of spatial principles and spatial effects in AI model design remain, significantly hindering the in-depth integration of AI with geospatial research. This article reports our work in developing a cutting-edge deep learning model that enables object detection, especially of natural features, in a weakly supervised manner. Our work has made three innovative contributions: First, we present a novel method of object detection using only weak labels. This is achieved by developing a spatially explicit model according to Tobler's first law of geography to enable weakly supervised object detection. Second, we integrate the idea of an attention map into the deep learning–based object detection pipeline and develop a multistage training strategy to further boost detection performance. Third, we have successfully applied this model for the automated detection of Mars impact craters, the inspection of which often involved tremendous manual work prior to our solution. Our model is generalizable for detecting both natural and man-made features on the surface of the Earth and other planets. This research has made a major contribution to the enrichment of the theoretical and methodological body of knowledge of GeoAI.
Key Words: deep learning, GeoAI, object detection, remote sensing, terrain feature, weakly supervised.

Machine learning represents an exciting new research area in artificial intelligence (AI) that incorporates machine intelligence and data-driven approaches for geospatial problem solving. Rapid advances in AI methods, the proliferation of spatial big data, and the increasing availability of computing power are transforming the way we conduct geospatial research and prompting new discoveries. Geospatial artificial intelligence (GeoAI) has emerged as a new research area that tackles data- and computation-intensive problems leveraging AI and geospatial big data (W. Li 2020). One key topic in GeoAI research is spatial object detection, the task to distinguish features, either natural or man-made, in remote sensing and other forms of images. The derivation of such information will greatly enrich existing gazetteers of both named and unnamed features, thereby advancing our spatial knowledge about the Earth and other planets (Hill, Frew, and Zheng 1999; Goodchild and Hill 2008; Zhu et al. 2020). Feature detection is also playing an important role in urban planning (Kamusoko 2017), environmental management (Barrett and Petropoulos 2013), search and rescue operations (Bejiga et al. 2017), and inspection of the living conditions of refugee camps (Tomaszewski et al. 2016).

Modern terrain analysis can also greatly benefit from advances in GeoAI. Traditional approaches often involve the use of object-based image analysis (OBIA) and some shallow machine learning techniques, such as support vector machine or random forest, to find, segment, and classify objects of interest in an image scene. Although proven successful with different detection problems (Jasiewicz and Stepinski 2013; Micheletti et al. 2014; Arundel, Li, and Zhou 2018), these approaches inevitably require manual work; for instance, manual determination of attributes or terrain factors that are important for distinguishing landform features of different types. In OBIA, data processing parameters such as scale factor, which controls the segmentation granularity, and the strategies for merging segmented super-pixels (a cluster of pixels with similar values), also need to be manually or semiautomatically selected. The breakthroughs in machine learning, especially the

Annals of the American Association of Geographers, 0(0) 2021, pp. 1–19
Initial submission, July 2020; revised submission, September 2020; final acceptance, November 2020
Published by Taylor & Francis, LLC.

rapid development of deep learning techniques, have offered ample opportunity to revolutionize the terrain analysis paradigm toward operating in a more intelligent and automated manner (W. Li et al. 2017).

One significant challenge in applying AI and deep learning for natural feature detection, however, is the lack of proper training data. This is due to the high cost and level of expertise required for collecting training data of good quality. A survey shows that among the available geospatial benchmark databases (Yang and Newsam 2010; Cheng, Han, and Lu 2017; W. Li et al. 2017; Xia et al. 2017; W. Zhou et al. 2018), only four contain terrain features of limited types. Moreover, existing data sets are primarily used for scene classification instead of object detection. They contain only labels of object types in each image; the extent of each object is not delineated. A second major challenge is that many deep learning applications using geospatial data are simply an importation of methods from computer science to geography. There is very limited work toward methodology innovation that incorporates spatial principles and the unique characteristics of spatial data to supervise the learning processes.

Spatial theories, however, such as Tobler's First Law of geography (TFL) should be taken deeply into account in studies related to space and place (Goodchild 2004). TFL states, "Everything is related to everything else, but near things are more related than distant things." Essentially, it is a perfect, informal description of spatial autocorrelation, the intrinsic relationships between geographical entities. In fact, TFL has been guiding the design of many spatial methods, such as local indicators of spatial autocorrelation (LISA; Anselin 1995) and geographically weighted regression (Fotheringham, Brunsdon, and Charlton 2003). To further exert the value of GeoAI and deep learning in geospatial research in general and terrain analysis in particular, this new machine learning–based paradigm needs to be revamped to deeply integrate spatial theory and laws, such as TFL.

To address these challenges, we developed a novel deep learning model to enable weakly supervised object detection (WSOD). A WSOD task aims to achieve object detection with only weak labels. Compared to strong supervision, which requires object-level annotation (object class and bounding box within each training image), weakly supervised learning only requires image-level annotation (object classes or an object count within a training image). Previous studies have shown that marking weak labels takes less than 5 percent of the time needed for marking strong labels (M. Gao et al. 2018). Predicting object location without explicitly providing this information, however, will require the model to be smarter than strongly supervised object detection models. Our proposed solution accomplishes this goal by injecting TFL into the deep learning model design to achieve good detection performance with a low labeling cost. The novel proposed strategy that explicitly models the spatial relations, particularly spatial autocorrelation, allows us to tackle both problems (object detection and lack of training data) within one solution framework. In addition, we integrated a machine vision strategy with an attention map and devised comprehensive training strategies to further boost the WSOD performance. Experiments were conducted using a large Mars crater data set and a natural feature data set on Earth, and the results showed that our proposed approach achieved state-of-the-art performance in WSOD.

The remainder of this article is organized as follows. We first review the current WSOD in both computer vision and the geography community. We then provide a formalized definition of the problem. After that, we introduce how spatial explicitness is achieved in the model design and then describe our methodology in detail. We describe the experimental setting and results and the discuss future research directions and conclusions.

Literature Review

Methods for image analysis and natural feature detection can be classified into two broad categories: knowledge driven and data driven. Next we provide an overview of the two approaches and discuss in detail the new deep learning technique and its usage in remote sensing.

Knowledge-Driven Terrain and Image Analysis

A knowledge-driven approach leverages prior knowledge or existing patterns to design an algorithm that performs various tasks. An advantage of such approaches is that they are based on a top-down, expertise-driven design pattern, so the data processing workflow (i.e., how it works and why it works) is well reasoned and often transparent and is

therefore easy to explain. Specifically, current terrain analysis leverages different techniques to identify natural features in image data. For instance, thresholding is a commonly used method in analyzing terrain. It can be considered a binary classification process, in which a threshold is applied to terrain parameters, such as elevation, slope, curvature, or the numerical difference between them, to help classify a pixel into an object or a nonobject (Blaschke 2010). Stream or drainage network analysis is another popular approach for extracting terrain features. By simulating how water accumulates at lower elevations in a watershed, a stream network can be created. Using the same approach, the valleyline features can be extracted from a digital elevation model, and so can ridgelines. The only difference is that for ridgeline extraction the elevation values need to be reversed first to form a pseudo-watershed (Lindsay and Dhun 2015). In recent years, spatial–contextual-based analysis, which formulates the spatial relationships between nearby pixels and the contextual background of a spatial object, has become an important form of expert knowledge that further improves OBIA for feature segmentation. X. Zhou, Li, and Arundel (2019) developed a visual descriptor that captures the spatial–contextual pattern through a probabilistic model to quantify the patterns of linear terrain features on a digital elevation model, and the approach has proven to be effective in identifying ridges and valleys in mountainous regions.

Although popular, these knowledge-driven approaches have been facing great challenges in terrain analysis, especially when dealing with big and high-resolution data. First, terrain features are complex, and they often possess intercategory similarity and intracategory heterogeneity. In addition, the same terrain feature might demonstrate different characteristics in different landscapes, making it very difficult to extract a common pattern to support its identification. Second, super-high-resolution and high-resolution terrain data, which have become increasingly available, also present major challenges to existing methods. Although these data provide more details about the terrain, the high-resolution data also capture more "noise," or local terrain spikes due to vegetation or a rocky surface. Existing methods, which are designed to process "smoothed" data, have shown limitations in handling data with uncertainties. Third, the knowledge-driven approaches often need accumulations of knowledge over a long time. They therefore require human intervention and are difficult to fully automate. To alleviate the aforementioned issues, data-driven approaches, such as AI and machine learning, have attracted researchers' close attention and become the fourth paradigm in scientific research (Gahegan 2020).

Data-Driven Object Detection Leveraging Weakly Supervised Object Detection in Computer Vision

In recent years, deep learning has emerged as the cutting-edge machine learning framework for object detection. It is known as the outstanding ability to learn context features of real-world objects from large quantities of labeled data. Typically, detectors are trained under strong supervision with object-level annotations. It is very time consuming for drawing bounding boxes (BBOX) manually, however. On the other hand, object detectors can also be trained under weak supervision using only image-level labels, the collection cost of which is substantially lower than obtaining object-level labels. The most popular pipeline for WSOD has three steps: feature extraction, proposal generation, and proposal classification. Note that in the context of object detection, a "proposal" refers to a candidate BBOX that covers the object in an image.

Based on this pipeline, different works have been introduced to improve model performance in each step. Bilen and Vedaldi (2016) introduced a phenomenal work, weakly supervised deep detection networks (WSDDN) that use a pretrained convolutional neural network (CNN) for WSOD. This model assumes that a pretrained CNN generates meaningful representations of the data that even contain location information of the entire and part of an object (B. Zhou et al. 2014). Based on this assumption, WSDDN is designed with a two-stream structure to explicitly reason about image regions. The limitation of this model, however, is its tendency to generate BBOX that contain only the part instead of the entire object. Tang et al. (2017) developed a new model named online instance classifier refinement (OICR) to leverage multistage learning to continue refining the BBOX to eliminate the issue of partial labeling caused by WSDDN. Each stage is a classifier trained by the outcome of the previous stage. As a result, the latter stages are trained not only with the most critical parts but also

the parts overlapping with them. In addition to the iterative refinement of BBOX, the initial selection of candidate proposals (BBOX) is important. Coupled multiple instance detection network (C-MIDN; Y. Gao et al. 2019) is a cutting-edge model that employs a parallel WSDDN structure and a segmentation map for selecting better proposals at the first stage of the optimization pipeline.

Almost all of these models use traditional proposal generation methods, such as selective search (Uijlings et al. 2013) or edge boxes (Zitnick and Dollár 2014) to create a large number of BBOX and then refine them until an optimal BBOX is found. There are much fewer works focusing on making improvements on the region proposal network, which aims to generate more accurate candidate BBOX in the first place (Hsu and Li 2020). High-quality proposals, however, have been proven to have a great influence on the performance of object detection tasks (Hosang et al. 2016). One pioneering research effort belonging to this second kind is the work by Tang et al. (2018), which proposed a multistage region proposal network where each stage filters out part of the original proposals based on different strategies. The model design is quite complex, however, involving the cascading of two WSOD models trained separately, which inevitably increases training time and is difficult to reproduce. Our work also focuses on making improvements on proposal generation. Unlike the work by Tang et al. (2018), our proposed deep neural network generates high-quality and precise proposals directly instead of a selection from randomly generated proposals.

Applications of Object Detection in Remote Sensing

Object detection can find a wide range of applications across urban and environmental science domains. Many recent studies work on tackling the challenges of object detection from remote sensing imagery; for example, W. Li and Hsu (2020) extended Faster R-CNN (Ren et al. 2015) to enable natural feature identification from remote sensing imagery. The authors evaluated performance of multiple deep CNN models and found that the very complex and deep CNN models do not always yield the best detection accuracy. Instead, the CNN model should be carefully designed according to the characteristics of training data and complexity of the objects and background scene. Other issues and studies like rotation-sensitive detection (Yu, Guan, and Ji 2015; Cheng, Zhou, and Han 2016; W. Li et al. 2017; Ding et al. 2018; Liao et al. 2018; Cheng et al. 2019; Z. Zhang et al. 2020), proposal quality (Long et al. 2017; Xu et al. 2017; Zhong, Han, and Zhang 2018), and real-time object detection (Liu and Mattyus 2015; Tang et al. 2017) are also developed.

In addition, weakly supervised learning and target detection from remote sensing imagery has been exploited by geospatial researchers. Han et al. (2015) used a multiscale sliding window to generate proposals from images and then leveraged a Bayesian classification network to classify proposals based on hand-crafted features, including low-level local features extracted from segmentation, statistics of these low-level features in an image patch (midlevel feature), and high-level features from a deep Boltzmann machine. D. Zhang et al. (2014) achieved weakly supervised learning by separating images into positive (with targets) and negative (without targets) samples without BBOX information. The positive images are initially obtained using saliency-based self-adaptive segmentation and then refined by negative mining to remove positive images without targets. This method is further improved by carefully selecting negative images that are informative and tend to be misclassified, diverse, and nonredundant (P. Zhou et al. 2016). A pretrained CNN is used to extract discriminative features. These works yield interesting results but are limited in single-class detection. Hence, their applicability in dealing with a large data set and complex object detection tasks still needs to be verified. In comparison, our work relies on a deep learning framework, and the entire learning process—from feature extraction, to proposal generation and classification—is all automatically done. In the next section, we present the formal problem statement.

Problem Formulation

In scenarios of strongly supervised object detection, the problem is to predict object-level labels, including object class ($O_{i,Class}$) and object bounding box (BBOX; $O_{i,BBOX}$), given the same information in the training data. In our study, we enabled a WSOD by replacing the BBOX information with simply a count of the objects in an image. Assuming

i ($i \in [0, m]$) is the index of an object in an image and j is the index of an image in the training samples, our WSOD problem can be formally represented as finding the mapping f such that

$$f :< Image_{j,\{Class\}},\ m > \rightarrow < O_{ij,Class},\ O_{ij,BBOX} >, \quad (1)$$

where m is the object count in an $Image_j$, $\{Class\}$ is all of the object classes in the same image, and O_{ij} is the index of an object in $Image_j$. As previous studies have shown, in the labeling process, the time for counting is substantially less than drawing the actual BBOX. Our research will therefore directly tackle the issue of high label cost in collecting training data, which is essential for supervised learning to achieve satisfying predictive performance. We accomplished this by explicitly incorporating spatial theory and principles in the model design, which is a brand new attempt in GeoAI and deep learning.

Spatial Explicitness in Deep Learning Models

A key issue that a WSOD network must solve is to identify the object location without the provision of this information in the training data. This location information can be represented as "critical points" that stay on or near an object such that a candidate BBOX can be drawn around these points for proposal generation. Here, we propose leveraging temporal data classification, such as with a recurrent neural network (RNN), to perform object detection in a weakly supervised manner. Current RNNs, such as long short-term memory (LSTM), have shown outstanding performance in classifying one-dimensional (1-D) sequence data, such as a speech segment, but it requires perframe labels in its learning process. To reduce the required labeling information, we further extended the LSTM by giving it a new objective function, namely connectionist temporal classification (CTC), which learns to identify the optimal segmentation location to separate different words in a speech sequence with only persequence labels, meaning using, for example, "Hello world" for a speech segment rather than labels for individual letters, "H-e-l-l-o-w-o-r-l-d," in each speech frame. Note that each data slice that forms a temporal or sequential piece of data could be letters in a text document, phonemes in a speech segment, or sequential image data (i.e., time series remote sensing data or a video clip). These data can all be fit into a temporal classification framework.

One key question in enabling the two-dimensional (2-D) WSOD using this 1-D temporal classification strategy is proper dimension reduction. Here, we argue that spatial theory and principles (i.e., spatial continuity and spatial autocorrelation) play a key role in the applicability of 1-D temporal classification for 2-D object detection. According to TFL (Tobler 1970), which states that nearby things are more related to each other, we can perform a serialization on the 2-D data by applying row-prime or column-prime scan orders. After serialization, spatial continuity in the main scanned direction will be retained. Although the spatial continuity perpendicular to this direction will be broken, because LSTM can capture contextual information in both immediate neighbors (short-term memory) and distant neighbors (long-term memory), this broken spatial dependency can still be "memorized" by the network.

Figure 1 illustrates the spatial theory–enabled WSOD. Given a feature map, we proposed to apply scan orders at four different directions (row order, reversed row order, column order, and reversed column order) to serialize the feature map into a 1-D feature sequence. The data were then sent to the temporal classification model to identify the optimal segmented locations for the objects. Our model differs from the traditional LSTM-CTC framework in two ways: First, a CTC requires labels of each word (both "hello" and "world") in a phrase ("hello world"). Our model instead takes the segmentation as a binary problem in which we treat objects of different types the same way; hence, there is only a need to provide a total object count (two for "hello world"). Our goal is to identify and separate the BBOX of all objects of interest (the foreground) from the background scene. Then, a CTC training process identifies the optimal mapping of predicted objects on the feature sequence to the object count information. Second, the objective of the CTC is to find where the words are located in each sentence and their order of occurrences; hence, the segmentation results based on probability estimation are always the location labels of each word. When applying this mechanism for segmenting 2-D objects with one spatial dimension broken, the segmented location will tend to locate on the most prominent spike (area with high visual attention) of the feature sequence. We call this location (or a series of such

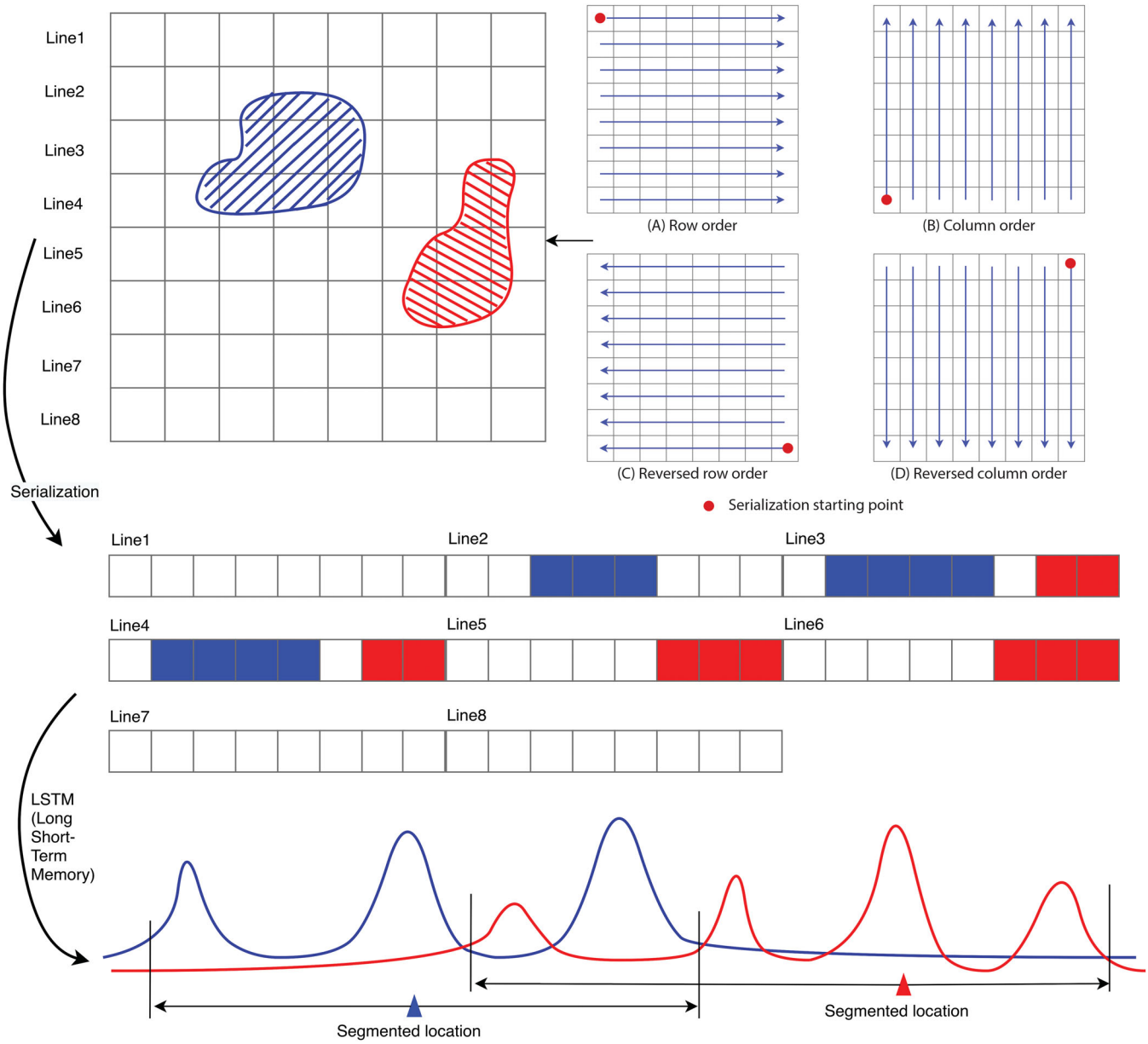

Figure 1. A visual example of a spatial theory– enabled strategy for weakly supervised object detection.

locations) the *critical point(s)* of an object, and it will be used as the center point for generating candidate proposals (BBOX) for object classifications in the next stage. In our proposed model, the feature maps are serialized in four different directions (as shown in Figure 1) because the object appearing in the feature maps might exert different temporal patterns. When training four temporal classification models (e.g., LSTM) with a shared CNN, behaviors of the LSTMs will be influenced by each other, and the four networks will tend to converge on the points falling on the same set of objects of interest, thereby generating more accurate predictions about object locations. Different from the commonly used selective search strategy, which needs to draw a great number of proposals (e.g., 2,000) to exhaust the potential locations and sizes of an object, our proposed approach can more intelligently locate the suspected area where an object will appear and therefore substantially reduce the number of proposals while increasing their quality. After the proposals are generated, they are sent to a region-based classifier for proposal ranking, refinement, and classification. Any region-based classifier under weak supervision could be integrated with this proposal generation network for object localization and prediction.

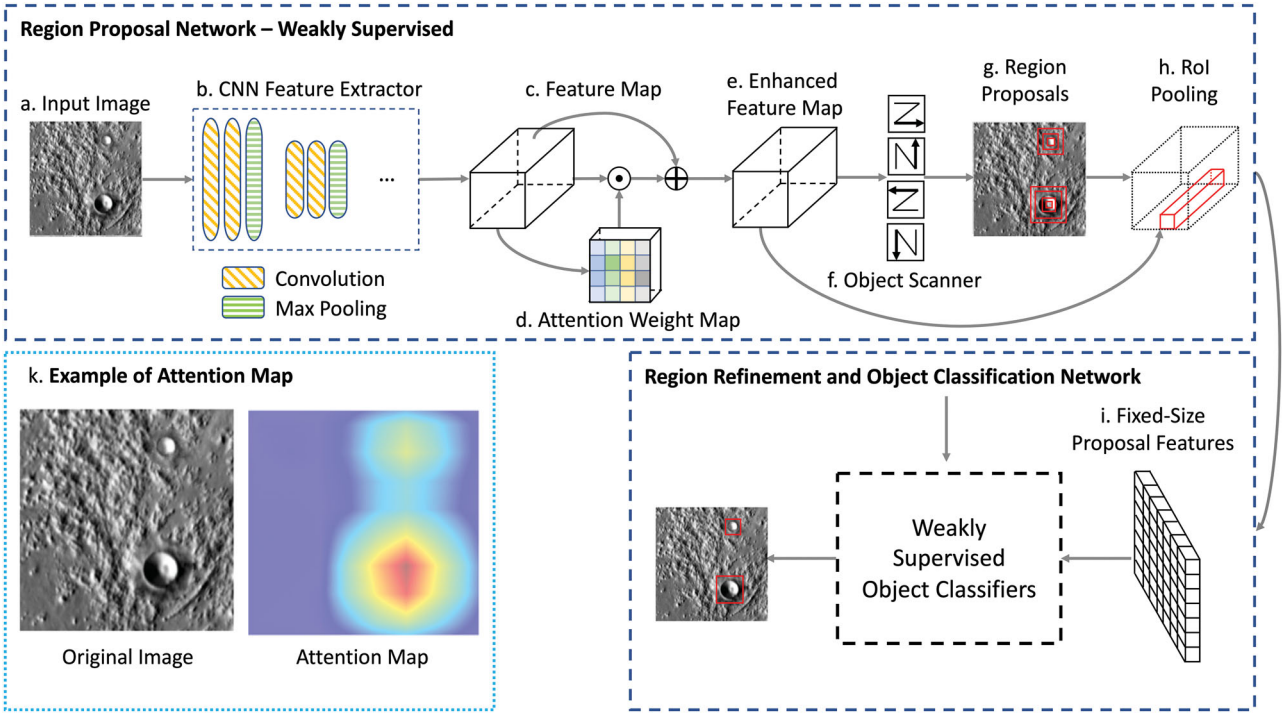

Figure 2. Proposed weakly supervised object detection pipeline. *Note:* CNN = convolutional neural network; RoI = region of interest.

Method

Weakly Supervised Object Detection Pipeline

Figure 2 illustrates the model structure and workflow of the proposed WSOD network. This network achieves active learning in two stages. The first stage is a region proposal network (RPN), which leverages the aforementioned temporal classification to identify candidate object proposals (upper pipeline). The second stage (lower pipeline) refines these proposals and conducts object classification. Instead of using strong instance-level supervision, both networks are supervised with weak labels, including a total object count and image-level annotations on object classes.

At the region proposal phase, the input image first goes through a CNN module (Figure 2B) for extracting the most prominent features that distinguish different images and objects. The resultant product is termed a feature map, $\mathbf{X} \in \mathbb{R}^{H \times W \times D}$ (H, W, and D are the height, weight, and depth of the feature map, respectively). Before sending the feature map $\mathbf{X}$ for subsequent processing, we augmented the feature map to highlight the most informative and relevant subregions. This was achieved by an attention mechanism that simulates how humans pay visual attention to different areas of an image once a prompt is given. Following this mechanism, a spatially normalized attention weight map $\mathbf{A} \in \mathbb{R}^{H \times W}$ is generated (Figure 2D). The values in $\mathbf{A}$ are normalized such that the sum of all values in $\mathbf{A}$ equals 1. Next, the original feature map $\mathbf{X}$ is multiplied (channel-wise) by the weight map $\mathbf{A}$ to create attention map $\mathbf{X}_a \in \mathbb{R}^{H \times W \times D}$. The enhanced feature map $\mathbf{X}'$ (Figure 2E) is created by adding $\mathbf{X}_a$ and $\mathbf{X}$ together.

Once the enhanced feature map is generated, it is sent to the core module of the RPN—the object scanner (Figure 2F), which contains four temporal classification modules running in parallel empowered by LSTM with CTC as the objective function. The four modules take serialized feature maps by the proposed scan orders as inputs (as shown in Figure 1) and output the segmented locations, which we call critical points, on or near the target objects. Then, multiple BBOX at different ratios and shapes are generated around the critical points to serve as the candidate region proposals (Figure 2G) for future refinement and object classification.

Next, the region of interest (RoI) pooling layer (Figure 2H) will extract and generate fixed-size feature maps of proposed regions (Figure 2I) from the enhanced feature map and send it to the object classifier (Figure 2, bottom pipeline). The common workflow of the object classifier is to make predictions on

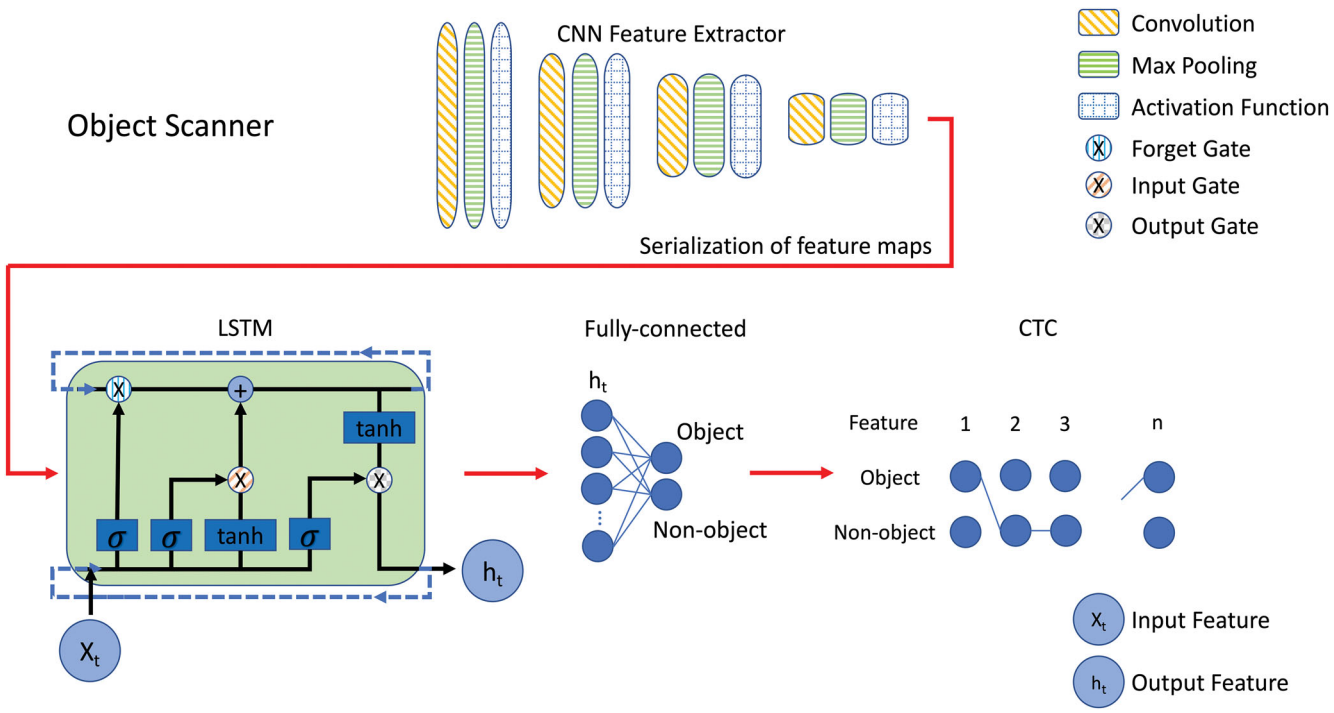

Figure 3. A spatially explicit design of the object scanner. *Note:* CNN = convoluted neural network; CTC = connectionist temporal classification; LSTM = long short-term memory.

the class of object within the candidate proposals. Because the deep network is weakly supervised, there is often a mechanism to rank and refine the proposals to increase the detection accuracy. Many deep learning models are designed to achieve this goal, such as the work by Bilen and Vedaldi (2016), Tang et al. (2017), and Y. Gao et al. (2019), among others. Almost all of these models target the improvement of the object classifier (Figure 2, bottom), however. Conversely, our model focuses on improving the RPN (Figure 2, top) by incorporating spatial theory and principles and then using existing models for object classification. It is worth noting that our framework is designed flexibly enough so it can be easily integrated into other models by replacing their RPN with our proposed network or by integrating the object classifiers of other models into our detection pipeline.

Taking the C-MIDN (Y. Gao et al. 2019) that is used as the object classifier in our task as an example (Figure 2, bottom), it partitions the object classification into two phases: initial detection and detection refinement. In initial detection, two proposal classifiers are applied; each contains two branches, with one responsible for detection and the other for classification. The first classifier will select top-scoring proposals as the detection result. These proposals might not always be of high quality, however (e.g., they might contain only partial objects). To address this issue, a segmentation map is integrated for removing poor-quality proposals, even if they are ranked highly but have little overlap with the detected area of interest (the foreground in the segmentation map). The second classifier will take the filtered proposals to perform classification with the aim of finding better results.

After the initial detection, the proposals and classification results are sent to a multistage refinement network that is also deep learning based. Each stage is trained under the supervision of instance labels obtained from the previous stage. To obtain instance labels for supervision, given an image with class label c, the proposal j with the highest score for class c will be used as the pseudo-ground-truth BBOX. In addition to labeling j, other proposals that have a high spatial overlap with j will be labeled as class c. Meanwhile, proposals that do not belong to class c will be labeled as background. Using this refinement strategy, the detected BBOX and object classification results can be further enhanced.

Region Proposal Network

As introduced, the RPN consists of four object scanners (Figure 3) integrated in parallel. Scanners

are responsible for locating the most discriminative part of objects and each consists of four core components: a joint CNN, an LSTM, a fully connected layer, and a CTC layer. For each image, the CNN extracts low-, mid-, and high-level features through consecutive convolution operations and generates so-called feature maps that contain latent features of candidate objects in the image. Next, serialization is applied on the feature maps, transferring 2-D maps into 1-D sequences $\mathbf{x}$, where $\mathbf{x} = (x_1, x_2, \ldots, x_T)$, representing the feature sequence after serialization. There are four different serialization orders (row-prime, column-prime, reversed row-prime, and reversed column-prime), and each scanner uses one of them. By this transformation, the 2-D spatial relations between objects are converted into four 1-D sequential relations. These 1-D sequences are fed into LSTMs, networks that persist historical information to leverage global spatial context for discriminative object detection. Different from classic object detection models, which are based on analysis of the 2-D images and their transformed feature maps after layers of convolution, our proposed deep learning model follows a very different path, which is to leverage temporal classification performed on 1-D sequential data for the object detection task.

The flow of information in LSTM is regulated by gates, which is a sigmoid operation followed by a pointwise multiplication operation. The gate control signals are a concatenation of current input and the output from the previous feature vector. There are three gates in each LSTM, an input gate, an output gate, and a forget gate. A forget gate controls the amount of information that will flow into the current state from the previous state, an input gate controls the percentage of the current input that will be added to the output of the previous state, and an output gate controls the output amount from the current state. These three gates make LSTM capable of learning long-term spatial dependencies that have been broken by the serialization and therefore can be leveraged to predict the objects' most discriminative parts.

More formally, the object recognition process can be described as follows: Given a feature sequence $\mathbf{x} = (x_1, x_2, \ldots, x_T)$, our target is to find a label sequence $\boldsymbol{L}^* = (l_1, l_2, \ldots, l_U)$ that annotates $\mathbf{x}$. Each l comes from a finite alphabet set (e.g., a to z for speech classification), and the length of the label sequence U will be equal to or shorter than the original sequence length T because the sequence might also contain nonclass elements. When using the 1-D sequence classification for 2-D WSOD, each x is an m-dimensional vector serialized from the 2-D feature map with m channels, and $l_i \in \{0, 1\}$ $(i \in [1, U])$ refers to the labels location for objects of interest. In our count-supervised learning context, all objects are considered as a single class (foreground), and the nonobject locations are annotated as 0, referring to the background. Our objective has then become to find the optimal placement of 1s in the feature sequence $\mathbf{x}$ such that $\sum_{i=1}^{U} l_i = n$, where n is the object count in an image.

More formally, let $\mathbf{y} \in (\mathbb{R}^2)^T$ be the output of the LSTM network, and $\mathbf{y}$ is denoted by y_t^k, where $k \in \{0, 1\}$. In addition, y_t^1 denotes the probability of y_t being a critical point on an object of interest, and y_t^0 means at time t, y is not activated and is therefore a background element. A typical CTC assumes that the network outputs $\boldsymbol{y}$ at different times t are conditionally independent; hence, the possibility of the original sequence $\mathbf{x}$ to contain a label sequence $\boldsymbol{\pi}$ can be expressed as

$$p(\boldsymbol{\pi}\boldsymbol{x}) = \prod_{t=1}^{T} y_t^k, \ \forall \boldsymbol{k} \in \{0, 1\} \text{ and } \boldsymbol{\pi} \in L^T, \tag{2}$$

where L^T is the set of all possible perframe label sequences that yield the final labels L^*. To avoid the occurrence of repeated labels for the same object, a many-to-one mapping of $\mathcal{B}$ from L^T to L^* is defined to remove the continuously repeated labels. Finally, $\mathcal{B}^{-1}$ is leveraged to identify the probability of a given label as the sum of all possible $\boldsymbol{\pi}$ that yields the same L^*:

$$p(L^*\boldsymbol{x}) = \sum_{\boldsymbol{\pi} \in \mathcal{B}^{-1}(L^*)} p(\boldsymbol{\pi}\boldsymbol{x}). \tag{3}$$

The objective function can therefore be written as

$$O_{\mathrm{RPN}} = -\ln(p(L^*|\mathbf{x})). \tag{4}$$

Different from the objective of the original CTC, instead of predicting the occurrence of different objects, our new model considers all objects to be the same type (foreground). Hence, each input element in the feature sequence will be predicted with the probability of being 1 (foreground) or 0 (background). The final prediction is to select $\boldsymbol{\pi}^*$, which maximizes the probability of the mapping from $\mathbf{x}$ to L^*, where

$$\boldsymbol{\pi}^* = arg\max_{\boldsymbol{\pi}} p(\boldsymbol{\pi}\boldsymbol{x}). \tag{5}$$

The units in $\boldsymbol{\pi}^*$ that are classified as 1 are the critical points predicted to be objects of interest. These

locations will serve as the center for generating multiple candidate proposals in different sizes, which are then used to perform proposal refinement and object classification.

Proposal Refinement and Object Classification Network

The object classifier is trained to sort proposals from the RPN into different classes. Because there are no object-level annotations, the candidate proposals generated are a good guess and will need to be further refined for more accurate localization. Here, we adopt the object classifier in the C-MIDN model to perform the proposal refinement and classification. There are three components in the classifier: proposal classification, initial selection, and optimization.

Proposal Classification. Given region proposals $\mathbf{R}$ in the images, an RoI pooling layer extracts the corresponding features and generates fixed-size proposal features. To generate image-level predictions, the proposals from each image are branched into two data streams, each generating a score matrix at the dimension of $|C| \times |R|$, where C is the set for classes and R is the set for proposals. These two score matrices later pass through a normalization process to generate two probability matrices, which will be continuously refined during the training phase to achieve different objectives. The one normalized along the row (dimension of object class) is a matrix that indicates the location probability distribution of a given class across all regions. Another matrix, normalized along the column (dimension of different proposals), stands for the probability distribution that a given region contains an object across all classes. The difference between these two matrices is that an object c could have a higher probability of being located in a certain region r than in other regions, but this region might contain an object other than c. For instance, a crater might be more likely to appear in a region where there are circular features, but these features might come from other classes, such as islands or lakes. Hence, a high score will be found at the class c row and the region r column in the former matrix but will have a low score in the same cell in the latter matrix.

The multiplication of corresponding cell values from two matrices can actually imply the true probability of a region containing a given class. A final matrix $\mathbf{p}_{cr}$ for generating image-level prediction is computed by element-wise multiplication of the two matrices. Then, we can obtain a prediction score for a given class c by the summation of the probability of this class over all proposals:

$$\varphi_c = \sum_{r=1}^{|\mathbf{R}|} \boldsymbol{p}_{cr}, \tag{6}$$

and because we have a prediction score for each class φ_c and image-level labels $\{\mathbf{y}_c \mid \mathbf{y}_c \in (0,1)\}$, the objective function is simply a multiclass cross-entropy function:

$$O_{classifier} = -\sum_{c=1}^{|\mathbf{C}|} \boldsymbol{y}_c \log \varphi_c. \tag{7}$$

Initial Selection. The proposal classification component gives each proposal a class label and its related score; however, the highest score is often localized at the most discriminative part of an object instead of the entire object. Two proposal classification components can be coupled. The top-scoring proposals of the first proposal classifier are removed from serving as the input of the second one, pushing the second classifier to search for other better proposals instead of localizing the same proposal again. Furthermore, in case the first classifier already locates the full-context proposal, a segmentation map of a given class in the image is employed to improve the robustness of the proposal removal process. If the overlap between the highlighted areas in the segmentation map and the top-scoring proposal is too small, it is more likely that there exists a better proposal. Suppose the set of pixels in the segmentation map for class c is M_c, and the set of pixels in the top-scoring proposal from the first proposal classification component is N_c, then the overlap r_c is defined as

$$r_c = \frac{|M_c \cap N_c|}{|M_c|}. \tag{8}$$

If r_c is smaller than a given threshold, the corresponding proposal will be removed from the input of the second proposal classifier; otherwise, it will be retained. In addition, the other candidate proposals that have an intersection over union (IoU) larger than a given threshold with the top-scoring proposals will also be removed. After the removal process, the two classifiers will be trained with the same objective function. The only difference is that the

input proposals are different. The final predicted proposal will be determined through a selection process of the outputs of the two classifiers. If the top-ranking proposals have little overlap, both proposals will be retained. If the top-ranking proposals have a high overlap (over 50 percent IoU), the proposal from the second classifier will be selected because the proposal tends to contain the entire object rather than its part.

Optimization. After the initial detection, the proposals and classification results are sent to a refinement network (Figure 2I) that is also deep learning based for further refinement of the detection result. Each stage is trained under the supervision of instance labels obtained from the previous stage. At iteration k $(k > 0)$, the instance classifier generates a proposal score matrix $\mathbf{p} \in \mathbb{R}^{(|C|+1)\times|R|}$, where C is the set for classes and R is the set for proposals. The extra row in the class dimension refers to the nonobject background. To obtain instance labels for supervision, given an image with class label c, the proposal j with the highest score for class c will be used as the pseudo-ground-truth BBOX. In addition to labeling j, other proposals that have a high spatial overlap with j will also be labeled as class c. Meanwhile, proposals that do not belong to class c will be labeled as background. Mathematically, the objective of this refinement process can be expressed as

$$O_{refinement} == -\frac{1}{|R|}\sum_{r=1}^{|R|}\sum_{c=1}^{C+1} y_r \log p_{c,r}, \qquad (9)$$

where c is the class index, $c \in C$, r is the proposal index, $c \in R$, y_r is the class label for proposal r, and $p_{c,r}$ is the probability for a proposal r to contain an object of class c. Using this refinement strategy, the detected BBOX and object classification results can be further enhanced.

The overall objective function of this WSOD model (O_{WSOD}) has become the addition of objectives in the RPN phase (O_{RPN} in Equation 4), the object classification phase ($O_{classifier}$ in Equation 7), and the refinement phase ($O_{refinement}$ in Equation 9). Namely,

$$O_{WSOD} = O_{RPN} + O_{classifier} + O_{refinement} \qquad (10)$$

The solution of this problem will move toward minimizing O_{WSOD}. Hence, O_{WSOD} can also be used as the loss function.

Results and Discussion

Data Preparation: Mars Crater Data Set

To assess the performance of our proposed model, we created a Mars crater data set that contains 10,000 image scenes of impact craters. Detecting a Mars crater is of significant scientific value to enhance the understanding of the geomorphological process of the Mars surface, facilitating various Mars exploration missions for identifying extraterrestrial life and water on the red planet. We leveraged two resources to create the training data set: (1) the global database of Mars impact craters $\geq$1 km (Robbins and Hynek 2012), containing 384,343 craters, along with their center location and diameter sizes, and (2) the Mars Odyssey Thermal Emission Imaging System–Infrared daytime global mosaic, which provides the imagery of Mars's surface at 100 m resolution (Edwards et al. 2011).

To create the training images, we randomly selected 10,000 locations and clipped the same number of image scenes at a size of 256×256 pixels^2 (covering an area of 655.36 km^2). According to the location and extent of each image scene, a search through the Mars crater database was conducted to find the craters within each scene. Note that a spatial index is built on top of the database to allow for fast spatial query of the crater data. Image scenes containing partial craters were discarded. For the craters within each scene, the count label—number of craters within an image—was generated for use as the input of our WSOD model. Strong labels (object class and BBOX) were also created for evaluating the predictive performance of our proposed model. In this way, we were able to automatically generate an authentic data set to guide the machine learning process.

The experiments were conducted on Amazon Elastic Compute Cloud. The g3x.xlarge instance with NVIDIA Tesla M60 GPU, which has an 8 GB memory, was used to run the experiments.

Detection Results

Figure 4 demonstrates some sample detection results. Overall, the network worked pretty well under weakly supervised learning. Using the Mars crater data set, our detector achieved 85 percent detection accuracy, as measured by the mean average precision (mAP). The result displayed in Figure 4A

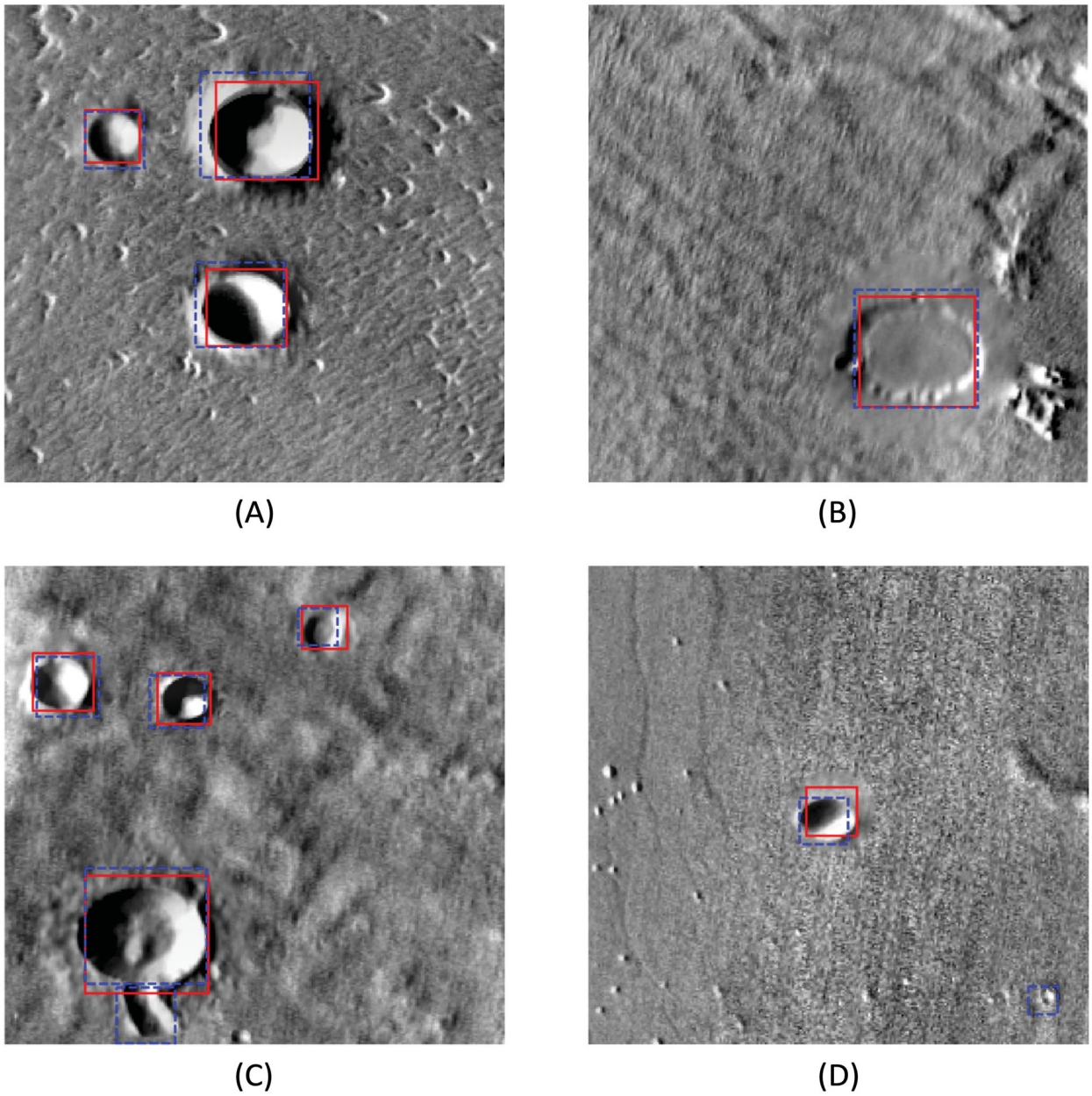

Figure 4. Example detection results. The solid red line is the ground-truth BBOX, and the dashed blue line is the BBOX predicted by our proposed model. BBOX = bounding box.

shows that craters of different sizes can all be detected correctly from the image. Even without the provision of the ground truth BBOX (red), the predicted BBOX is still highly accurate and has significant overlap with the ground truth. In Figure 4B, even the pattern is not obvious, and the difference between the object and the background is small, but the network still does well with detection. Furthermore, the network detects craters that are not in the benchmark data set. For instance, in Figure 4C, even though the "crater" at the bottom of the images does not exist in the Mars crater database, our model is still able to detect it. With this strong detection capability, scientists can further clarify the relations between this object and another labeled crater it connects with; for instance, whether they are both impact craters and belong to the same crater or they should be labeled as two different craters. Another example is shown in Figure 4D. In the original data source, craters with diameters less than 1 km are not labeled. There are a vast number of such craters on the Mars surface, however; relying

Table 1. Comparison of detection performance among different models

WSOD models	Predictive performance measured by mAP (%)	Run time (images/second)
WSDDN	62.7	2.89
OICR	68.2	3.58
C-MIDN	75.8	5.41
Our proposed model	84.8	4.83

Note: A nearly 10% performance (mAP) increase was achieved with our model, as compared to the cutting-edge C-MIDN model. Our model is also more computationally efficient than C-MIDN. WSOD = weakly supervised object detection; mAP = mean average precision; WSDDN = weakly supervised deep detection network; OICR = online instance classifier refinement; C-MIDN = coupled multiple instance detection network.

on manual labeling of these smaller craters would not be feasible because the task would be too labor intensive and could take years or even decades to finish. In Figure 4D, such a small crater appearing near the bottom right of the image is successfully detected by our proposed model. This detection result shows the generalizability of our approach.

Performance Comparison with Cutting-Edge Weakly Supervised Object Detection Models

In this experiment, our proposed network was compared with cutting-edge WSOD models, using the same crater detection task to evaluate their effectiveness. The models compared with our approach included (1) a WSDDN (Bilen and Vedaldi 2016), one of the phenomenal works that addresses the WSOD problem using CNN; (2) an OICR (Tang et al. 2017), a popular model that improves WSDDN through multistage classification refinement; and (3) a C-MIDN (Y. Gao et al. 2019), a WSOD model with cutting-edge performance achieved by using a double WSDDN and a segmentation map for better proposal selection. Table 1 demonstrates both the prediction accuracy and run time for each model. It can be seen that our proposed model yielded the overall best performance in terms of detection accuracy (mAP) among all models. This is due to its smart spatial principle enhanced strategy for generating proposals on or near the objects of interest instead of exhausting possible locations using traditional approaches. In addition, the use of the attention map could further guide our model to better detect these critical points, yielding better detection accuracy than both the classic and cutting-edge models. Regarding model run time at the prediction phase, as the model is improved by adding more components to achieve better performance, their run time also increases. For instance, OICR is an improvement made on top of WSDDN by adding an iterative optimization procedure, and C-MIDN advances OICR by adding a segmentation map for better object localization and classification. Compared to these cutting-edge models, especially C-MIDN, our proposed model yields the best predictive accuracy and reasonable run time.

Effect of Optimized Training Strategies

It is known that for deep neural networks, training strategies are highly important for achieving a satisfying result. There is no universal guideline, however, about how to train a network, and the training strategy often varies according to different tasks. In our crater detection task, instead of tuning multiple hyperparameters with multiple models, we adopted another strategy using an experimental search to train and fine-tune the model. During training, we monitored the model's performance, and each time the accuracy curve became flat (meaning that the model was stuck in a local optimum) we saved and retrained the model with different hyperparameters from the point where the phenomenon occurred. The improvement introduced by our proposed strategy is illustrated in Figure 5. Dashed vertical lines in Figure 5A indicate the points at which we stopped the training and retrained the model with different hyperparameters. At the first stop, we changed the learning rate. When training a deep neural network, the learning rate is often the most important factor in deciding a model's performance. A small learning rate can slow down the learning process and make the model very difficult to converge. A large value, in comparison, might help the model to quickly converge but at a suboptimal solution. Hence, we first set a large value to train the model (0.001), and when the accuracy curve became

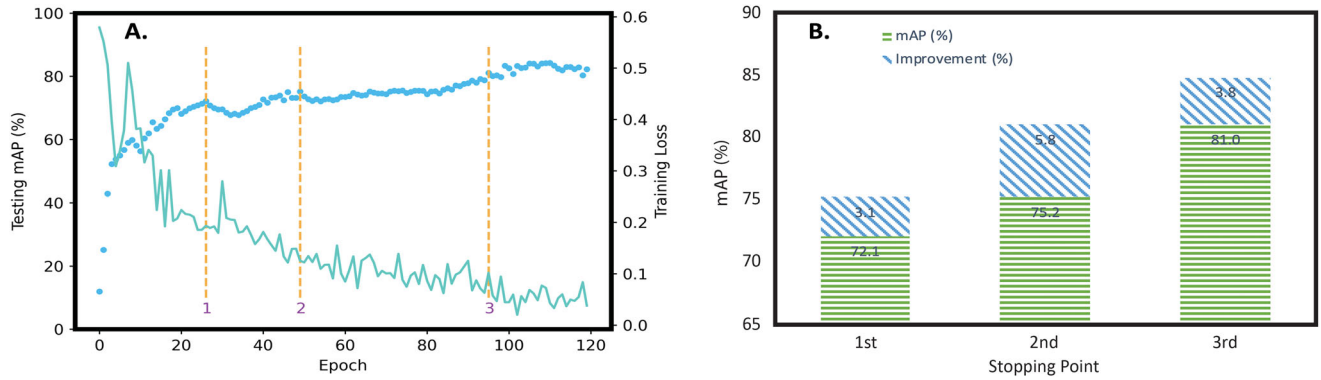

Figure 5. Performance improvement by different training optimization strategies. (A) Training loss (solid line that shows a declining trend) and the predictive accuracy measured by mAP (discrete points showing an increasing trend). (B) Quantitative improvement by proposed optimization strategies: dynamic learning rate (first), batch normalization (second), and reduced candidate proposals (third). *Note:* mAP = mean average precision.

flat we modified it with a smaller value (0.0005) to increase the search space toward a better solution. A 3.1 percent mAP increase was found by changing the learning rate (Figure 5B).

Next, when the curve became flat again, we integrated batch normalization, a technique to renormalize the data during training to further boost performance. To achieve this, we added a batch normalization layer to renormalize the data before they were used by our RPN and the object classifier. Adding a normalization layer means that the layers after it need to be retrained. It can be seen in Figure 5A that after the second stop it took a relatively long training period before the accuracy curve went up again. This strategy yields a 5.8 percent increase in mAP. Finally, at the last stop we reduced the total number of proposals. A regular RPN generates proposals (candidate BBOX) with several ratios. The classifier will then choose the best fitted proposal containing the target object. When the number of total proposals increases, the difficulty of training a classifier increases. Because the Mars crater data set only contains nearly round objects, we removed other ratios to reduce the training efforts of the classifier. As Figure 5B demonstrates, this modification improved the accuracy by an additional 3.8 percent. Figure 5B also quantifies the accuracy increase introduced by each optimization strategy. Through the experimental search, we were able to identify the proper training strategies for this task. Furthermore, additional experiments were conducted, and we found that these strategies can also be applied in combination at the beginning of the training to further reduce training time and achieve a satisfying performance.

Effects of Size and Diversity of the Training Data Set

In this experiment, we tested the impact of the number of training images on model performance. Although our approach can, in theory, generate any number of training images, how many would be sufficient and how many would be overabundant? We argue here that the right number of images needed is the set that can fully represent the data distribution in the original data source. More images than this number will not contribute further to the performance but will instead increase training time. To answer this question, we trained our model with different sets of training data randomly selected from the original training data set containing 9,000 image scenes, and all of these models used the same extra 1,000 images for testing. The result shown in Figure 6 provides the comparative results. It can be seen that the overall model performance (prediction accuracy) became stable after the number of training images reached $N = 3{,}000$ and more. To further illustrate the relationship between the model performance and the number of training images, we extracted feature maps of the testing images from a different training set and performed principal component analysis to obtain the prominent features in an abstract space and then projected the first two components onto a 2-D plane (Figures 6C–6I). We use colors to represent the images containing different numbers of craters (zero to six) such that the figures can show not only the overall distribution of the feature components but also the distribution of images separated by the number of craters they contain.

These figures demonstrate a better cluster separation and a similar feature distribution when training

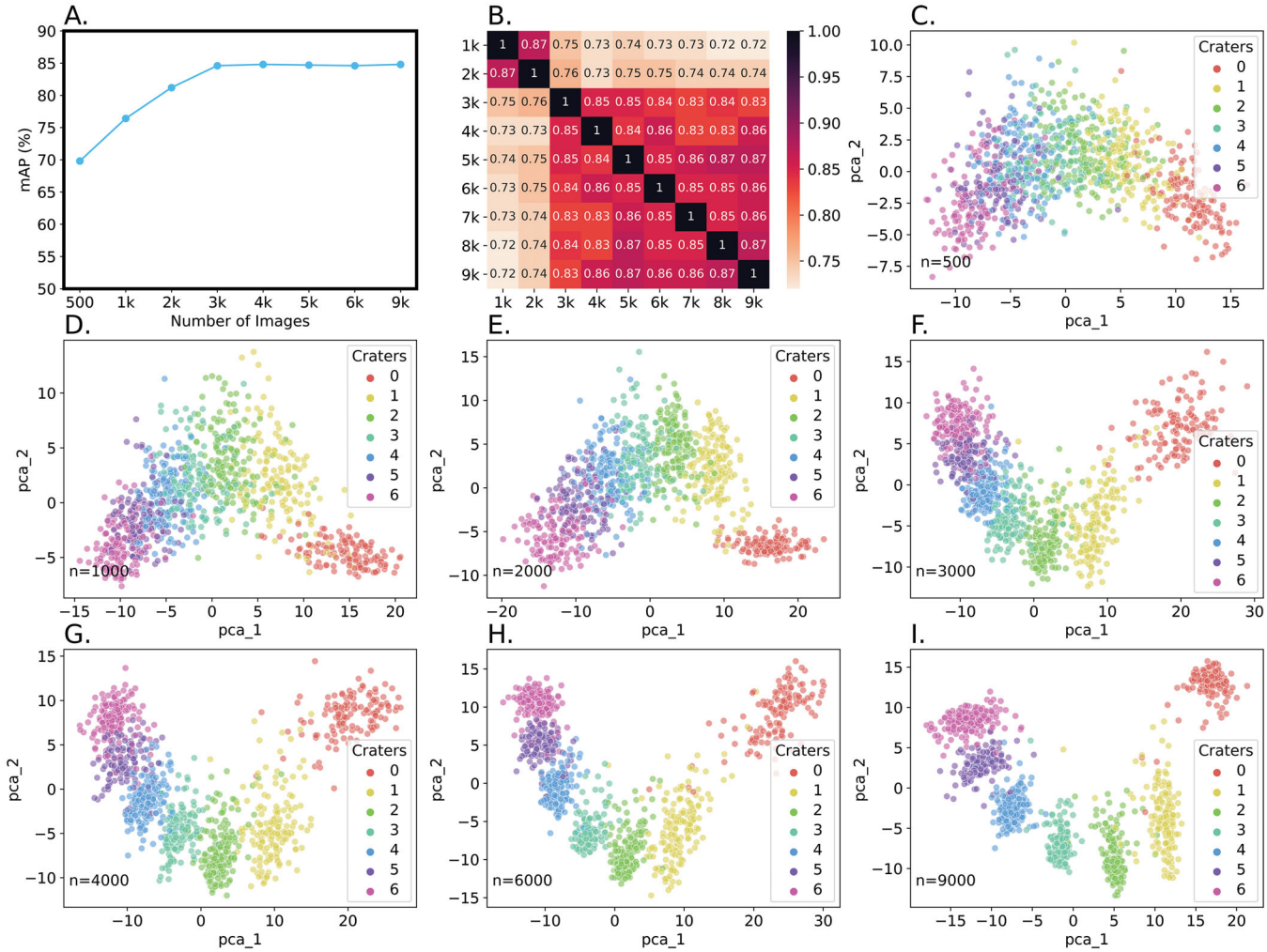

Figure 6. Model performance as an effect of the size of the training data set. *Note:* mAP = mean average precision; pca = principal component analysis.

sets are 3,000 items and above. In particular, the better separated clusters indicate that the model can better identify different objects and generate discriminating features that are important for the RPN to achieve object localization. We also calculated the cosine similarity of image features generated by the models trained with these sets. The similarity matrix between different input sets ($N = 1{,}000$–$9{,}000$) shown in Figure 6B quantitatively verifies the observation. This result provides guidance and confidence in selecting the set with 3,000 images to train the model for equally good performance and better training efficiency.

Model Generalizability in Object Detection of Natural Features on Earth and Other Planets

To verify the generalizability of our proposed WSOD model, we performed additional experiments on a natural feature data set composed of four terrain categories (crater, hill, volcano, and sand dunes). Whereas most of these features are natural features on Earth, we also added some images from Mars (i.e., craters) to diversify the data sets. The training data contain ~120 remote sensing images per terrain category. Table 2 provides the prediction accuracy on this new data set using different WSOD models. Among all comparable models, ours performs the best, with the highest AP achieved for craters, volcanoes, and sand dunes. It also beats other models in the overall mAP. This is attributed to the novel strategy we developed to enable more accurate object localization without this information being explicitly provided in the training data.

Figure 7 demonstrates the prediction results of each feature category (in dashed blue box) and the ground-truth labels (in solid red box). Note that the

ground-truth labels are only used for results evaluation and not for model training in the weak supervision context. Almost perfect object extents are predicted using our proposed approach. The model is also capable of predicting multiple instances in the same image scene (Figure 7B). For crater detection, it can be seen that not only simple craters are detected but nested ones and small craters that do not exist in the original database can also be detected (Figure 7D). This result clearly demonstrates the effectiveness of the proposed approach in detecting natural features.

Table 2. Comparison of prediction accuracy in terms of average precision for individual terrain category and mAP across all categories

Models	Crater	Hill	Volcano	Dunes	mAP
WSDDN	0.68	0.59	0.66	0.45	0.60
OICR	0.79	0.77	0.72	0.67	0.74
C-MIDN	0.85	**0.79**	0.79	0.71	0.79
Ours	**0.88**	0.77	**0.81**	**0.73**	0.80

Notes: mAP = mean average precision; WSDDN = weakly supervised deep detection network; OICR = online instance classifier refinement; C-MIDN = coupled multiple instance detection network. Bold text refers to the highest mAP in each column. For instance, i.e., 0.88 is the highest mAP obtained for detecting craters among all methods.

Summary and Outlook

This article reports a brand new attempt to develop a WSOD framework using in-depth integration of cutting-edge deep learning models and spatial theory and principles. It is known that natural feature detection has suffered tremendous challenges due to the lack of proper training data, the various forms of objects of the same class, and their vague boundaries, as compared to man-made features. This article tackles the problem with an innovative solution that converts the 2-D object detection problem into a 1-D temporal classification to use the latter's advanced search optimization technique, which allows for the use of weak labels for the detection task. What enables such a conversion is TFL, which states that nearby things are more similar than distant things. Using the scan order along the x or y direction, the spatial continuity along one direction is retained, and its continuity along the perpendicular direction will still be captured by the long- and

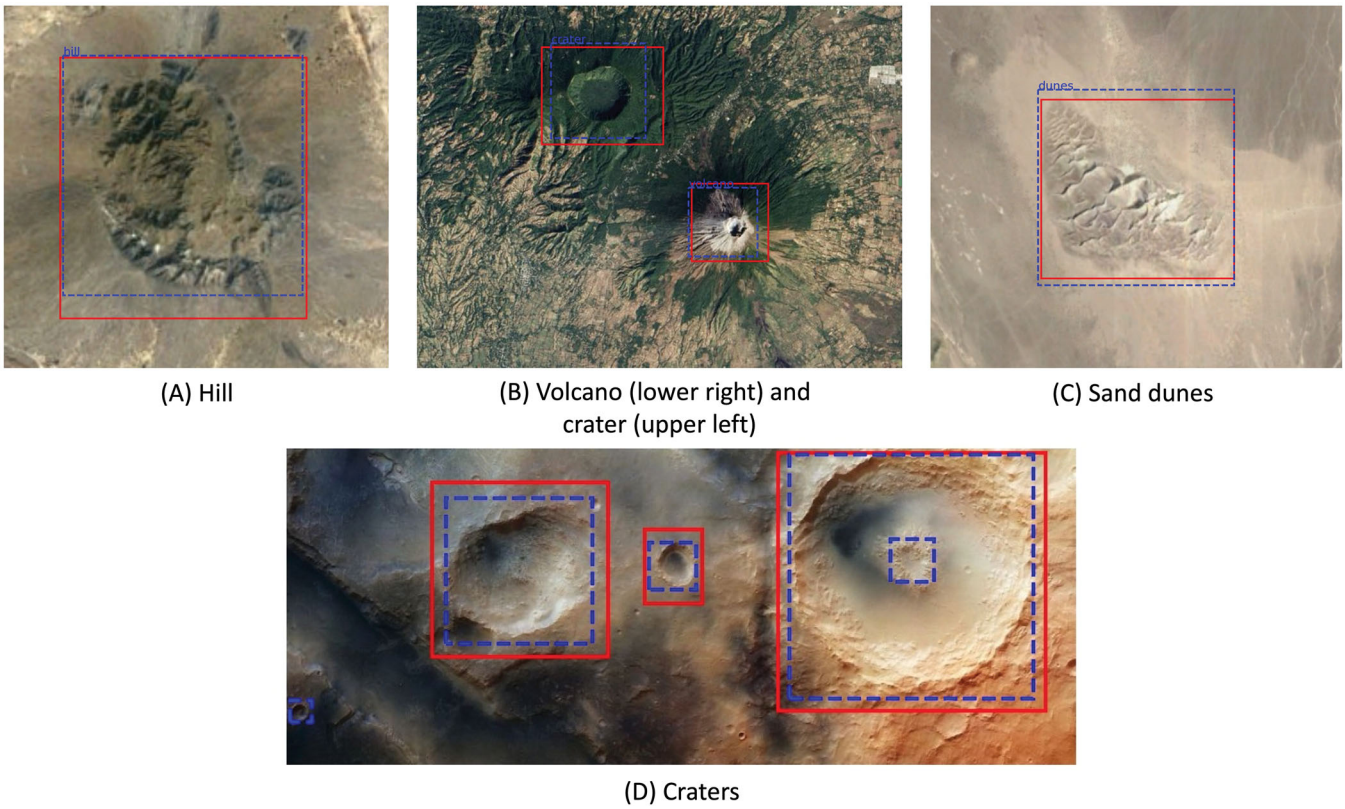

Figure 7. Prediction results for natural terrain features. The solid red line is the ground-truth BBOX, and the dashed blue line is the BBOX predicted by our proposed model. BBOX = bounding box.

short-term memory of the RNN (i.e., LSTM). Experiments were conducted to prove that our proposed model achieved state-of-the-art performance. In the future, we will further refine our model and leverage its capabilities to enrich the existing Mars crater databases. As the most comprehensive database containing more than 380,000 impact craters on Mars, the one developed by Robbins and Hynek (2012) was the result of multiyear research containing tremendous manual work and labeling efforts. A large number of small craters with diameters less than 1 km have not yet been included in the database, however. Our work in AI and especially GeoAI will play a key role in automating the detection of Mars craters, as well as many other natural features on Earth's surface, especially with the lack of proper training data. This research will no doubt advance the science for better exploration of Earth and space.

Acknowledgments

The authors sincerely thank Editor Dr. Ling Bian and the anonymous reviewers for their valuable comments.

Funding

This work is in part supported by the National Science Foundation under Grants BCS-1853864, BCS-1455349, OIA-2033521, OIA-1936677, and OIA- 1937908.

ORCID

Wenwen Li http://orcid.org/0000-0003-2237-9499

References

Anselin, L. 1995. Local indicators of spatial association—LISA. *Geographical Analysis* 27 (2): 93–115.

Arundel, S. T., W. Li, and X. Zhou. 2018. The effect of resolution on terrain feature extraction. *PeerJ Preprints* 6:e27072v1. doi: 10.7287/peerj.preprints.27072v1.

Barrett, B., and G. P. Petropoulos. 2013. Satellite remote sensing of surface soil moisture. In *Remote sensing of energy fluxes and soil moisture content*, ed. G. P. Petropoulos, 85–111. Boca Raton, FL: CRC Press.

Bejiga, M. B., A. Zeggada, A. Nouffidj, and F. Melgani. 2017. A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery. *Remote Sensing* 9 (2):100. doi: 10.3390/rs9020100.

Bilen, H., and A. Vedaldi. 2016. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, ed. L. Agapito, T. Berg, J. Kosecka, and L. Zelnik-Manor, 2846–54. Las Vegas: IEEE.

Blaschke, T. 2010. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing* 65 (1):2–16. doi: 10.1016/j.isprsjprs.2009.06.004.

Cheng, G., J. Han, and X. Lu. 2017. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE* 105 (10):1865–83. doi: 10.1109/JPROC.2017.2675998.

Cheng, G., J. Han, P. Zhou, and D. Xu. 2019. Learning rotation-invariant and Fisher discriminative convolutional neural networks for object detection. *IEEE Transactions on Image Processing* 28 (1):265–78. doi: 10.1109/TIP.2018.2867198.

Cheng, G., P. Zhou, and J. Han. 2016. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing* 54 (12):7405–15. doi: 10.1109/TGRS.2016.2601622.

Ding, J., N. Xue, Y. Long, G.-S. Xia, and Q. Lu. 2018 Learning ROI transformer for detecting oriented objects in aerial images. *arXiv*:1812.00155.

Edwards, C. S., K. J. Nowicki P. R. Christensen, J. Hill, N. Gorelick, and K. Murray. 2011. Mosaicking of global planetary image datasets: 1. Techniques and data processing for Thermal Emission Imaging System (THEMIS) multi-spectral data. *Journal of Geophysical Research: Planets* 116(E10). doi: 10.1029/2010JE003755.

Fotheringham, A. S., C. Brunsdon, and M. Charlton. 2003. *Geographically weighted regression: The analysis of spatially varying relationships*. Chichester, UK: Wiley.

Gahegan, M. 2020. Fourth paradigm GIScience? Prospects for automated discovery and explanation from data. *International Journal of Geographical Information Science* 34 (1):1–21. doi: 10.1080/13658816.2019.1652304.

Gao, M., A. Li, R. Yu, V. I. Morariu, and L. S. Davis. 2018. C-WSL: Count-guided weakly supervised localization. In Proceedings of the European conference on computer vision (ECCV), 152–68. Munich, Germany, September 8–14.

Gao, Y., et al. 2019. C-MIDN: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 9834–43. Seoul, Korea, October 27–November 2, 2019.

Goodchild, M. F. 2004. The validity and usefulness of laws in geographic information science and geography. *Annals of the Association of American Geographers* 94 (2): 300–303.

Goodchild, M. F., and L. L. Hill. 2008. Introduction to digital gazetteer research. *International Journal of*

Geographical Information Science 22 (10):1039–44. doi: 10.1080/13658810701850497.

Han, J., D. Zhang, G. Cheng, L. Guo, and J. Ren. 2015. Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning. *IEEE Transactions on Geoscience and Remote Sensing* 53 (6):3325–37. doi: 10.1109/TGRS.2014.2374218.

Hill, L. L., J. Frew, and Q. Zheng. 1999. Geographic names: The implementation of a gazetteer in a georeferenced digital library. *D-Lib* 5 (1). doi: 10.1045/january99-hill.

Hosang, J., R. Benenson, P. Dollár, and B. Schiele. 2016. What makes for effective detection proposals? *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (4):814–30. doi: 10.1109/TPAMI.2015.2465908.

Hsu, C. Y., and W. Li. 2020. Learning from counting: Leveraging temporal classification for weakly supervised object localization and detection. The 31st British Machine Vision Virtual Conference, September 7-10, 2020 (virtual), Paper ID: 0621. https://www.bmvc2020-conference.com/assets/papers/0621.pdf

Jasiewicz, J., and T. F. Stepinski. 2013. Geomorphons—A pattern recognition approach to classification and mapping of landforms. *Geomorphology* 182:147–56. doi: 10.1016/j.geomorph.2012.11.005.

Kamusoko, C. 2017. Importance of remote sensing and land change modeling for urbanization studies. In *Urban development in Asia and Africa*, ed. Y. Murayama, C. Kamusoko, A. Yamashita, R. C. Estoque, 3–10. Singapore: Springer.

Li, H., X. Dou, C. Tao, Z. Hou, J. Chen, J. Peng, M. Deng, and L. Zhao. 2017. RSI-CB: A large scale remote sensing image classification benchmark via crowdsource data. *arXiv*:1705.10450.

Li, W. 2020. GeoAI: Where machine learning and big data converge in GIScience. *Journal of Spatial Information Science* 20 (20):71–77. doi: 10.5311/JOSIS.2020.20.658.

Li, W., and C.-Y. Hsu. 2020. Automated terrain feature identification from remote sensing imagery: A deep learning approach. *International Journal of Geographical Information Science* 34 (4):637–60. doi: 10.1080/13658816.2018.1542697.

Li, W., B. Zhou, C. Y. Hsu, Y. Li, and F. Ren. 2017. Recognizing terrain features on terrestrial surface using a deep learning model. An example with crater detection. In *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*, ed. H. Mao, Y. Hu, and B. Kar, 33–36. Los Angeles: ACM.

Liao, M., Z. Zhu, B. Shi, G.-S. Xia, and X. Bai. 2018. Rotation-sensitive regression for oriented scene text detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, 5909–18. Salt Lake City, UT, June 18–22.

Lindsay, J. B., and K. Dhun. 2015. Modelling surface drainage patterns in altered landscapes using LiDAR. *International Journal of Geographical Information Science* 29 (3):397–411. doi: 10.1080/13658816.2014.975715.

Liu, K., and G. Mattyus. 2015. Fast multiclass vehicle detection on aerial images. *IEEE Geoscience and Remote Sensing Letters* 12:1938–42. doi: 10.1109/LGRS.2015.2439517.

Long, Y., Y. Gong, Z. Xiao, and Q. Liu. 2017. Accurate object localization in remote sensing images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing* 55 (5):2486–98. doi: 10.1109/TGRS.2016.2645610.

Micheletti, N., L. Foresti, S. Robert, M. Leuenberger, A. Pedrazzini, M. Jaboyedoff, and M. Kanevski. 2014. Machine learning feature selection methods for landslide susceptibility mapping. *Mathematical Geosciences* 46 (1):33–57. doi: 10.1007/s11004-013-9511-0.

Ren, S., K. He, R. Girshick, and J. Sun. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (6):1137–49.

Robbins, S. J., and B. M. Hynek. 2012. A new global database of Mars impact craters $\geq$1 km: 1. Database creation, properties, and parameters. *Journal of Geophysical Research: Planets* 117 (E5). doi: 10.1029/2011JE003966.

Tang, P., X. Wang, X. Bai, and W. Liu. 2017. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, ed. R. Chellappa, Z. Zhang, and A. Hoogs, 2843–51. Honolulu, HI: IEEE.

Tang, P., X. Wang, A. Wang, Y. Yan, W. Liu, J. Huang, and A. Yuille. 2018. Weakly supervised region proposal network and object detection. In *Proceedings of the European Conference on Computer Vision (ECCV2018)*, ed. V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, 352–68. Cham, Switzerland: Springer.

Tobler, W. R. 1970. A computer movie simulating urban growth in the Detroit region. *Economic Geography* 46:234–40. doi: 10.2307/143141.

Tomaszewski, B., S. Tibbets, Y. Hamad, and N. Al-Najdawi. 2016. Infrastructure evolution analysis via remote sensing in an urban refugee camp—Evidence from Za'atari. *Procedia Engineering* 159:118–23. doi: 10.1016/j.proeng.2016.08.134.

Uijlings, J. R., K. E. Van De Sande, T. Gevers, and A. W. Smeulders. 2013. Selective search for object recognition. *International Journal of Computer Vision* 104 (2):154–71. doi: 10.1007/s11263-013-0620-5.

Xia, G.-S., J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu. 2017. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing* 55 (7):3965–81. doi: 10.1109/TGRS.2017.2685945.

Xu, Z., X. Xu, L. Wang, R. Yang, and F. Pu. 2017. Deformable convnet with aspect ratio constrained nms for object detection in remote sensing imagery. *Remote Sensing* 9 (12):1312. doi: 10.3390/rs9121312.

Yang, Y., and S. Newsam. 2010. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information*

Systems, ed. D. Agrawal and P. Zhang, 270–79. New York: ACM. doi: 10.1145/1869790.1869829.

Yu, Y., H. Guan, and Z. Ji. 2015. Rotation-invariant object detection in high-resolution satellite imagery using superpixel-based deep Hough forests. *IEEE Geoscience and Remote Sensing Letters* 12 (11):2183–87. doi: 10.1109/LGRS.2015.2432135.

Zhang, D., J. Han, G. Cheng, Z. Liu, S. Bu, and L. Guo. 2014. Weakly supervised learning for target detection in remote sensing images. *IEEE Geoscience and Remote Sensing Letters* 12:701–5.

Zhang, Z., R. Jiang, S. Mei, S. Zhang, and Y. Zhang. 2020. Rotation-invariant feature learning for object detection in VHR optical remote sensing images by double-net. *IEEE Access* 8:20818–27. doi: 10.1109/ACCESS.2019.2960931.

Zhong, Y., X. Han, and L. Zhang. 2018. Multi-class geospatial object detection based on a position-sensitive balancing framework for high spatial resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing* 138:281–94. doi: 10.1016/j.isprsjprs.2018.02.014.

Zhou, B., A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. 2014. Object detectors emerge in deep scene CNNs. *arXiv:*1412.6856.

Zhou, P., G. Cheng, Z. Liu, S. Bu, and X. Hu. 2016. Weakly supervised target detection in remote sensing images based on transferred deep features and negative bootstrapping. *Multidimensional Systems and Signal Processing* 27 (4):925–44. doi: 10.1007/s11045-015-0370-3.

Zhou, W., S. Newsam, C. Li, and Z. Shao. 2018. PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS Journal of Photogrammetry and Remote Sensing* 145:197–209. doi: 10.1016/j.isprsjprs.2018.01.004.

Zhou, X., W. Li, and S. T. Arundel. 2019. A spatio-contextual probabilistic model for extracting linear features in hilly terrains from high-resolution DEM data. *International Journal of Geographical Information Science* 33 (4):666–86. doi: 10.1080/13658816.2018.1554814.

Zhu, D., F. Zhang, S. Wang, Y. Wang, X. Cheng, Z. Huang, and Y. Liu. 2020. Understanding place characteristics in geographic contexts through graph convolutional neural networks. *Annals of the American Association of Geographers* 110 (2):408–20. doi: 10.1080/24694452.2019.1694403.

Zitnick, C. L., and P. Dollár. 2014. Edge boxes: Locating object proposals from edges. In *Proceedings of the European Conference on Computer Vision*, ed. D. Fleet, T. Paidla, B Schiele, and T. Tuylelaars, 391–405. Springer.

WENWEN LI is an Associate Professor in GIScience in the School of Geographical Sciences and Urban Planning, Arizona State University, Tempe, AZ 85287-5302. E-mail: wenwen@asu.edu. She has directed the Cyberinfrastructure and Computational Intelligence lab since 2012 at ASU. Her research interests include cyberinfrastructure, geospatial big data, machine learning, and their applications in data-intensive environmental and social sciences.

CHIA-YU HSU is an Associate Scientific Software Engineer in the School of Geographical Sciences and Urban Planning, Arizona State University, Tempe, AZ 85287-5302. E-mail: chsu53@asu.edu. His research interests are computer vision, deep learning, and object detection.

MAOSHENG HU is a Lecturer in the School of Geography and Information Engineering, China University of Geosciences, Wuhan, China. E-mail: humsh@cug.edu.cn. His research interests include multiscale spatial data organization, modeling, and space–time data mining.