

Shunping Ji · Dawen Yu · Chaoyong Shen · Weile Li · Qiang Xu

Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks

Abstract Convolution neural network (CNN) is an effective and popular deep learning method which automatically learns complicated non-linear mapping from original inputs to given labels or ground truth through a series of convolutional layers. This study focuses on detecting landslides from high-resolution optical satellite images using CNN-based methods, providing opportunities for recognizing latent landslides and updating large-scale landslide inventory with high accuracy and time efficiency. Considering the variety of landslides and complicated backgrounds, attention mechanisms originated from the human visual system are developed for boosting the CNN to extract more distinctive feature representations of landslides from backgrounds. As deep learning needs a large number of labeled data to train a learning model, we manually prepared a landslide dataset which is located in the Bijie city, China. In the dataset, 770 landslides, including rock falls, rock slides, and a few debris slides, were interpreted by geologists from the satellite images and digital elevation model (DEM) data and further checked by fieldwork. The landslide data was separated into a training set that trains the attention boosted CNN model and a testing set that evaluates the performance of the model with a ratio of 2:1. The experimental results showed that the best F₁-score of landslide detection reached 96.62%. The results also proved that the performance of our spatial-channel attention mechanism was fairly over other recent attention mechanisms. Additionally, the effectiveness of predicting new potential landslides with high efficiency based on our dataset is demonstrated.

Keywords Landslide detection · Satellite optical images · Convolution neural network · Attention mechanism · Remote sensing landslide dataset

Introduction

As a common and frequent geological disaster, landslide causes severe damages to natural environments, properties, and personal safety all over the world. For example, more than 200 people died from landslide disasters in Guizhou province of China between 2013 and 2017. A landslide may be triggered by several factors, such as seismic shaking, heavy rainfalls, and human activities, and result in the downward and outward movement of slope-forming materials including rock, soil, or their combination (Cruden and Varnes 1996; Cruden 1991). There are various studies based on field surveying or remote sensing data, which apply a wide range of methods on landslide monitoring, detection, potential landslide prediction, and hazard mitigation.

Field surveying for discovering potential landslide regions and updating landslide inventories are common and reliable strategies, but they are time-consuming, costly, and inefficient. With the rapid advancement of remote sensing technology, automatic landslide detection from aerial InSAR or satellite InSAR has been

widely applied (Bellotti et al. 2014; Sun et al. 2015; Czikhardt et al. 2017; Rosi et al. 2018). Landslide detection from optical images also has drawn attention increasingly. Digital elevation model (DEM) data, providing topographic information, plays an important role in landslide prediction and recognition (Dou et al. 2019; Barlow et al. 2003; Rau et al. 2013).

Regarding landslide detection as an image processing problem, statistical and machine learning methods have been extensively applied. For example, fuzzy logic (Ercanoglu and Gokceoglu 2002), logistic regression (Lee 2005), analytical hierarchy process (Kayastha et al. 2013), evidential belief function (Althuwanyee et al. 2012), weighted linear combination method (Ayalew et al. 2004), multivariate statistics analysis (Komac 2006), and fuzzy algebraic operators (Lee 2007) have been developed to discriminate landslide regions. Machine learning methods, such as support vector machine (SVM) (Marjanović et al. 2011), random forest (RF) (Catani et al. 2013), artificial neural network (ANN) (Ermini et al. 2005), have also been exploited. As the mainstream of current machine learning methods, convolutional neural networks (CNNs) have been applied (Yu et al. 2017). These supervised machine learning methods and most of the statistical methods require high-quality labeled landslide data as the training and assessment dataset, from which distinctive features of a landslide can be empirically designed or automatically learned. It is therefore crucial to construct labeled remote sensing landslide datasets for predicting, monitoring, and analyzing the landslide regions. However, up to now, there are no public remote sensing landslide datasets available, although there did exist a few studies and reports from governments or companies but the datasets are not accessible (Ghorbanzadeh et al. 2019).

The performance of landslide detection from remote sensing images heavily depends on state-of-the-art machine learning methods. Recently, on the strength of powerful representation learning ability achieved through a series of convolutions, CNNs have surpassed empirical feature designing methods in various image-based tasks, including image classification (Krizhevsky et al. 2012; He et al. 2016), object detection (Girshick 2015; Ren et al. 2015), and semantic segmentation (Ronneberger et al. 2015; Chen et al. 2017). However, the CNN-based methods designed for landslide detection are just getting started and there are only a handful of studies. Yu et al. (2017) trained a simple convolution network to retrieve the landslide images from the test image set. The discriminant area, boundary, and the center of the landslides were extracted using a region growing algorithm. Ding et al. (2016) used a CNN to extract the features of the pre-landslide images and the post-landslide images. The Euclidean distance between two feature vectors was calculated to identify whether a landslide occurred. Most recently, Ghorbanzadeh et al. (2019) analyzed the effects of different numbers of convolution layers, different sizes,

and channels of the input data on landslide detection. In all these studies (Ding et al. 2016; Yu et al. 2017; Ghorbanzadeh et al. 2019), very simple and basic network architectures, i.e., a series of convolutional layers and pooling layers, followed by fully connected layers, were adopted to detect landslides. The network depth is shallow (7 layers for Yu et al. 2017, 5 layers for Ding et al. 2016, 5 and 8 layers for Ghorbanzadeh et al. 2019). It is essential to develop more sophisticated and specific CNN architectures to extract distinctive representations of landslide regions from complicated backgrounds in future work.

Considering the lacking of open datasets and the requirement of advanced algorithms for landslide detection from remote sensing data, we made two main contributions in this paper:

1. We designed a novel attention module, which produced a 3D spatial and channel attention feature map, to emphasize the distinctive features of various landslide instances from complicated backgrounds. The attention module is integrated into the state-of-the-art CNN structures. The structures were considerably strengthened by the attention module in landslide detection. Our algorithm experimentally outperformed both the baseline networks and the ones boosted by other recent attention mechanisms.
2. We created an open, accurate, and large landslide dataset consisting of landslide/non-landslide images, shapefiles of landslide boundaries, and corresponding DEM data, which is available at <http://study.rsgis.whu.edu.cn/pages/download/>. It is the first open remote sensing landslide dataset with a careful threefold inspection. We believe it will promote the automatic landslide detection study from optical remote sensing images.>

Related work

A CNN architecture, usually consisting of convolutional layers, pooling layers, and full-connected layers, determines the basic feature representation ability. Since the large-scale AlexNet (Krizhevsky et al. 2012) was proposed, many well-designed architectures have been developed. The VGGNet (Simonyan and Zisserman 2014) stacked more convolutional layers and used smaller convolutional kernels, to make the decision function more discriminative and decrease the number of parameters simultaneously. In an Inception architecture (Szegedy et al. 2015), the feature maps from the filters of different sizes were concatenated to gain better representation capacity. Moreover, the fully connected layers were replaced with a global average pooling operation, in order to greatly reduce the parameter number and improve the accuracy. The ResNet (He et al. 2016) introduced skip connection to solve the gradient vanishing problem, in which a very deep network up to a thousand layers could be trained well. The DenseNet (Huang et al. 2017) concatenated the feature maps from different layers to make full use of the information of previous layers. Due to the success of the ResNet and Inception, many networks have been developed from them. The WideResNet (Zagoruyko and Komodakis 2016) argued that a higher accuracy could be achieved when using a wider but shallower network. The ResNeXt (Xie et al. 2017) showed that using the residual block with higher cardinalities brought the improvement of performance. The Xception (Chollet 2017) exploited the depth-wise separable

convolutions to obtain better performance and reduce computational burden. Some other light-weight networks (networks with fewer trainable parameters) were also developed for reducing computational complexity and parameter numbers, such as the SqueezeNet (Iandola et al. 2016) and MobileNet (Howard et al. 2017), to reach real-time applications.

These networks have been widely proven effective and of high performance, which have acted as the baseline algorithm for feature extraction both in close-range and remote sensing images. In this study, we developed our attention module based on these classic networks and choose the best one for landslide detection.

The attention mechanism is originated from the human visual system, which pays more attention to an essential or distinctive object (foreground) from the background of a scene or an image. In a CNN-based method, the feature maps learned from the input signal can be adjusted with the attention mechanism to strengthen the foreground. A CNN-based attention module can be described as a building block composed with convolution, activation, pooling, and fully connected layers. The attention module is integrated into backbone networks and trained without extra information. Specifically, the attention module takes the feature maps from a CNN and output regularized feature maps by emphasizing the foreground. Wang et al. (2017) introduced a residual attention block, which could be incorporated into the feed-forward networks and trained end-to-end. Hu et al. (2018) designed the squeeze-and-excitation (SE) block, which exploited the channel-wise interdependencies explicitly. The bottleneck attention module (BAM) (Park et al. 2018) and convolutional block attention module (CBAM) (Woo et al. 2018) proposed very recently focused on both spatial and channel dimensions of the extracted feature maps. To refine the intermediate feature maps, the BAM was placed at every bottleneck, i.e., after each pooling layer, while the CBAM was placed at every convolutional block.

In this work, we developed our 3D attention module which simultaneously extracts the integrated spatial and channel information in the stage of inferring the attention map, which is different from the above attention modules (Wang et al. 2017; Hu et al. 2018; Park et al. 2018; Woo et al. 2018), which treated space and channel separately. Our attention module obtained the best performance when compared with the advanced SE, BAM, and CBAM modules in the landslide detection task.

Study area and the open dataset

The study area covers the whole Bijie city with an area of 26,853 km², which lies in the northwest of Guizhou province, China (Fig. 1). The area is located in the transitional slope zone from the Tibet Plateau to the eastern hills, with the altitude ranging from 457 to 2900 m. The unstable geology, numerous steep hillsides, abundant rainfall (annual average rainfall is 849~1399 mm), and fragile ecological environment make this area to be one of the most serious landslide-prone areas in China. The landslide types in the Bijie city include rock falls and rock slides, and a few debris slides. Every year, many fresh landslides occur and some of them may cause devastating damages to human settlements, roads, bridges, transmission lines, and agricultural lands. Currently, the landslides here are mainly discovered by two methods or their combination. One is through indoor manual interpretations from satellite/aerial optical images and DEM, usually followed by accurate field surveying. The other is through villagers' reports, e.g.,

wall cracking or falling rocks. The visual interpretation based on images is inefficient, which requires relatively extensive geological knowledge as well. The field surveys are dangerous and time-consuming. The reports are usually lagging. There is a strong demand for automated, efficient, and reliable landslide detection methods for early warning, risk evaluation, and post-disaster recovery, especially for those emergent cases.

For developing automatic landslide detection methods, we created a remote sensing landslide dataset, the Bijie landslide dataset, consisting of satellite optical images, shapefiles of landslides' boundaries, and digital elevation models. Seven hundred seventy landslide samples, including rock falls, rock slides, and a few debris slides, and 2003 negative samples covering various backgrounds were cropped from the TripleSat satellite images captured from May to August 2018. The red points in Fig. 1 are the locations of actual landslides.

The sizes of landslide samples are different from each other and equal to the bounding box of current landslide instance plus 40-m extension as backgrounds. The various backgrounds are introduced to help train a network grasping the features of landslides themselves. The ground resolution of the RGB image is 0.8 m and the elevation accuracy of the DEM is 2 m. The shape vector of each landslide is manually delineated using ArcGIS. The non-landslide samples of mountains, villages, roads, rivers, forests, and agricultural lands were chosen manually to evaluate the performance of landslide detection methods comprehensively. Some landslide instances of this area are displayed in Fig. 2.

To prepare this landslide dataset, we took three steps sequentially. Firstly, according to the historical landslide inventory data (points with geographical positions) that were collected from the field survey, the landslides were initially located in the satellite images. Second, the boundary of each landslide instance was delineated carefully by geologists from the State Key Laboratory of Geohazard Prevention and Geoenvironment Protection of China. The position biases of a few landslides in the inventory data were corrected. A few landslides in the inventory which cannot be recognized from the image were removed as they can only be identified from field investigations and would confuse a CNN model. Some regions with obvious morphological characteristics of geological landslides are marked as new potential landslides. Finally, those potential landslide samples and those samples with large position biases from the inventory data were further confirmed by the careful field investigations of the authors, under the support of local officials in charge.

Methods

In this study, we develop a novel attention module to refine the feature maps of a CNN through emphasizing the important regions (i.e., the landslides) in a channel-spatial feature cube. In this section, we briefly review the theory of convolutional neural networks and attention modules, and then introduce our 3D attention module.

Convolutional neural network

CNN is a kind of feed-forward neural network, which takes the images as input and updates the network's parameters (i.e., the weights of convolutional kernels) automatically according to the difference between the output of the network and the given ground truth during the process of iterative training. After a successful training, the network (or the thousands of weights) will give a prediction result closest to the ground truth.

The architecture of a typical CNN contains convolutional layers, activation layers, pooling layers, and fully connected layers. A linear convolution layer is usually followed by a non-linear activation operation σ :

$$O^l = \sigma(O^{l-1} \times W^l + b^l) \quad (1)$$

where O^l represents the output feature map of the l th layer, W^l and b^l are the trainable convolutional kernel and bias of the l th layer respectively, and σ is the elementwise activation function, for example, ReLU ($\max(0, x)$) or sigmoid ($1/(1 + e^{-x})$), where x is the element of the output feature maps. The convolution operation is a linear computation and the followed activation function σ is used to increase the non-linear representation ability of the neural network. A convolutional kernel scans the entire input map pixel-by-pixel to generate an output feature map as shown in Fig. 3; multiple convolutional kernels can be used in a convolution layer to extract various features. The size of the convolutional kernel is settable, e.g., $3 \times 3 \times c$ or $5 \times 5 \times c$, where c is the number of channels of the input. After several convolution and activation operations, a pooling layer is usually adopted to condense the feature maps and reduce the size of the feature maps for efficient computation of the following layers. There are two typical pooling operations, i.e., the maximal pooling and the average pooling, as shown in Fig. 4. They keep the maximal and average values of the pooling

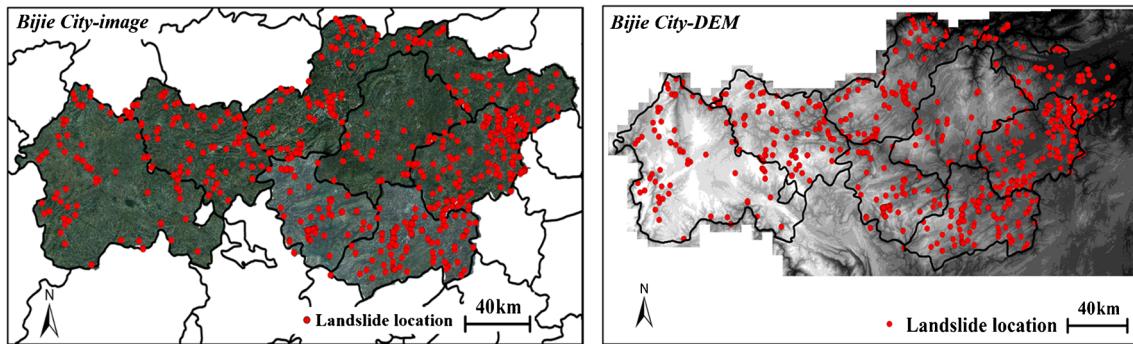


Fig. 1 The image (left) and DEM (right) data of the study area (the Bijie city); the red points are the locations of identified landslides

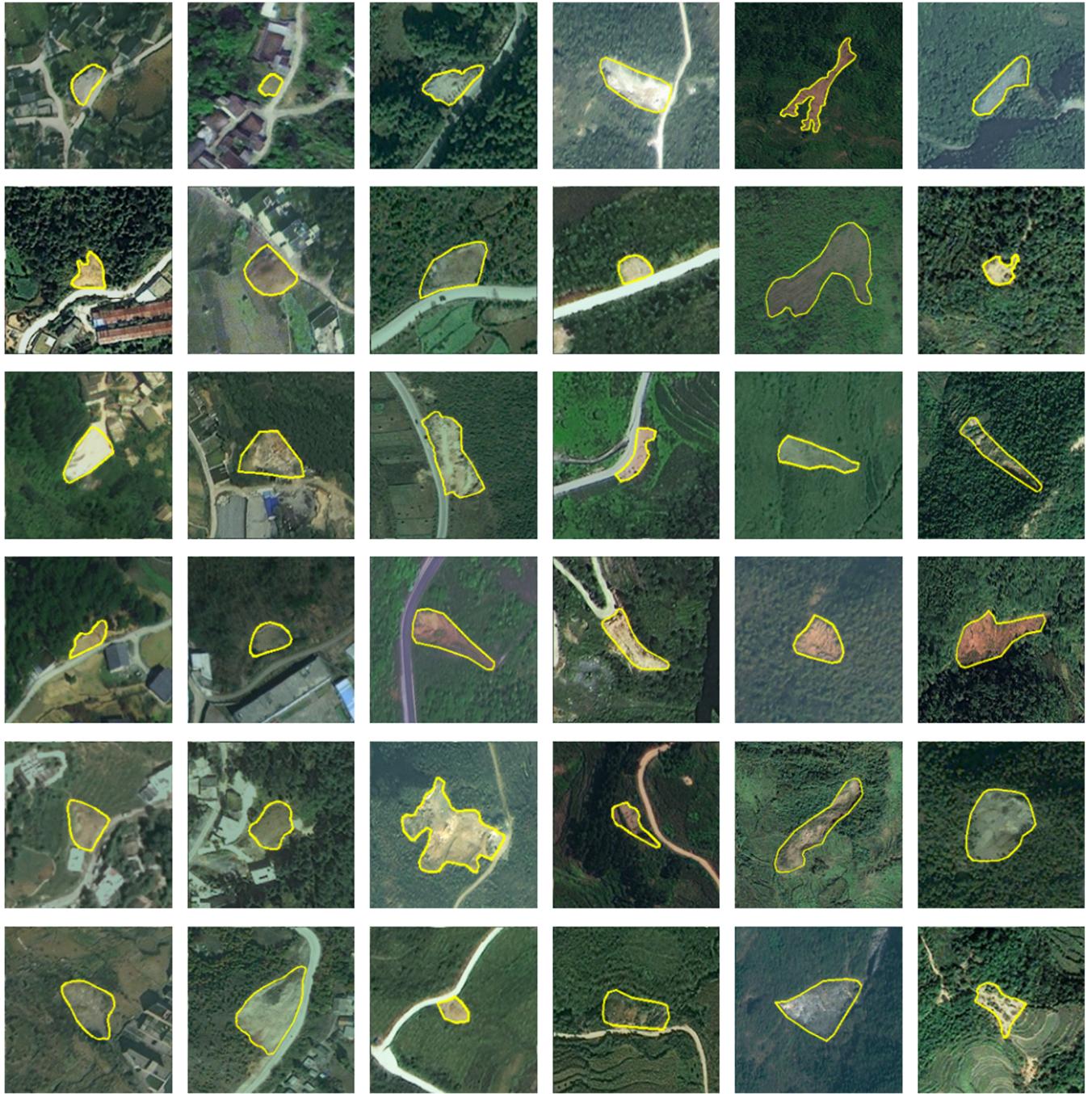


Fig. 2 Various landslide instances in the study area. The size of each sample is different. For each instance, a 40-m extension is preserved as background

windows in the input feature maps respectively. The size of a pooling window is typically 2×2 , which corresponds to a 2×2 downsampling. A “global” pooling means the size of pooling window equals to the size of the input, which downsamples a 2D input to 1×1 .

The convolution layers, the activation layers, and the pooling layers consist of a feature extractor (called encoder), while the fully connected layers at the end of the network (called decoder) are used to integrate and weight all the extracted features to a required

output (e.g., categories for a classification task, positions for an object detection task). The word “fully connected” means any neuron pair between two layers has a direct connection. Figure 5 shows a simple and popular network called VGG-16 (Simonyan and Zisserman 2014) that is used for our classification task, i.e., labeling an input image with one of the two categories, landslide or non-landslide. The VGG-16 consists of 16 neuron layers (13 convolutional layers and 3 fully connected layers) and 5 max-pooling operations. The sigmoid function is applied to the last

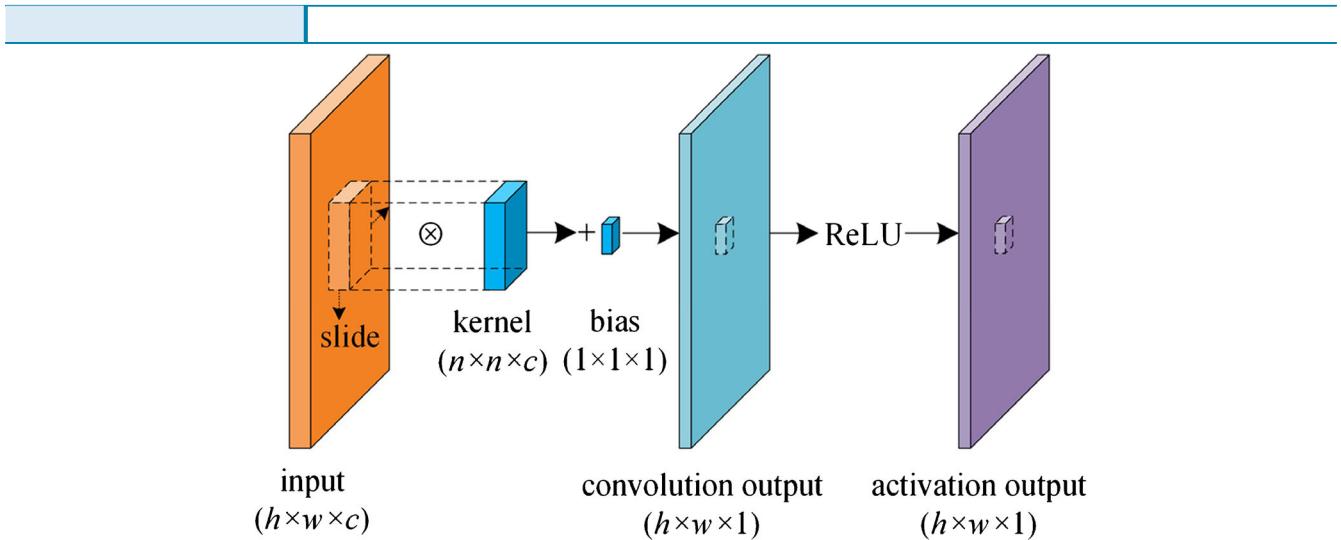


Fig. 3 The convolution operation in the CNN. The convolutional kernel W is applied on the input feature map or image pixel-by-pixel, then the results are added with the bias b to obtain the corresponding values in the output feature map, which is then activated by ReLU

fully connected layer and outputs two real numbers between 0 and 1 indicating the probability belonging to landslide and non-landslide, respectively. The network parameters are trained with given images and corresponding labels in a training set. Then the model is applied to new images to predict their categories.

The other popular CNN structure applied in our landslide detection is called ResNet (He et al. 2016). Figure 6 shows a shallow ResNet structure called ResNet-18, which consists of 17 convolutional layers, 1 fully connected layer, and 2 pooling operations. The ResNet is featured with “residual block,” where an addition operation (denoted as “ \otimes ” in Fig. 6) is applied between a previous layer and a current layer. This operation is called skip connection, which allows very deep networks up to more than 100 layers (e.g., ResNet-152) to be well trained. The batch normalization (BN) (Ioffe and Szegedy 2015) is used to normalize each input of a layer and speed up convergence.

Attention mechanisms

CNN-based attention mechanisms can be viewed as adaptive modules that emphasize some parts of the input feature maps and

suppress the other parts in a backbone CNN. An attention module is also composed with convolution, activation, pooling, or fully connected layers, which is trained along with the backbone CNN. The attention module in an image classification task is designed to emphasize foregrounds and resist to noisy backgrounds.

Recently, there are three popular attention modules for boosting the performance of a CNN. The squeeze-and-excitation (SE) module (Hu et al. 2018) focuses on the between-channel relationship for a given input feature map $F_{in} \in \mathbb{R}^{C \times H \times W}$ where C , H , and W represent the number of the channel, the height, and width of the feature map respectively. Firstly, the global average pooling (*AvgPool*) is used to squeeze and aggregate the spatial dimension information to obtain channel descriptor $F_c \in \mathbb{R}^{C \times 1 \times 1}$. Second, F_c is fed into a multi-layer perceptron (*MLP*) with one hidden layer to learn channel-wise dependencies. The *MLP* used here is actually two fully connected layers. Finally, a sigmoid activation σ is applied to produce the final SE attention map $M_c(F_{in}) \in \mathbb{R}^{C \times 1 \times 1}$.

$$M_c(F_{in}) = \sigma(MLP(AvgPool(F_{in}))) \quad (2)$$

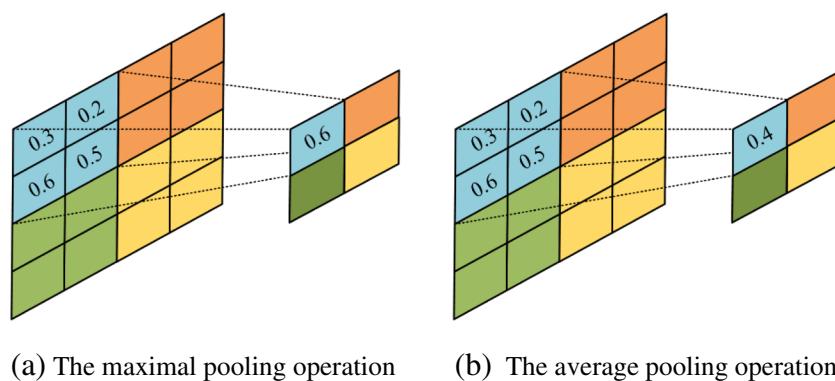


Fig. 4 The two typical pooling operations, both using a 2×2 pooling window. The maximal pooling (a) and the average pooling (b) keep the maximal and average values of the pooling windows of the input feature maps respectively. The size of the output feature maps is a quarter of the input

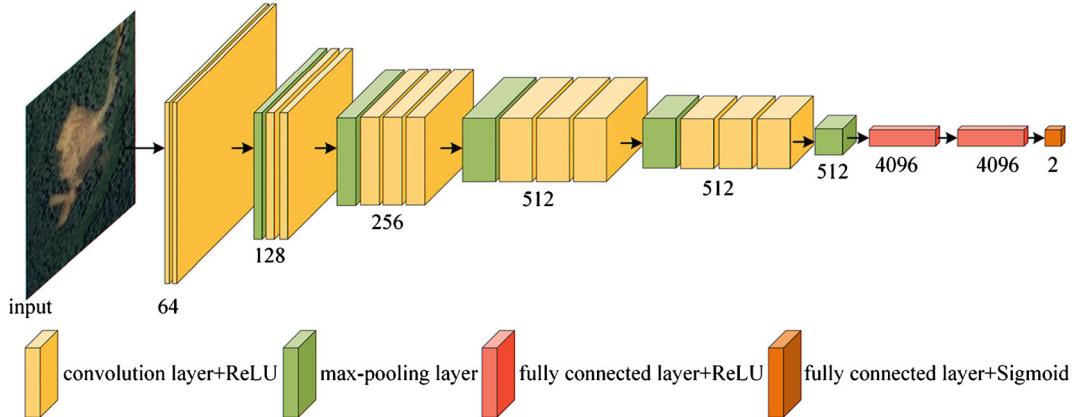


Fig. 5 The VGG-16 structure

The refined feature map $F_{out} \in \mathbb{R}^{C \times H \times W}$ is obtained by channel-wisely reweighting the input feature map F_{in} with the SE attention map $M_c(F_{in})$,

$$F_{out} = F_{in} \otimes M_c(F_{in}) \quad (3)$$

The symbol \otimes denotes element-wise multiplication throughout the paper.

The bottleneck attention module (BAM) (Park et al. 2018) leverages channel and spatial attention mechanisms separately. The

global *AvgPool*, the *MLP*, and the *BN* layer are applied sequentially to produce the channel attention map $M_c(F_{in}) \in \mathbb{R}^{C \times 1 \times 1}$,

$$M_c(F_{in}) = BN(MLP(AvgPool(F_{in}))) \quad (4)$$

The spatial attention branch in the BAM produces a spatial attention map $M_s(F_{in}) \in \mathbb{R}^{1 \times H \times W}$ to help the network discover which locations of the feature map should be focused on. The spatial attention map is obtained by performing four convolution operations and a *BN* operation sequentially,

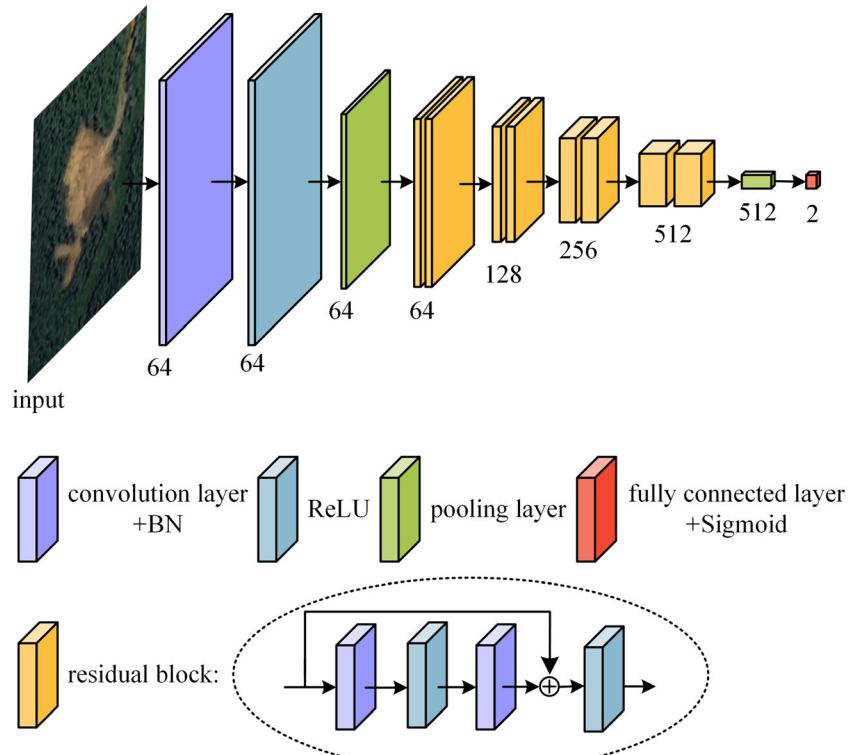


Fig. 6 The structure of ResNet-18. The circled “+” is an addition operation

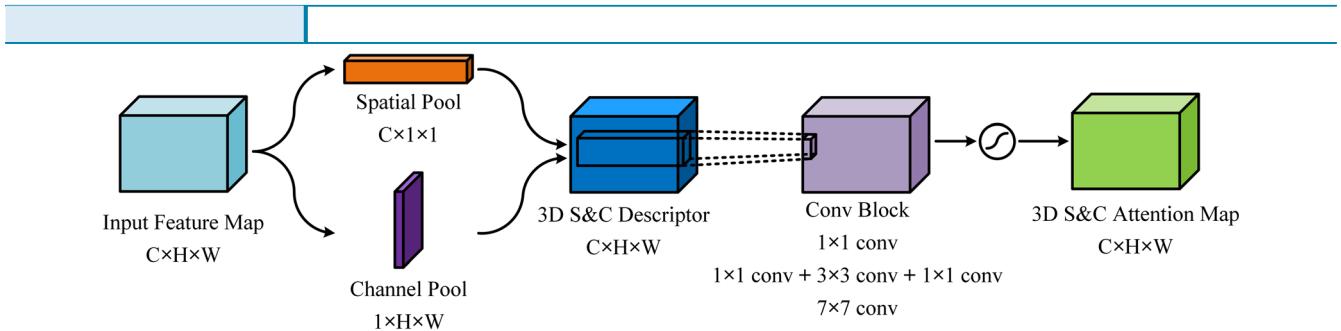


Fig. 7 The workflow of generating a 3D spatial and channel attention map

$$M_s(F_{in}) = BN\left(f_3^{1 \times 1}(f_2^{3 \times 3}(f_1^{3 \times 3}(f_o^{1 \times 1}(F_{in}))))\right) \quad (5)$$

The symbol f is a convolution operator and its superscript refers to the size of the convolutional kernel. The channel reduction ratio $r=16$ is used in the first (f_o) and last (f_3) convolution layers for reducing the number of parameters (i.e., the number of the feature map channels is reduced r times at f_o and then expanded r times at f_3) and computational cost. 3×3 dilated convolutions (Yu and Koltun 2015) in f_1 and f_2 are used to enlarge the receptive fields.

The final attention map $M(F_{in}) \in \mathbb{R}^{C \times H \times W}$ is obtained by combining the generated channel attention map $M_c(F_{in}) \in \mathbb{R}^{C \times 1 \times 1}$ and the spatial attention map $M_s(F_{in}) \in \mathbb{R}^{1 \times H \times W}$. Specifically, the two attention maps are firstly expanded to the size of $\mathbb{R}^{C \times H \times W}$, i.e., the $M_c(F_{in})$ is duplicated $W \times H$ times and $M_s(F_{in})$ is duplicated C times, then element-wisely summed, and finally activated by the sigmoid function. The computation process can be written as follows:

$$M(F_{in}) = \sigma(M_s(F_{in}) + M_c(F_{in})) \quad (6)$$

Similar to the BAM module, the convolutional block attention module (CBAM) (Woo et al. 2018) exploits a spatial submodule and a channel attention submodule to discover which locations and channels of the feature map should be strengthened or suppressed. In the channel attention submodule, both global average pooling (*AvgPool*) and global maximal pooling (*MaxPool*) are applied for aggregating spatial information, and then a weight-shared *MLP* is applied, followed by an element-wise summation and a sigmoid activation function σ :

$$M_c(F_{in}) = \sigma(MLP(AvgPool(F_{in})) + MLP(MaxPool(F_{in}))) \quad (7)$$

For the spatial attention submodule, a global average pooling and a global maximal pooling along the channel axis are applied;

the outputs of them are concatenated. The spatial attention map $M_s(F_{in}) \in \mathbb{R}^{1 \times H \times W}$ is generated by applying a convolution with the filter size of 7×7 and a sigmoid activation function σ sequentially to the concatenated feature descriptor.

$$M_s(F_{in}) = \sigma(f^{7 \times 7}([AvgPool(F_{in}); MaxPool(F_{in})])) \quad (8)$$

For a given input feature map $F_{in} \in \mathbb{R}^{C \times H \times W}$, the channel attention submodule and the spatial attention submodule of CBAM are carried out sequentially to refine it, i.e.,

$$F_{out} = F_{in} \otimes M_c(F_{in}) \otimes M_s(F_{in} \otimes M_c(F_{in})) \quad (9)$$

3D spatial-channel attention module

Our spatial and channel attention module (3D SCAM) is different from the above methods. We produce an integrated spatial and channel attention map instead of processing a channel and a spatial attention map separately; the latter cannot reach the global consistency between channel and space. The workflow of generating our 3D spatial and channel attention map is shown in Fig. 7. For a given input feature map $F_{in} \in \mathbb{R}^{C \times H \times W}$, the global pooling operations along the spatial and channel dimensions are applied respectively to obtain the channel descriptor $F_c \in \mathbb{R}^{C \times 1 \times 1}$ and the spatial descriptor $F_s \in \mathbb{R}^{1 \times H \times W}$ firstly. Then, the F_c and F_s are expanded to the size of $C \times H \times W$. Element-wise multiplication (Eq. 9) is used for computing the initial integrated 3D spatial-channel descriptor $F_{sc} \in \mathbb{R}^{C \times H \times W}$.

$$F_{sc} = F_s \otimes F_c \quad (10)$$

Then, a convolution block is applied to F_{sc} to refine the spatial-channel dependencies. The convolutional block consists of three subblocks (Fig. 8). A convolution with 1×1 kernel and a channel

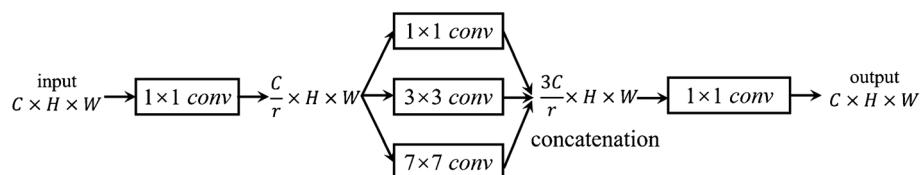


Fig. 8 The convolutional block in our 3D SCAM

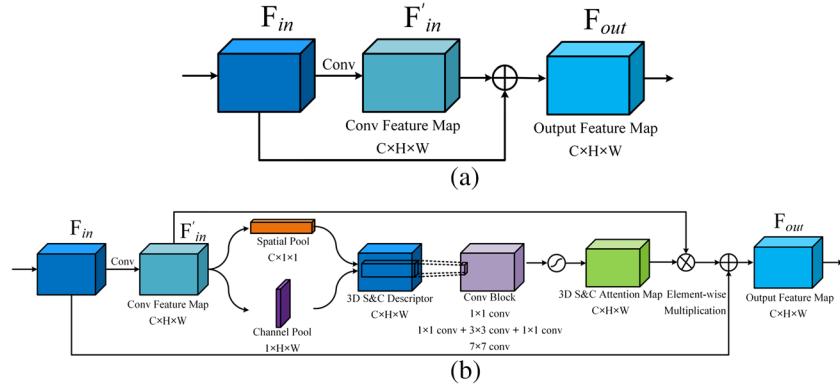


Fig. 9 The original residual block (a) and the 3D SCAM boosted residual block (b)

reduction ratio $r=16$ is applied in the first subblock for reducing computational burden and the number of parameters. The middle subblock contains three convolution operations with the kernel sizes of 1×1 , 3×3 , and 7×7 separately. Hence, the effective usage of the contextual information can be guaranteed by the different receptive fields. The outputs of the three convolutions are concatenated and fed into the third subblock with a kernel size of 1×1 to output a feature map with the same channel number as the input map. The computation process of the convolution block can be written as follows:

$$F'_{sc} = f_3^{1 \times 1}([f_2^{1 \times 1}(f_1^{1 \times 1}(F_{sc})); f_2^{3 \times 3}(f_1^{1 \times 1}(F_{sc})); f_2^{7 \times 7}(f_1^{1 \times 1}(F_{sc}))]) \quad (11)$$

where f is a convolution operator and the superscripts denote the sizes of the convolution filters.

Finally, a sigmoid activation σ is applied to compute the final 3D spatial-channel attention map $M_{sc}(F_{in}) \in \mathbb{R}^{C \times H \times W}$, i.e.,

$$M_{sc}(F_{in}) = \sigma(F'_{sc}) \quad (12)$$

We found both the global average-pooled descriptors and the global max-pooled descriptors are helpful for inferring a finer 3D attention map. Therefore, we apply both global average pooling and global maximal pooling during the process of generating our 3D attention map. The overall implementation process is summarized as follows: the global average pooling operations and global maximal pooling operations are applied parallelly to obtain the initial integrated spatial and channel descriptors F_{sc}^{avg} and F_{sc}^{max} , then the two descriptors are input into two convolution blocks $ConvBlock_1$ and $ConvBlock_2$ (Fig. 8) with non-shared weights, and the outputs are element-wisely summed and activated by the sigmoid function σ to produce the final 3D spatial and channel attention map M_{sc} . The whole process can be

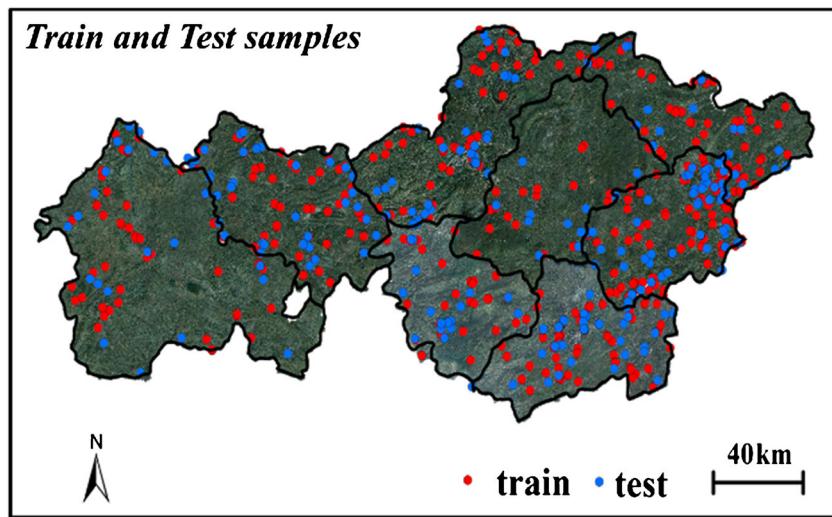


Fig. 10 The training (red) and test (blue) landslide samples of the study area

Model	Input	Precision	Recall	Accuracy	F_1 -score
VGG-16	RGB	0.9409	0.8677	0.9481	0.9028
VGG-16	DEM	0.6642	0.3463	0.7697	0.4552
VGG-16	RGM + DEM	0.9472	0.9066	0.9600	0.9264
ResNet-50	RGB	0.9595	0.9222	0.9676	0.9405
ResNet-50	DEM	0.6391	0.5720	0.7914	0.6037
ResNet-50	RGB + DEM	0.9602	0.9377	0.9719	0.9488

written as follows:

$$\begin{aligned} M_{sc}(F_{in}) &= \sigma(ConvBlock_1(F_{sc}^{avg}) + ConvBlock_2(F_{sc}^{max})) \\ &= \sigma(ConvBlock_1(F_s^{avg} \otimes F_c^{avg}) + ConvBlock_2(F_s^{max} \otimes F_c^{max})) \end{aligned} \quad (13)$$

Our 3D SCAM can be integrated easily into the advanced network architectures to refine the feature maps. Different from the attention methods of the SE, BAM, and CBAM, we place our 3D attention module in the last residual block of a ResNet-style network. We also experimentally found it is optimal to apply our 3D SCAM right after the “hidden layer” of a ResNet block rather than after the output map of the ResNet block; the details will be discussed in “[The placement of the attention module](#).” The original residual block and the 3D SCAM boosted residual block are shown in Fig. 9a, b separately. The attention boosting process of a residual block can be written as (13).

$$F_{out} = F_{in} + F'_{in} \otimes M_{sc}(F'_{in}) \quad (14)$$

Experiments and analysis

In this section, experimental settings and evaluation measures are described in “[Settings](#)”; the effects of DEM data on the results of landslide detection are analyzed in “[DEM-assisted landslide detection](#)”; the mainstream network architectures are compared in “[Comparison of different network architectures](#)”; our proposed attention mechanism is tested and compared to the other recent attention mechanisms in “[Comparison of different attention methods](#)”; discovering potential landslides using the model pretrained on our landslide dataset is given in “[Discovering potential landslides using the pretrained model](#).”

Settings

In all the experiments, two-thirds of the images and DEM data in the landslide dataset are used for training and the rest for testing; the locations of the training (red) and test (blue) landslide samples are shown in Fig. 10. A Linux PC with a GeForce GTX 1080 TI 11G GPU and an Intel i5-8400 CPU is used for executing all the experiments. All the networks share the same training settings. Specifically, we trained all the networks, which had been

Table 2 The landslide detection results on the Bijie landslide dataset using the popular network architectures. The italic numbers indicate the best results of each indicator

Model	Params	Precision	Recall	Accuracy	F_1 -score
VGG-13	128.96M	0.9249	0.9105	0.9546	0.9176
VGG-16	134.27M	0.9472	0.9066	0.9600	0.9264
VGG-19	139.57M	0.9274	0.8949	0.9514	0.9109
ResNet-18	11.18M	0.9549	0.9066	0.9622	0.9301
ResNet-50	23.51M	0.9602	0.9377	0.9719	0.9488
ResNet-101	42.51M	0.9673	0.9222	0.9697	0.9442
ResNet-152	58.15M	0.9639	0.9339	0.9719	0.9486
Inception-v3	21.79M	0.9562	0.9339	0.9697	0.9449
DenseNet-121	6.96M	0.9657	0.8755	0.9568	0.9184
DenseNet-169	12.49M	0.9504	0.8949	0.9578	0.9218
DenseNet-201	18.10M	0.9393	0.9027	0.9568	0.9206

Original Paper

Table 3 The landslide detection results of different attention modules on the ResNet-18. The italic numbers indicate the best results of each indicator

Description	Params	GFLOPs	Precision	Recall	Accuracy	F ₁ -score
ResNet-18 (baseline)	11.18M	1.86	0.9549	0.9066	0.9622	0.9301
ResNet18 + SE	11.27M	1.86	0.9551	0.9105	0.9632	0.9323
ResNet18 + BAM	11.20M	1.87	0.9555	0.9183	0.9654	0.9365
ResNet18 + CBAM	11.27M	1.86	0.9514	0.9144	0.9632	0.9325
ResNet18 + 3D SCAM	11.43M	1.87	0.9597	0.9261	0.9686	0.9426

pretrained on the ImageNet dataset, for 60 epochs using the Adam optimizer with a batch size of 32. The initial learning rate was started from 10^{-4} and decreased by a factor of 10 after 30 epochs. The input images and DEM were resized to 224×224 pixels (299×299 pixels for Inception-v3; Szegedy et al. 2016), and the pixel values of images and DEM data were normalized to 0~1. Data augmentation methods including random horizontal and vertical flips, random rotations, random Gaussian and pepper-salt noises, and histogram equalization were adopted during the training stage. The network outputs the probability of an image/DEM belonging to a landslide, and a threshold of 0.5 was adopted. All the codes were implemented in the PyTorch deep learning environment.

We use precision, recall, accuracy, and F₁-score to evaluate the landslide detection performances of different methods. Precision is the ratio of correctly predicted landslides to the total predicted landslides, recall is the ratio of correctly predicted landslides to the ground truth, accuracy is the ratio of correct predictions to the total predictions, and F₁-score is the weighted average of precision and recall, is computed as follows:

$$F_1\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (15)$$

DEM-assisted landslide detection

The topographic information (elevation, slope, aspect) in a DEM can be supplementary to some confusing textures and shapes in

optical images, which would result in prediction errors. We chose two popular network architectures, VGG-16 (Simonyan and Zisserman 2014) and ResNet-50 (He et al. 2016), to evaluate the effect of introducing the DEM. Table 1 shows the four evaluation indexes after the DEM data was introduced were all relatively improved compared with the results without the DEM data. However, the results of only using DEM data is much worse, indicating the leading role of the optical images in landslide detection. In all of the subsequent experiments, we use the combination of RGB images and DEM data as input.

Comparison of different network architectures

The popular network architectures, VGGNet (Simonyan and Zisserman 2014), ResNet (He et al. 2016), Inception (Szegedy et al. 2016), and DenseNet (Huang et al. 2017), were evaluated in our landslide detection dataset. Some other variants based on these classic networks were not listed due to little performance differences and some were not compared due to memory constraints. These variants, including the adjustments of network's width or depth, and the introduction of depth-wise separable convolutions or group convolutions, only achieve slight and trivial improvements. Table 2 shows the ResNet structure has an overall advantage compared to other structures marginally. The ResNet-50 obtained the highest recall, accuracy, and F₁-score. The ResNet-101 reached the best precision, and the ResNet-152 obtained the best accuracy as well. The VGGNets have the most parameters but performed worst. The Inception-v3 is slightly lower than the ResNet-50 and

Table 4 The landslide detection results of different attention modules on the ResNet-50. The italic numbers indicate the best results of each indicator

Description	Params	GFLOPs	Precision	Recall	Accuracy	F ₁ -score
ResNet-50 (baseline)	23.51M	4.16	0.9602	0.9377	0.9719	0.9488
ResNet50 + SE	26.04M	4.16	0.9677	0.9339	0.9730	0.9505
ResNet50 + BAM	23.87M	4.24	0.9641	0.9416	0.9741	0.9528
ResNet50 + CBAM	26.05M	4.16	0.9719	0.9416	0.9762	0.9565
ResNet18 + 3D SCAM	27.54M	4.35	0.9878	0.9455	0.9816	0.9662

Description	Params	GFLOPs	Precision	Recall	Accuracy	F_1 -score
ResNet-101 (baseline)	42.51M	7.88	0.9673	0.9222	0.9697	0.9442
ResNet101 + SE	47.28M	7.90	<i>0.9873</i>	0.9066	0.9708	0.9452
ResNet101 + BAM	42.86M	7.97	0.9834	0.9222	0.9741	0.9518
ResNet101 + CBAM	47.29M	7.89	0.9679	0.9377	0.9741	0.9526
ResNet101 + 3D SCAM	46.54M	8.08	0.9681	<i>0.9455</i>	<i>0.9762</i>	<i>0.9567</i>

ResNet-152. As the ResNets have shown relatively better performance, we will compare the performances of different attention methods based on the ResNets.

Comparison of different attention methods

Four different attention modules, the SE module (Hu et al. 2018), the BAM module (Park et al. 2018), the CBAM module (Woo et al.

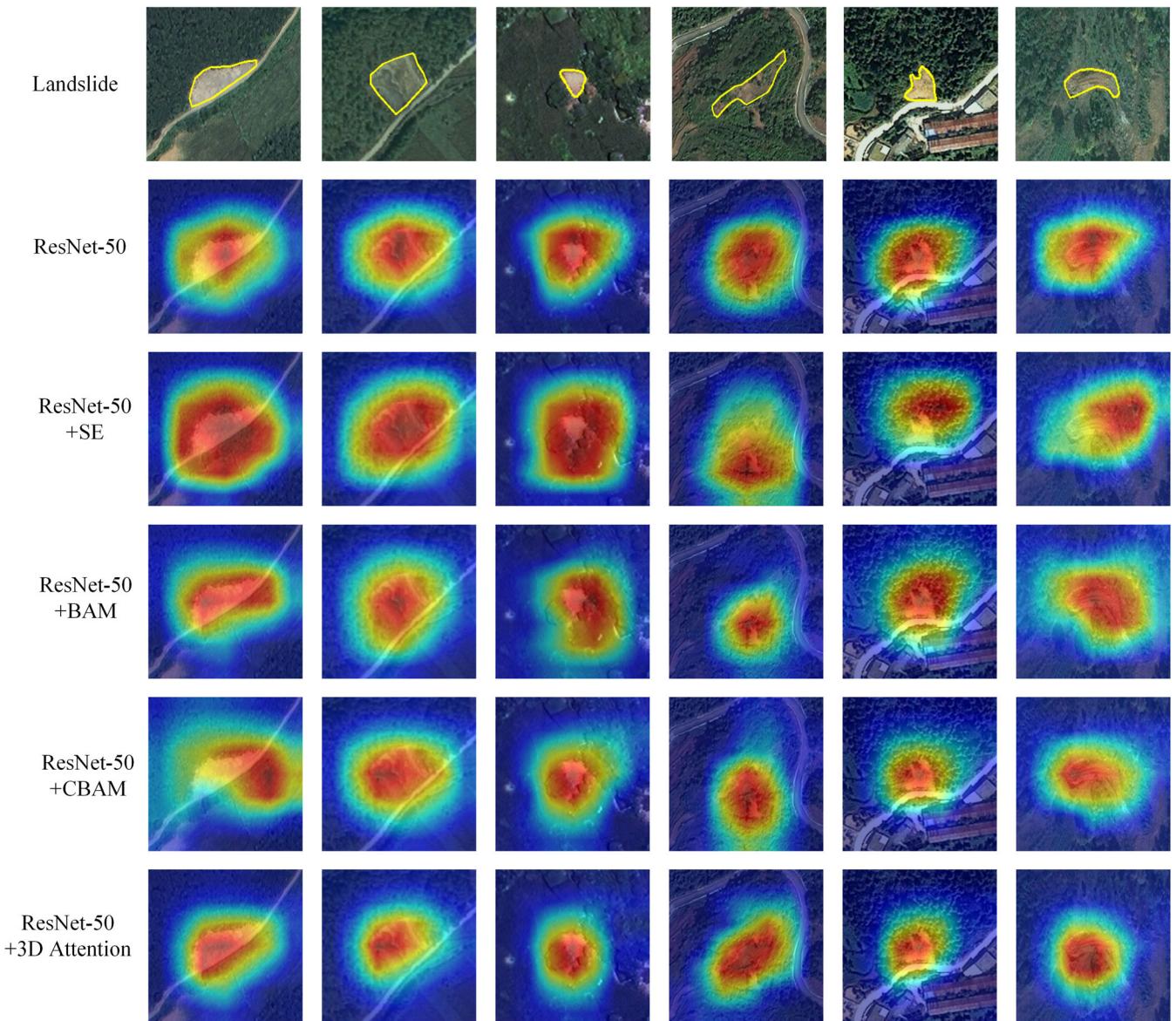


Fig. 11 The Grad-CAM visualization results of landslide samples

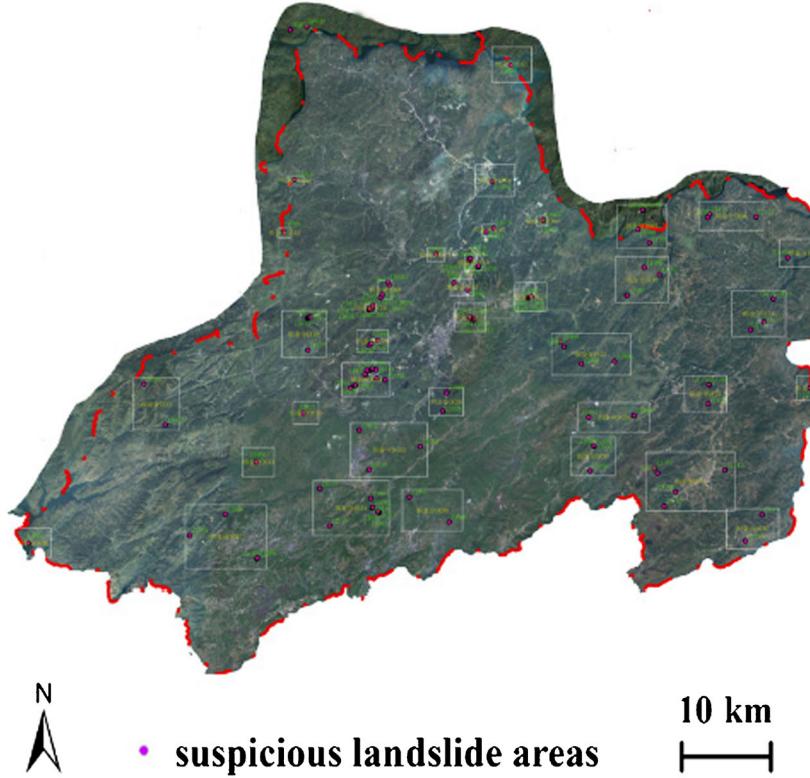


Fig. 12 The potential landslides (red points in rectangles) in Zhijin county, which need to be further confirmed by the field survey

2018), and our proposed 3D spatial-channel attention module (3D SCAM), were used for comparison. The SE module and the CBAM

module are integrated into the “hidden layer” of each residual block, the BAM module was integrated into the bottleneck of the

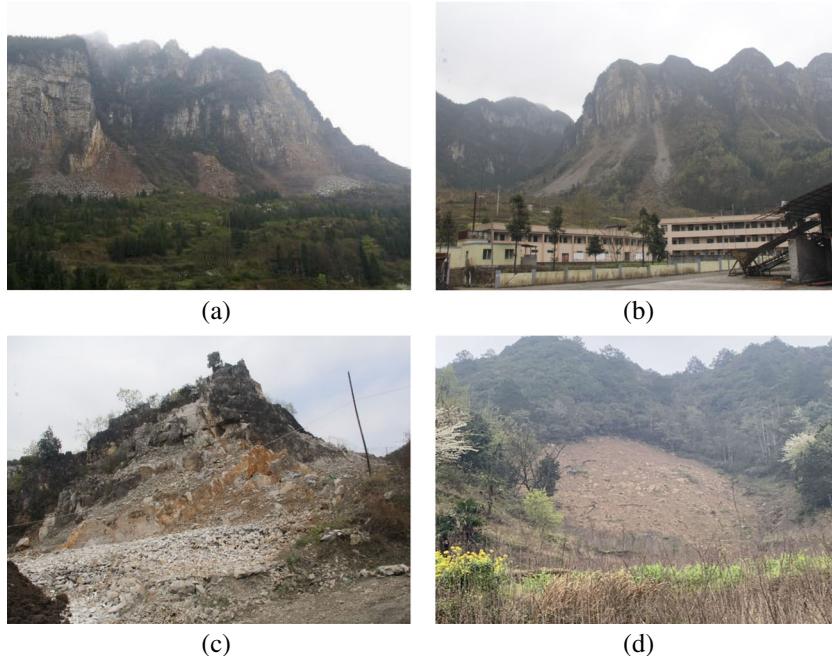


Fig. 13 The field survey of suspicious landslide areas predicted by the algorithm. **a** and **b** are correctly detected landslides. **c** is a false-positive case where a quarry is mistaken as a landslide. **d** is a case where a new landslide is identified by the algorithm which is neither on the province nor on the local landslide inventory

Description	Params	Precision	Recall	Accuracy	F_1 -score
ResNet-50 (baseline)	23.51M	0.9602	0.9377	0.9719	0.9488
Global average pooling (GAP)	25.53M	0.9758	0.9416	0.9773	0.9584
Global maximal pooling (GMP)	25.53M	0.9605	0.9455	0.9741	0.9529
GAP + GMP and shared <i>ConvBlock</i>	25.53M	0.9760	0.9494	0.9795	0.9625
GAP + GMP and non-shared <i>ConvBlock</i>	27.54M	0.9878	0.9455	0.9816	0.9662

residual networks, and our 3D SCAM was integrated into the hidden layer of the last residual block. The experiment results are shown in Tables 3, 4, and 5.

Firstly, the results from Tables 3, 4, and 5 show that all the networks imbedded with attention modules outperformed the baseline networks on accuracy and F_1 -score, which demonstrates the effects of attention mechanisms in landslide detection. Second, our 3D SCAM obtained overall improvement over all the other attention modules on all the indicators. The F_1 -scores of our 3D SCAM are 1.25%, 1.74%, and 1.25% higher than that of the baselines of ResNet-18, 50, and 101 respectively, and 1%, 1%, and 0.4% higher than the second-best attention modules respectively. Third, the ResNet-50 embedded with our 3D SCAM obtained the best performance in all the tests.

We used the Grad-CAM (Selvaraju et al. 2017) visualization method for the qualitative analysis of different attention modules with the ResNet-50 backbone. The Grad-CAM discovers which locations of the feature map are important for the final output through tracking the backward gradient flow. The visualization results of the SE, BAM, CBAM, and our 3D SCAM are shown in Fig. 11, where the heat maps of the last feature maps of predicted landslides are compared with the input signals. It is observed that the heat maps of our 3D SCAM cover the actual landslide regions more accurately than the other attention modules, indicating our method is more robust to the disturbance of various backgrounds.

Discovering potential landslides using the pretrained model

To demonstrate the validity of the proposed dataset and the attention boosted CNN model, we evaluate whether the model pretrained on the Bijie dataset can identify potential landslides and further help update the landslide inventory. Zhijin county, covering an area of 2868 km², is a landslide-prone region in the Bijie city. As the current inventory data may be incomplete and the fresh landslides tend to occur recently, we

applied our pretrained model (3D SCAM boosted ResNet-50) on the optical images captured from the same TripleSat satellite in early 2019. To reduce the workload of field surveying, 62 most suspicious landslide candidates (Fig. 12) are selected empirically by the geologists from the top 200 candidates predicted from the algorithm.

With the support of local officials in charge, the 62 candidates were on-site confirmed by the authors. It is revealed that more than 95% of candidates were real landslides. Some representative pictures of the field survey are shown in Fig. 13. Figure 13a, b is those obvious and typical landslides from a close-range observation, both of which were not listed on the province landslide inventory (and our dataset) but the local government officials claimed they had the records. Figure 13c is a rare case where a quarry was mistaken as a landslide. However, they are visually similar as both of them are featured with falling rocks and steep slope. This case can be avoided by comparing the predicted site with the record of local quarries. Figure 13d is a “successful case” as this landslide is neither on the province nor on the local landslide inventory. A local official remembered there was a landslide triggered by a heavy rain in 2016; however, it had been forgotten thereafter.

The results proved that our proposed method could predict those lurked landslides effectively and efficiently and reduce possible loss through prevention, even though the empirical checking process had reduced the search scope. Otherwise, in the indoor interpretation, a geologist must check every inch of the country instead of selecting 62 candidates from 200 samples.

Discussion

In this section, we discuss the impacts of different convolutional blocks, pooling strategies, reduction ratios in the MLP, and the

Table 7 Comparison of different filters in the convolution blocks of our 3D SCAM. The italic numbers indicate the best results of each indicator

Description	Params	Precision	Recall	Accuracy	F_1 -score
ResNet-50 (baseline)	23.51M	0.9602	0.9377	0.9719	0.9488
ResNet-50 + SCAM (1×1)	24.59M	0.9755	0.9300	0.9741	0.9522
ResNet-50 + SCAM (3×3)	24.86M	0.9836	0.9339	0.9773	0.9581
ResNet-50 + SCAM (7×7)	26.17M	0.9679	0.9377	0.9741	0.9526
ResNet-50 + SCAM ($1 \times 1 + 3 \times 3 + 7 \times 7$)	27.54M	0.9878	0.9455	0.9816	0.9662

Table 8 Comparison of different reduction ratio values in the 3D SCAM. The italic numbers indicate the best results of each indicator

Description	Params	Precision	Recall	Accuracy	F ₁ -score
ResNet-50 (baseline)	23.51M	0.9602	0.9377	0.9719	0.9488
ResNet-50 + SCAM (<i>r</i> = 4)	62.83M	0.9677	0.9339	0.9730	0.9505
ResNet-50 + SCAM (<i>r</i> = 8)	35.44M	0.9567	<i>0.9455</i>	0.9730	0.9511
ResNet-50 + SCAM (<i>r</i> = 16)	27.54M	<i>0.9878</i>	<i>0.9455</i>	<i>0.9816</i>	<i>0.9662</i>
ResNet-50 + SCAM (<i>r</i> = 32)	25.05M	0.9605	<i>0.9455</i>	0.9741	0.9529

placement of the attention module on the final classification results.

Global pooling methods

We compared the performance of different global pooling methods and their combination on our attention module. In Table 6, the combination of global average pooling and global maximal pooling (as parallel threads) can acquire higher scores. When the parameters of the convolution blocks in the two parallel pooling threads are not shared, the best F₁-score can be obtained. The conclusion is consistent with Woo et al. (2018).

The design of ConvBlock

The convolution concatenations with 1 × 1, 3 × 3, and 7 × 7 filters were utilized in the middle subblock of our 3D SCAM (Fig. 8), which is compared with separate filters in Table 7. Using the single convolution of 1 × 1, 3 × 3, or 7 × 7 kernels can improve the results of the baseline network, and the combination of them obtained the highest scores in all of the four evaluation measures. This is attributed to the filters of different sizes capturing different scopes of contextual information in the attention map.

The reduction ratio

The reduction ratio *r* in the convolution block of the 3D SCAM is a hyperparameter for alleviating the computational burden and adjusting the capacity of the attention module. The different values of *r* are tested and the results are shown in Table 8. It can be concluded that 16 is the optimal choice. Although more parameters are introduced, the performances of using 4 and 8 are lower than that of using 16, which may be due to the overfitting caused by the imbalance between the substantial extra parameters and limited training data. The relatively worse results of using 32 can be interpreted as the loss of some useful information caused by over-compressing.

The placement of the attention module

The attention module can be used for reweighting the feature maps of different places in a CNN. In this experiment, we placed our 3D SCAM module in two different places of the last residual block in the ResNet-50. In Fig. 14a, the module was used for refining the “hidden layer” of the ResNet block; in Fig. 14b, it was used for refining the output map of the ResNet block. In Table 9, it can be observed that the SCAM at both of the places can significantly improve the performance of the original ResNet-50, and using the attention module to refine the hidden layer is a better choice.

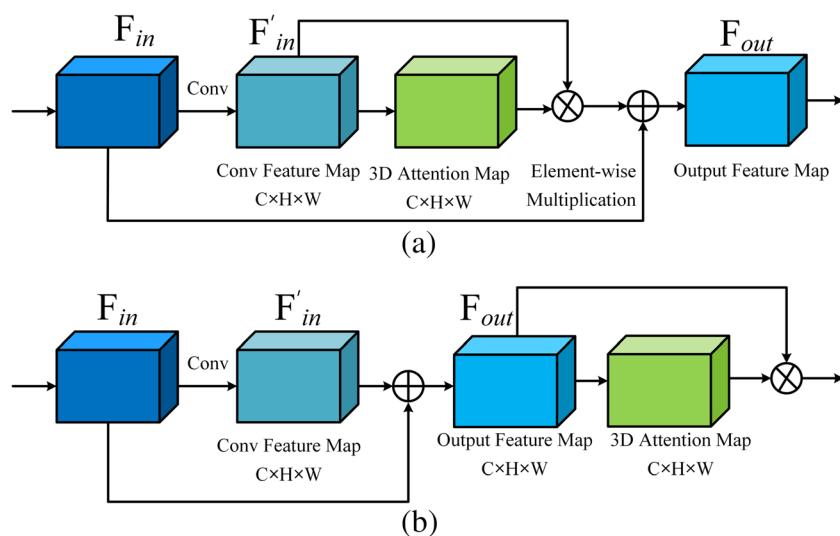


Fig. 14 Placing our SCAM module at different locations of the ResNet block. **a** After the “hidden layer.” **b** After the output layer

Description	Params	Precision	Recall	Accuracy	F ₁ -score
ResNet-50 (baseline)	23.51M	0.9602	0.9377	0.9719	0.9488
ResNet-50 + SCAM (a)	27.54M	0.9878	0.9455	0.9816	0.9662
ResNet-50 + SCAM (b)	27.54M	0.9720	0.9455	0.9773	0.9586

Conclusion

In response to the lack of open and accurate remote sensing landslide datasets, we create a large-scale landslide dataset which consists of satellite images with a 0.8-m ground resolution, delineated landslide boundaries, and high-accurate DEM. Through threefold inspection, i.e., the comparison between the historical landslide inventory and the satellite images, the landslide boundary delineation and position bias revision, and the field survey of potential landslides, the accuracy of the dataset was guaranteed. Moreover, we proposed a novel 3D attention mechanism, which extracts an integrated spatial-channel attention map with global consistency instead of separate spatial and channel maps extracted in other recent studies. The experiments proved our attention module is better than all the other recent attention modules, and the ResNet-50 boosted by our 3D attention module obtained the best result. The high scores of landslide detection in the test data, together with the successful experiment on discovering potential landslides on the Zijin country, demonstrated the great significance of the dataset, indicating the new applications and prospects of landslide detection using deep learning and optical satellite images.

Acknowledgments

We thank the geologists from the State Key Laboratory of Geohazard Prevention and Geoenvironment Protection for the indoor interpretation of landslide regions from satellite images; we thank the local officials who helped us check those potential landslides; we thank Yunhong Tian from Wuhan University and the University of Waterloo helped polish the English.

Funding information This work was supported by the National Key Research and Development Program of China, Grant No. 2018YFB0505003.

References

- Althuwanyee OF, Pradhan B, Lee S (2012) Application of an evidential belief function model in landslide susceptibility mapping. *Comput Geosci* 44:120–135
- Ayalew L, Yamagishi H, Ugawa N (2004) Landslide susceptibility mapping using GIS-based weighted linear combination, the case in Tsugawa area of Agano river, Niigata Prefecture, Japan. *Landslides* 1:73–81
- Barlow J, Martin Y, Franklin S (2003) Detecting translational landslide scars using segmentation of Landsat ETM+ and DEM data in the northern Cascade Mountains, British Columbia. *Can J Remote Sens* 29:510–517
- Bellotti F, Bianchi M, Colombo D, Ferretti A and Tamburini A (2014) Advanced InSAR techniques to support landslide monitoring. *Mathematics of planet earth*, Springer, pp 287–290
- Catani F, Lagomarsino D, Segoni S, Tofani V (2013) Landslide susceptibility estimation by random forests technique: sensitivity and scaling issues. *Nat Hazards Earth Syst Sci* 13:2815–2831
- Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2017) Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans Pattern Anal Mach Intell* 40:834–848
- Chollet F (2017) Xception: deep learning with depthwise separable convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1251–1258
- Cruden DM (1991) A simple definition of a landslide. *Bull Eng Geol Environ* 43:27–29
- Cruden D, Varnes D (1996) Landslide types and processes. Turner, AK & Schuster, RL (eds.) *Landslides: investigation and mitigation*, 36–75. Transportation Research Board Special Report 247
- Czirkhardt R, Papco J, Bakon M, Liscak P, Ondrejka P, Zlocha M (2017) Ground stability monitoring of undermined and landslide prone areas by means of sentinel-1 multi-temporal InSAR, case study from Slovakia. *Geosciences* 7:87
- Ding A, Zhang Q, Zhou X and Dai B (2016) Automatic recognition of landslide based on CNN and texture change detection. *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, IEEE, pp 444–448
- Dou J, Yunus AP, Bui DT, Merghadi A, Sahana M, Zhu Z, Chen C-W, Han Z and Pham BTJL (2019) Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed, Japan. *Landslides* 1:18
- Ercanoglu M, Gokceoglu C (2002) Assessment of landslide susceptibility for a landslide-prone area (north of Yenice, NW Turkey) by fuzzy approach. *Environ Geol* 41:720–730
- Ermini L, Catani F, Casagli N (2005) Artificial neural networks applied to landslide susceptibility assessment. *Geomorphology* 66:327–343
- Ghorbanzadeh O, Blaschke T, Gholamnia K, Meena SR, Tiede D, Aryal J (2019) Evaluation of different machine learning methods and deep-learning convolutional neural networks for landslide detection. *Remote Sens* 11:196
- Girshick R (2015) Fast r-cnn. Paper presented at the Proceedings of the IEEE international conference on computer vision, pp 1440–1448
- He K, Zhang X, Ren S and Sun J (2016) Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 770–778
- Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*
- Hu J, Shen L and Sun G (2018) Squeeze-and-excitation networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 7132–7141
- Huang G, Liu Z, Van Der Maaten L and Weinberger KQ (2017) Densely connected convolutional networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 4700–4708
- Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ and Keutzer K (2016) SqueezeNet: Alexnet-level accuracy with 50x fewer parameters and < 0.5 mb model size. *arXiv preprint arXiv:1602.07360*
- Ioffe S and Szegedy Claja (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*
- Kayastha P, Dhital MR, De Smedt F (2013) Application of the analytical hierarchy process (AHP) for landslide susceptibility mapping: a case study from the Tinau watershed, west Nepal. *Comput Geosci* 52:398–408
- Komac M (2006) A landslide susceptibility model using the analytical hierarchy process method and multivariate statistics in perialpine Slovenia. *Geomorphology* 74:17–28
- Krizhevsky A, Sutskever I and Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pp 1097–1105
- Lee S (2005) Application of logistic regression model and its validation for landslide susceptibility mapping using GIS and remote sensing data. *Int J Remote Sens* 26:1477–1491
- Lee S (2007) Application and verification of fuzzy algebraic operators to landslide susceptibility mapping. *Environ Geol* 52:615–623
- Marjanović M, Kovačević M, Bajat B, Voženilek V (2011) Landslide susceptibility assessment using SVM machine learning algorithm. *Eng Geol* 123:225–234
- Park J, Woo S, Lee J-Y and Kweon IS (2018) Bam: bottleneck attention module. *arXiv preprint arXiv:1807.06514*

- Rau J-Y, Jhan J-P, Rau R-J (2013) Semiautomatic object-oriented landslide recognition scheme from multisensor optical imagery and dem. *IEEE Trans Geosci Remote Sens* 52:1336–1349
- Ren S, He K, Girshick R and Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, pp 91–99
- Ronneberger O, Fischer P and Brox T (2015) U-net: convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, Springer, pp 234–241
- Rosi A, Tofani V, Tanteri L, Stefanelli CT, Agostini A, Catani F, Casagli N (2018) The new landslide inventory of Tuscany (Italy) updated with PS-InSAR: geomorphological features and landslide distribution. *Landslides* 15:5–19
- Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D and Batra D (2017) Grad-cam: visual explanations from deep networks via gradient-based localization. *Proceedings of the IEEE International Conference on Computer Vision*, pp 618–626
- Simonyan K and Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
- Sun Q, Zhang L, Ding X, Hu J, Li Z, Zhu J (2015) Slope deformation prior to Zhouqu, China landslide from InSAR time series analysis. *Remote Sens Environ* 156:45–57
- Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V and Rabinovich A (2015) Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1–9
- Szegedy C, Vanhoucke V, Ioffe S, Shlens J and Wojna Z (2016) Rethinking the inception architecture for computer vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2818–2826
- Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H, Wang X, Tang X (2017) Residual attention network for image classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 3156–3164
- Woo S, Park J, Lee J-Y and So Kweon I (2018) Cbam: convolutional block attention module. *Proceedings of the European Conference on Computer Vision (ECCV)*, pp 3–19
- Xie S, Girshick R, Dollár P, Tu Z and He K (2017) Aggregated residual transformations for deep neural networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1492–1500
- Yu F and Koltun VJapa (2015) Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*
- Yu H, Ma Y, Wang L, Zhai Y and Wang X (2017) A landslide intelligent detection method based on cnn and rsg_r. *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, IEEE, pp 40–44
- Zagoruyko S and Komodakis N (2016) Wide residual networks. *arXiv preprint arXiv:1605.07146*
-
- S. Ji · D. Yu**
School of Remote Sensing and Information Engineering,
Wuhan University,
129 Luoyu Road, Wuhan, 430079, China
- S. Ji**
e-mail: jishunping@whu.edu.cn
- D. Yu**
e-mail: yudawen@whu.edu.cn
- C. Shen**
Beijing Forestry University,
Beijing, 100083, China
- C. Shen (✉)**
The Third Surveying and Mapping Institute of Guizhou Province,
Guiyang, 550004, China
Email: whimfast@163.com
- W. Li (✉) · Q. Xu**
State Key Laboratory of Geohazard Prevention and Geoenvironment Protection,
Chengdu University of Technology,
Chengdu, 610059, China
Email: liweile08@mail.cduto.edu.cn
- Q. Xu**
e-mail: xq@cduto.edu.cn