CS291K Final Project Report: The Application of Pre-trained Models on Next-Generation Landslides Detection

Yuchen Hou
University of California, Santa Barbara
yuchenhou@ucsb.edu

Vihaan Akshaay Rajendiran
University of California, Santa Barbara
vihaanakshaay@ucsb.edu

Abstract

Landslide detection is critical for disaster risk management in vulnerable regions, where such events can profoundly impact communities. This paper explores deep learning-based pre-training approaches, inspired by Natural Language Processing, to detect landslides in satellite imagery. Our approach includes exploring various pre-training strategies to improve the model's recognition of landslide features. Additionally, we apply transfer learning from different datasets to assess its impact on detection capabilities. Utilizing advanced image encoder architectures within the Faster R-CNN framework, we found that a Swin Transformer backbone pre-trained using Masked Autoencoder yielded the best results in the landslide detection application. Our experiments demonstrate the potential of deep learning and transfer learning to refine landslide detection, underscoring the value of diverse dataset knowledge in the training process.

1. Introduction

1.1. Why Landslides?

Landslide is a type of natural hazard commonly observed in mountainous regions due to strong seismic, meteorological, or anthropogenic activities, and has affected about five million people worldwide [6]. According to a review published in Geoenvironmental Disasters, landslides have resulted in more than 55,000 deaths from 2004 to 2016, with overall losses estimated at USD 20 billion annually [22]. Therefore, it is crucial to predict landslides efficiently and provide early warnings to mitigate landslides' destructive impact on society and the environment [23].

1.2. Landslide Annotated Datasets

Landslides annotated datasets were generally based on small hand-curated datasets prepared by the researchers themselves, and there are two primary problems in preparing large datasets that can serve as benchmarks to evaluate custom methods or deep learning architectures. First, landslide events occur during different periods in time, and the satellite mission used differs from year to year (Fig. 1). Different satellites use different hardware, resulting in images having different resolutions and wavelength ranges resulting in different data distributions for the same region. Second, research groups use commercialized/licensed images and do not generally make their annotations public, making it hard to leverage the power of large deep-learning models for landslide detection.

A recently available dataset called "Landslide4Sense" [6] represents a major step forward to address these challenges. This dataset overcomes previous limitations by providing detailed landslide images with consistent resolutions and corresponding masks that delineate the shape and location of each landslide incident, facilitating precise landslide instance segmentation.

In this work, we primarily concentrate on constructing and evaluating deep-learning models using this public dataset for landslide prediction. We approach this by reframing the landslide detection challenge as an "instance segmentation problem." Our objective is to experiment with various pre-training methods to explore the potential improvements in the accuracy of landslide detection.

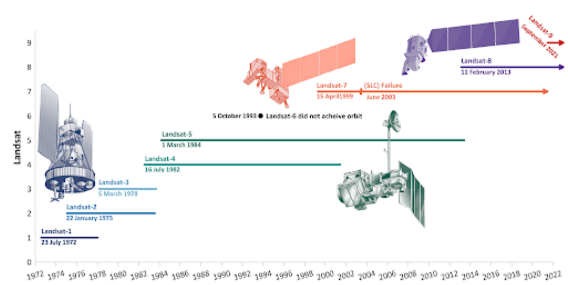

Figure 1. Different Landsat missions (satellites) that were active during different years.

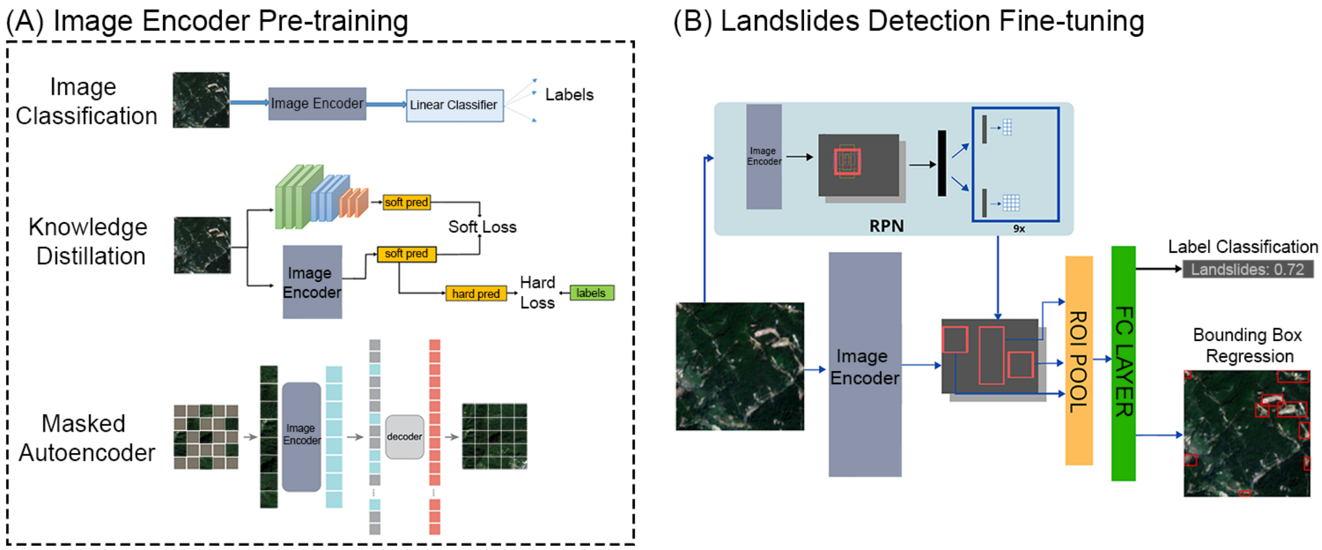

Figure 2. Proposed training pipeline overview. (A) An image encoder (purple) was pre-trained using three different pre-training methods. (B) The pre-trained encoder (purple) then served as the backbone in Faster R-CNN for landslide detection. Adapted from [10, 19, 21].

2. Related Works

2.1. Object Detection

Object detection is one of the most essential techniques in computer vision. It involves localizing the object of interest within images and classifying it based on categories. Inspired by transformers' success in Natural Language Processing, Vision Transformers have greatly improved object detection task performances by treating image patches as individual tokens and making use of the attention mechanism to extract complex local and global feature representations of each image [4].

Currently, there are two major types of deep-learning-based detection frameworks. The first one is single-stage object detection, which divides images into grids and predicts the bounding boxes along with object classification confidence scores for each grid [19]. The second method is two-stage object detection, which combines a region proposal framework for generating possible regions of objects and a neural network backbone for predicting each object's location and category [7].

2.2. Faster R-CNN

The Faster R-CNN architecture, proposed by Ren et al. [21], is one of the most influential two-stage object detection frameworks. It consists of three modules: a feature extraction image encoder backbone, a Region Proposal Network (RPN), and a detector. The backbone is a neural network that extracts complex input image features. The RPN takes the feature map produced by the feature-extraction backbone as the input. On each feature map position, a maximum k number of candidate boxes are predicted and passed to a convolutional layer with 3×3 filters. The resulting features are forwarded into two separate convolutional layers: the binary classification softmax layer that outputs $2k$ values to represent the estimated probability of whether the corresponding anchor contains objects or not, and the regression layer that outputs $4k$ values encoding the predicted bounding boxes coordinates for the objects detected in k anchors. The detector module takes the feature map produced by the backbone and object proposals produced by RPN as inputs. For each object proposal, a Region of Interest (RoI) max pooling layer extracts information from the feature map, which is then fed into fully connected layers to output the softmax probabilities for m categories plus 1 for background, as well as the refined four values for each of the per-class bounding box positions.

2.3. Vision Transformer and Swin Transformer

The Vision Transformer (ViT) model [4], utilizing the standard Transformer architecture, processes images by segmenting them into patches with positional embeddings, followed by feature extraction via the Transformer Encoder with multi-head self-attention. A Class token `[CLS]` is integrated for feature fusion and subsequent classification.

Building on ViT, the Swin Transformer [17] introduces a sliding window approach, incorporating the multi-scale concept from Convolutional Neural Networks (CNNs) with the multi-head self-attention mechanism. This allows for hierarchical feature extraction akin to CNNs while maintaining the structural integrity and global feature recognition capability inherent in ViT. Swin Transformer optimizes input data processing by reducing sequence length through window shifting, dividing the feature map into disjoint regions for localized attention, and facilitating information exchange between adjacent windows and through downsampling.

2.4. Landslides Detection

Traditional landslide detection methods require experts to manually label the landslide areas based on the textural and terrain characteristics derived from pixel-based, multi-scale image segmentation [23]. While this approach guarantees high accuracy, its labor-intensive nature prevents it from being applied to large-scale applications.

Recently, deep learning frameworks have been proposed for landslide detection using satellite images. For example, [18] implemented a YOLOv7-based model with an attention mechanism to localize landslides in Google Earth images, and it was able to achieve 93.94% F1-score. [8] proposed an Attention U-Net structure on Synthetic Aperture Radar amplitude satellite images to accurately map landslides across four datasets even under different cloud cover conditions.

However, these approaches mentioned above either require training from scratch or use natural image pre-trained weights to initialize the model. Training from scratch demands extensive computational resources, and utilizing pre-trained weights trained on natural images may not fully capture the unique characteristics of satellite imagery used for landslide detection.

Meanwhile, pre-training models using Masked Autoencoders (MAEs) or knowledge distillation has shown success in enhancing model performance across a range of tasks. Although the use of MAEs and knowledge distillation with satellite imagery has been extensively studied (e.g., [3, 13, 15, 16]), their effectiveness in landslide detection remains unexplored. Additionally, given that the pre-trained MAEs have shown superior performance on small object detection downstream tasks and are better suited to discriminate instances of the same class inside an image [5], investigating their impact on small landslide detection is particularly promising.

3. Methods

In this section, we present our proposed multiple-stage training pipeline (Fig. 2). We first fixed the pre-training strategy and searched for the best-performing backbone architectures for landslide detection. Then, we pre-trained the selected architectures using different pre-training strategies. The impact of pre-training was assessed by utilizing these architectures, equipped with pre-trained weights from different pre-training strategies, as the image encoder backbone within the Faster R-CNN framework to predict landslide bounding boxes in each satellite image.

3.1. Datasets

During *pre-training*, our image encoders were trained on three datasets: ImageNet-1K [14], fMoW-RGB [2], and Landslide4Sense [6]. See Fig. 3 for sample images in each dataset.

ImageNet-1K ImageNet-1K [14] is one of the most commonly used natural image datasets in the computer vision field. The dataset contains over 1 million images spanning 1,000 different object categories, ranging from animals and plants to everyday objects and scenes.

fMoW-RGB fMoW-RGB [2] contains high-resolution aerial-view satellite image time series across the world with 3 color channels and across 62 categories such as airports and zoos. In total, there are 712,874 images for training, 84,939 images for validation, and 84,966 images for testing.

Landslide4Sense Landslide4Sense [6] is a landslide segmentation dataset that contains 3,799 images by integrating spectral bands from Sentinel-2 with digital elevation models and slopes. Unfortunately, the original dataset link has been lost, and only a subset is available to the public. This subset includes 1,567 images labeled as "No-Landslide" and 2,231 images containing at least one "Landslide" label. For images with "Landslide" labels, corresponding binary masks are provided to represent the landslide locations.

During object detection *fine-tuning*, the Landslide4Sense dataset was used. To improve the computational efficiency and facilitate rapid convergence, we converted the landslide masks into bounding boxes by extracting the coordinates of each landslide location in the binary mask and calculating the topmost, bottommost, leftmost, and rightmost coordinates.

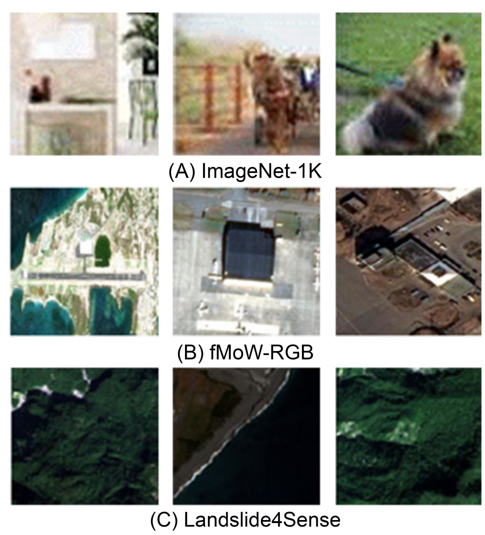

Figure 3. Sample images from ImageNet-1k [14], fMoW-RGB [2], and Landslide4Sense [6].

3.2. Pre-training Strategies

We then studied the effect of three distinct pre-training strategies on facilitating the performance of image encoders in the context of landslide detection: image classification, knowledge distillation, and Masked Autoencoder (Fig. 2A).

Image Classification We adopted the standard pre-training strategy in computer vision. That is, an image encoder (see Section 3.3 for encoders' architectures) was trained with the ImageNet dataset to predict the corresponding image category given an input image. The image encoder with trained weights was then used for the downstream landslide detection task.

Knowledge Distillation We "distilled" a more complex model into our smaller, task-specific image encoder. This was done by the following steps: 1) load the ImageNet-pre-trained weights onto a predetermined teacher model; 2) fine-tune the teacher model using the Landslide4Sense dataset and task the model to classify whether a given image contains landslide or not; 3) freeze the teacher model's weights; 4) train a smaller student model to predict the teacher model's soft target probabilities along with the ground-truth hard labels [11]. The loss function of the student is defined as:

$$L = (1-\alpha)L_{CE}(y, \sigma(z_s)) + \alpha T^2 L_{CE}\left(\sigma\left(\frac{z_t}{T}\right), \sigma\left(\frac{z_s}{T}\right)\right) \quad (1)$$

where α is a hyper-parameter that balances the soft probability loss with the hard groud-truth loss, L_{CE} is the cross-entropy loss, y is the ground-truth label of each image, σ is the soft-max function, z_t and z_s are the output probability estimated by the teacher and student models, and T is a temperature hyper-parameter [9].

Maksed Autoencoder Unlike the above two methods, the MAE pre-training strategy's self-supervised nature does not reply to image labels or categories, making it more applicable in the satellite images field where the amount of labels is much scarcer compared with the number of images. Following the original design [10], our MAE has an encoder that encodes masked images into tokens and maps them to high-dimensional space, and a decoder that learns from the encoded latent image features and reconstructs the original image. The loss function is the Mean Squared Error (MSE) between the original and the reconstructed images:

$$L = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (2)$$

where N is the total number of images, y_i is the ground-truth, and $\hat{y}_i$ is the corresponding reconstructed image. Only the masked image patches were included in the loss function.

3.3. Backbone Architectures

We searched for suitable deep-learning architectures that can be used for landslide bounding box detection by directly changing the backbone networks within the Faster R-CNN pipeline (Fig. 2B). There are two categories of backbone networks: CNN and Transformers.

For CNN-based backbones, we selected ResNet-18 and EfficientNet-B0 as these are frequently used in Faster R-CNN object detection (e.g., [21]). Other architectures such as DarkNet [19] and ResNet's other variants have been attempted during the early stage of the experiments, but due to computational resource constraints and time limits, these results are not represented in this study.

For transformer-based backbones, ViT [4] and Swin Transformer [17] were chosen because of their superior performance in traditional natural image-related computer vision tasks (e.g., [1]). We experimented with three variants of ViT (ViT-Tiny, ViT-Base, and ViT-Large) and two variants of Swin Transformer (Swin-Tiny and Swin-Base).

3.4. Training Pipeline and Experimental Setups

In total, we included three datasets, three pre-training strategies, and seven deep-learning backbone models in the context of landslide bounding box detection with a Faster R-CNN pipeline. All the dataset-strategy-model combinations can be found in Tables 1 and 2.

In terms of the *dataset*, we kept the original training-validation split configuration of the ImageNet-1K and fMoW datasets. The Landslide4Sense dataset was randomly divided into a training set and a validation set with an 8:2 ratio. Each image in all datasets was resized into 224×224, and only the RGB color channels were used for all the pre-training/fine-tuning tasks. Throughout the experiments, image augmentation techniques, such as random flip, rotate, or scaling were applied to the original images.

During the *pre-training* phase, we learned a set of weights that can be used for the downstream detection task for each model. Specifically:

- When the pre-training strategy was image classification: we utilized the pre-trained model weights available in the PyTorch framework for each backbone model.
- When the pre-training strategy was knowledge distillation: we set the teacher model to be ViT-Large/16 and the student model to be Swin-Tiny-Patch4-Window7 or Swin-Base-Patch4-Window7. Model parameters were optimized with Adam for 50 epochs to minimize the distillation loss between the teacher and student models. The learning rate was 0.0001, and the batch size was 16.
- When the pre-training strategy was MAE: model weights trained on Landslide4Sense data were optimized using AdamW with a 0.01 learning rate for 200 epochs, and the batch size was 48. Model weights trained on fMoW dataset were derived from [20].

During the landslide detection *fine-tuning* phase, we used the pre-trained models derived from the above methods as the backbone in Faster R-CNN architecture. The objective was to classify and localize the landslides from the background, and the number of classes was set to 2 (landslides and the background). The SGD optimizer was used with an initial learning rate of 0.001 and a momentum of 0.9. The model had a batch size of 8 and was trained for 90 epochs. An early stopping with a patience of 5 epochs was used.

All models were implemented in PyTorch and trained on two NVIDIA RTX 3090 with 24GB of memory. The code is available on GitHub: github.com/subawocit/cs291k and github.com/VihaanAkshaay/AI_Landslide.

3.5. Evaluation

The model performance was evaluated using the COCO standard evaluation metrics. The Intersection Over Union (IoU) is calculated based on the overlapped and union areas between the predicted and the ground-truth bounding boxes:

$$IoU = \frac{\textit{Area of overlapping}}{\textit{Area of union}} \tag{3}$$

The result is considered a correct detection (true positive) if the detection with IoU is larger than or equal to a predefined threshold and considered an incorrect detection (false positive) otherwise.

The precision and recall are calculated by:

$$Precision = \frac{\textit{true positive}}{\textit{the number of detections}} \tag{4}$$

$$Recall = \frac{\textit{true positive}}{\textit{the number of ground truths}} \tag{5}$$

Holding everything else constant, as the detection threshold increases, the precision will increase while the recall will decrease. The precision-recall curve represents this trade-off between precision and recall under different detection thresholds, and the area under the precision-recall curve is defined as the Average Precision (AP). The mean Average Precision (mAP) is the AP averaged across all classes of objects and various detection thresholds, with a higher mAP indicating better model performance. In this study, the mAP was computed by setting the IoU threshold to 0.5 (IoU $= 0.5$) or averaging the mAP values across 0.5 to 0.95 IoU thresholds (IoU $= 0.5 : 0.95$).

4. Experiments

4.1. Evaluating Backbone Architectures

We first investigated the performance of various backbone architectures on landslide object detection. To control for model capacity, the pre-training strategy was fixed

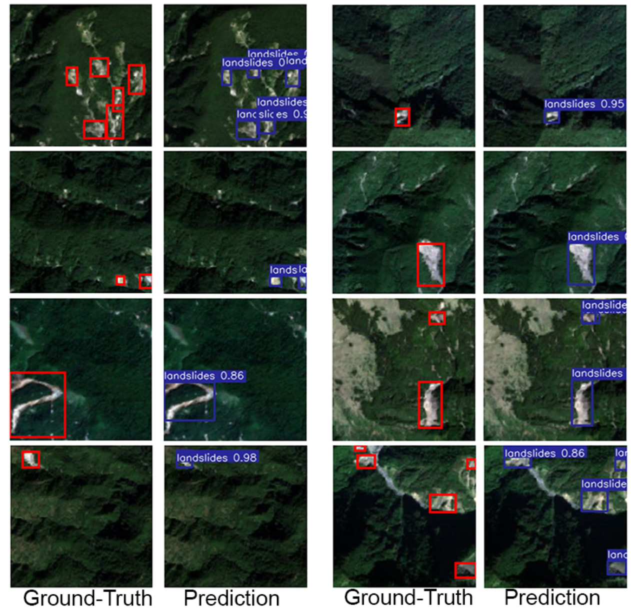

Figure 4. Sample predictions from Swin-Base backbone pre-trained on MAE.

to be image classification and the pre-trained dataset was ImageNet-1K for all architectures. As shown in Table 1, the Swin-Base outperformed other architectures across both evaluation metrics.

4.2. Evaluating Pre-training Strategies

We then selected the three best-performing architectures (i.e., ViT-Large, Swin-Tiny, and Swin-Base) and investigated the effect of various pre-training strategies on further improving the models' performance for this landslide detection task. Across all dataset-strategy-model combinations, the Swin-Base pre-trained using MAE yielded the best performance (Table 2).

Furthermore, we investigated the performance of the Swin-Base by inspecting the predicted images. As Fig. 4 shows, the model was able to detect the landslides in the validation set in most cases. The two most common detection mistakes across images were detecting white patches (e.g., snow) with complex shapes as landslides, and being unable to detect landslides when they are small in relation to the image (Fig. 5).

4.3. Additional Experimental Results

Additionally, we assessed the model's efficacy relative to the duration of MAE pre-training. Our analysis indicated that an optimal MAE pre-training period spanned 100 epochs, and extending this duration would result in diminished performance (Fig. 6).

Backbone	Pre-training		Performance	
	Strategy	Dataset	mAP (IoU $= 0.5 : 0.95$)	mAP (IoU $= 0.5$)
ResNet-18			<0.0010	<0.0100
EfficientNet-B0			0.0028	0.0125
ViT-Tiny			0.0980	0.2894
ViT-Base	Image Classification	ImageNet	0.1037	0.3205
ViT-Large			0.1048	0.3094
Swin-Tiny			0.1259	0.3544
Swin-Base			**0.1327**	**0.3766**

Table 1. Landslides object detection performance across different Faster-RCNN backbones. All backbones' weights were pre-trained on ImageNet classification. mAP (IoU $= 0.5$): mean average precision value when the intersection over union (IoU) is 0.5. mAP (IoU $= 0.5 : 0.95$): average mAP at 0.5 to 0.95 IoU thresholds. Cells with the highest performance score are marked in bold.

Backbone	Pre-training		Performance	
	Strategy	Dataset	mAP (IoU $= 0.5 : 0.95$)	mAP (IoU $= 0.5$)
ViT-Large	Image Classification	ImageNet	0.1048	0.3094
ViT-Large	MAE	fMoW	0.1015	0.3200
ViT-Large	MAE	Landslide4Sense	0.0800	0.2593
Swin-Tiny	Image Classification	ImageNet	0.1259	0.3544
Swin-Tiny	KD	Landslide4Sense	0.1251	0.3537
Swin-Tiny	MAE	Landslide4Sense	0.1304	0.3637
Swin-Base	Image Classification	ImageNet	0.1327	0.3766
Swin-Base	KD	Landslide4Sense	0.1168	0.3442
Swin-Base	MAE	Landslide4Sense	**0.1330**	**0.3853**

Table 2. Landslides object detection performance across different pre-training strategies. KD: knowledge distillation. MAE: Masked Autoencoder. Cells with the highest performance score are marked in bold.

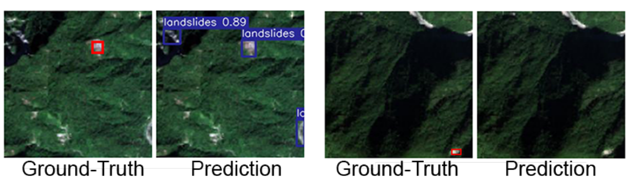

Figure 5. The model sometimes mistakenly identifies white patches with complex shapes as landslides (left) and struggles to detect landslides when they are small in size (right).

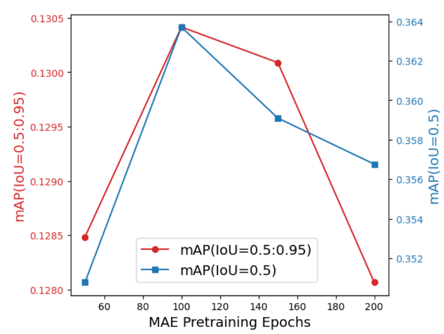

Figure 6. Model's landslide detection performance as a function of the number of epochs being pre-trained on MAE.

5. Discussion and Conclusion

We presented a comprehensive approach to landslide detection using deep-learning techniques, focusing on the use of pre-trained image encoder architectures within the Faster R-CNN framework. Our findings highlight the importance of selecting appropriate pre-training strategies and backbone architectures for improving landslide detection performance. The Swin-Base architecture, pre-trained using Masked Autoencoder (MAE), emerged as the most effective configuration, achieving superior performance in detecting landslide occurrences within satellite imagery.

5.1. Future Work:

- **Multi-spectral Information:** Several past studies have shown the effectiveness of using multi-spectral images in satellite fields (e.g., [12]). In future work, we aim to utilize various bands of multi-spectral landslide data to leverage the rich, spectral-specific information contained within multi-spectral imagery, thereby potentially improving our model's applicability for downstream tasks.

- **Edge-related Features:** Recognizing the importance of edge and crack features in the accurate prediction of landslides, similar to the approach taken by human annotators, we plan to explicitly incorporate these features into our models. This feature-guided strategy is expected to enhance our model's predictive accuracy by focusing on the critical visual cues that are indicative of landslide susceptibility.
- **Scale-Invariance & Scale-Specific:** Another area of focus will be to adapt models through the use of super-resolution and downsampling techniques, tuned specifically to achieve scale-invariance and scale-specificity (e.g., [20]). This will allow us to utilize models that have been pre-trained on the imagery of one scale for downstream tasks involving images of different scales. Such adaptability is crucial for the broad applicability of our models across diverse tasks and datasets, which often vary greatly in their scale and resolution.

As we look to the future, we are optimistic about the role of deep learning in revolutionizing landslide detection, offering a promising avenue for mitigating the impacts of landslides through timely and accurate detection.

References

[1] ImageNet benchmark (image classification) leaderboard. 4

[2] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6172–6180. ISSN: 2575-7075. 3

[3] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David B. Lobell, and Stefano Ermon. SatMAE: Pre-training transformers for temporal and multi-spectral satellite imagery. 3

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. 2, 4

[5] Paul Engstler, Luke Melas-Kyriazi, Christian Rupprecht, and Iro Laina. Understanding self-supervised features for learning unsupervised instance segmentation, 2023. 3

[6] Omid Ghorbanzadeh, Yonghao Xu, Pedram Ghamisi, Michael Kopp, and David Kreil. Landslide4sense: Reference benchmark data and deep learning models for landslide detection. 60:1–17. 1, 3

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. 2

[8] Steve Goley, Rohan Pradhan, and Austin Welch. Towards masked autoencoding pre-training for wide area motion imagery. In *Geospatial Informatics XIII*, page 8. SPIE. 3

[9] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks, 2017. 4

[10] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. 2, 4

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 4

[12] Danfeng Hong, Bing Zhang, Xuyang Li, Yuxuan Li, Chenyu Li, Jing Yao, Naoto Yokoya, Hao Li, Pedram Ghamisi, Xiuping Jia, Antonio Plaza, Paolo Gamba, Jon Atli Benediktsson, and Jocelyn Chanussot. SpectralGPT: Spectral remote sensing foundation model. 6

[13] Jeremy Irvin, Lucas Tao, Joanne Zhou, Yuntao Ma, Langston Nashold, Benjamin Liu, and Andrew Y. Ng. Usat: A unified self-supervised encoder for multi-sensor satellite imagery, 2023. 3

[14] R. Socher L. J. Li Kai Li J. Deng, W. Dong and Li Fei-Fei. "imagenet: A large-scale hierarchical image database,". 3

[15] Praveen Kandula and A. N. Rajagopalan. Distortion disentanglement and knowledge distillation for satellite image restoration. 60:1–11. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 3

[16] Junyan Lin, Feng Gao, Xiaocheng Shi, Junyu Dong, and Qian Du. Ss-mae: Spatial-spectral masked auto-encoder for multi-source remote sensing image classification, 2023. 3

[17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. 2, 4

[18] Lorenzo Nava, Kushanav Bhuyan, Sansar Meena, O. Monserrat, and Filippo Catani. Rapid mapping of landslides on SAR data by attention u-net. 14:1449. 3

[19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788. ISSN: 1063-6919. 2, 4

[20] Colorado J. Reed, Ritwik Gupta, Shufan Li, Sarah Brockman, Christopher Funk, Brian Clipp, Kurt Keutzer, Salvatore Candido, Matt Uyttendaele, and Trevor Darrell. Scale-MAE: A scale-aware masked autoencoder for multiscale geospatial representation learning. 7

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. 2, 4

[22] Kwan Ben Sim, Min Lee Lee, and Soon Yee Wong. A review of landslide acceptable risk and tolerable risk. 9(1):3. 1

[23] Hong Zhang, Mingzhe Liu, Tao Wang, Xin Jiang, Bingqi Liu, and Pengyu Dai. An overview of landslide detection: Deep learning and machine learning approaches. In *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 265–271. IEEE. 1, 3