Journal of Terramechanics 108 (2023) 33–38

Contents lists available at ScienceDirect

Journal of Terramechanics

journal homepage: www.elsevier.com/locate/jterra

Terrain classification using mars raw images based on deep learning algorithms with application to wheeled planetary rovers

Junlong Guo, Xingyang Zhang [*], Yunpeng Dong, Zhao Xue, Bo Huang [*]

State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150001, China

ARTICLE INFO

Article history:
Received 21 February 2023
Revised 11 April 2023
Accepted 17 April 2023
Available online 25 April 2023

Keywords:
Terrain classification
Deep convolutional neural network
Mars raw images
Wheeled planetary rover

ABSTRACT

Scene information plays a crucial role in motion control, attitude perception, and path planning for wheeled planetary rovers (WPRs). Terrain recognition is the fundamental component of scene recognition. Due to the rich information, visual sensors are usually used in terrain classification. However, teleoperation delay prevents WPRs from using visual information efficiently. End-to-end learning method of deep learning (DL) that does not need complex image preprocessing was proposed to deal with this issue. This paper first built a terrain dataset (consists of loose sand, bedrock, small rock, large rock, and outcrop) using real Mars images to directly support You Only Look Once (YOLOv5) to test its performance on terrain classification. Because the capability of end-to-end training scheme is positively correlated with dataset, the performance of YOLOv5 can be significantly improved by exploiting orders of magnitude more data. The best combination of hyperparameters and models was achieved by slightly tuning YOLOv5, and data augmentation was also applied to optimize its accuracy. Furthermore, its performance was compared with two other end-to-end network architectures. Deep learning algorithms can be used in the future planetary exploration missions, such as WPRs autonomy improvement, traversability analysis, and avoiding getting trapped.

1. Introduction

Due to the complex unstructured surface, autonomous mobility of wheeled planetary rovers (WPRs) plays a crucial role in Mars exploration. Safely accessing rough and steep terrain is essential to expand the search space to increase scientific value. Most of the recent review work on WPRs focuses on trajectory planning, localization, and obstacle avoidance (except terrain recognition).

The Opportunity got stuck in a sand dune in Meridiani Planum (Cowen, 2005), and the engineers took five weeks to resume the rover from this entrapment. The twin mars exploration rover, Spirit, got trapped in late 2009, and the mission was declared concluded on May 24, 2011. While Curiosity's wheels were injured due to sharp terrain features (Rankin et al., 2020).

Terrain classification can be used to avoid these aforementioned hazardous scenarios. Terrain classification focuses on safe navigation, and deals with the problem of identifying terrain being traversed (or to be traversed) from a list of candidate terrains (Ojeda et al., 2006).

There are two main bodies of research for terrain classification. Because of the intuitive, rich shapes, colors and textures, the most popular body of research uses visual cameras to classify ground surface by extracting color, texture, or objects like rocks (Bellutta et al., 2000; Manduchi et al., 2005). The second body of research uses both proprioceptive and exteroceptive signals, such as acoustic signals (Brooks and Iagnemma, 2005; Valada et al., 2018), motor current (Ojeda et al., 2006), accelerations (Giguere and Dudek, 2009), radar (Walas and Nowicki, 2014), and vibration (Sadhukhan et al., 2004).

Furthermore, with the development of artificial intelligence, deep learning (DL) technique was also applied to conduct terrain recognition because of its powerful learning capabilities and scalability. Deep learning was first used in R-CNN (an important DL method) to detect target, while DL can realize terrain recognition on real-time maps (Chavez-Garcia et al., 2017). Combination of DL and Bayesian classifier can be used to classify the upcoming terrains (Cheng et al., 2014). The performance of DL in terrain classification can be improved through data augmentation methods (Li and Hsu, 2020).

However, there are some challenges that hinder the application of DL models. The first challenge is the lack of high-quality datasets for model training. Developing effective strategies to improve existing DL models for terrain classification is the second chal-

[*] Corresponding authors.
E-mail addresses: 21s130372@stu.hit.edu.cn (X. Zhang), huangboweihai@hit.edu.cn (B. Huang).

https://doi.org/10.1016/j.jterra.2023.04.002

lenge. The authors used high resolution pictures of Mars surface to build high-quality datasets for model training. Hyperparameters optimization and data augmentation were applied to improve existing DL models. Considering the training cost, three typical DL model architectures, YOLOv5, CenterNet, and Faster-RCNN were selected to detect terrains.

This paper is organized as follows: Section II introduces three typical DL model architectures. The collection, annotation, and enhancement of Martian terrain dataset are presented in Section III. Section IV presents a series of experiments to investigate factors influencing the optimal performance of different models. Finally, Section V summarizes this study.

2. Selected deep learning frameworks

Current mainstream target detection algorithms are developed based on anchor box, such as You-Only-Look-Once (YOLO; Redmon et al., 2016), Single Shot MultiBox Detector (SSD; Liu et al., 2016) and Region-CNN (R-CNN; Girshick et al., 2014).

YOLO divides an image into $n \times n$ sub-grids, and then places a fixed-size anchor box at the center point of each grid, and finally uses a regression model to predict bounding boxes and classes. The single-stage detector represented by the YOLO series only uses a network to determine category and location, and thus it is very fast.

Faster-RCNN (Ren et al., 2015) relies on the region proposal network, which can identify plenty of bounding boxes (Bboxes) to localize objects. Meanwhile, Faster-RCNN uses region of interest pooling to refine Bboxes. Fig. 1 shows the schematic diagram of Faster-RCNN architecture.

Faster-RCNN and YOLO are representatives of one-stage and two-stage target detection networks, respectively. Both candidate frame and complex calculation are needed in these two networks. CenterNet that is an anchor-free and NMS-free (Non-Maximum Suppression) model can be used to solve these aforementioned issues (Zhou et al., 2019). Anchor-based detectors generate multi-scale anchor boxes to detect target at a same pixel point, which improves target detection accuracy. But it also increases repetition detection rate of a same object, which greatly decreases detection speed. For one identical object, anchor-based detector will generate multiple detection boxes, and it is necessary to use NMS to select the best detection box.

Discriminating between positive and negative training samples is the difference between anchor-based and anchor-free detection algorithms (Zhang et al., 2020). Instead of multi-scale anchor boxes, anchor-free detectors convert positive samples detection into a critical point detection problem, which is an anchor-free and NMS-free model. Based on successful keypoint estimation networks, CenterNet detects object center and regression to their size. CenterNet regards objects as points and models an object as a center point of its bounding box. CenterNet is trained to predict all center points quickly. The bounding box size and other objects properties are inferred from the key-point feature without IoU-based (Intersection over Union) NMS or other post-processing.

Instead of bounding-box detection and regression, CenterNet extracts the peak points in the heatmap for each category independently through CNN operations, and uses three head layers to determine category number, center point positions, and bias terms. Its structure is shown in Fig. 2.

3. Methodology

3.1. Training data collection

The performance of DL depends on high-quality image datasets. Although open access datasets, such as ImageNet and MS COCO, can be used to detect object, neither can be used to do terrain recognition. The training dataset in this work was built using NASA's Planetary Data System (PDS) Imaging Atlas. The Martian surface was divided into five categories (loose sand, large rock, small rock, outcrop, and bedrock) in this study, and Table 1 lists the definitions of five terrains.

One thousand two hundred and fifty images were selected from more than 30,000 images in the PDS, and each category contains 250 images. First, rapid preliminary screening can remove a large amount of useless data. Second, images with various terrain features will be more likely to be selected as part of the dataset. Third, images whose features are difficult to parse will be replaced during model pre-training. Fourth, verify the availability of the dataset. Fifth, repeat the aforementioned four steps to construct the dataset.

Table 1 lists the number of occurrences per each class in 1250 images. Large rocks with more prominent structure shape could be found in various scenes, and thus its number of occurrences is higher compared to other terrains. Target detection annotation tool was then used to annotate these images, as shown in Fig. 3. Finally, high-quality terrain dataset was obtained using data cleaning methods.

3.2. Data annotation

LabelImg (Yu et al., 2019) is an open-source graphical image annotation application, which is developed to provide samples for machine learning. It is a Bounding Box that specifically annotates objects in images. Actors manually label images by assigning meaning to regions in images using labels. The human-labeled rectangles are fine-tuned during the model pre-training, such as changing rectangle size and position, resizing a large rectangle containing many invalid areas into two small rectangles.

3.3. Data augmentation

The size of the dataset will greatly influence the performance of the DL model. To further expand the size of the dataset, data

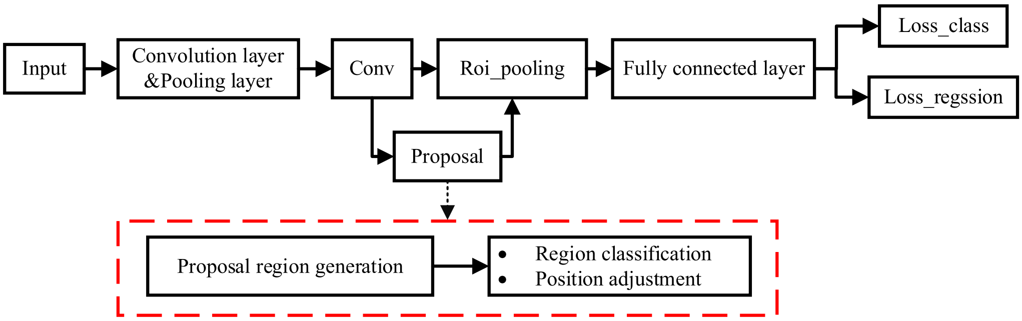


Fig. 1. Schematic diagram of Faster-RCNN architecture.

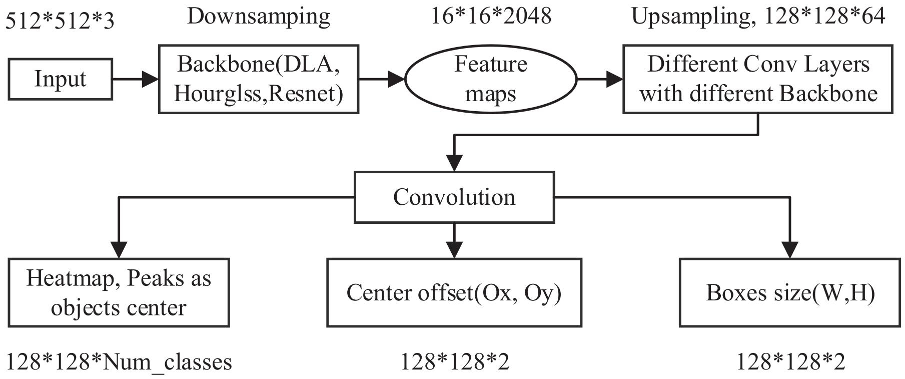


Fig. 2. Schematic diagram of CenterNet architecture.

Table 1
Detailed definition of terrain classes.

Class	Definition	Number of occurrences
Loose sand	Without any visible rocks, prone to high slip and wheel sinkage	277
Large rock	Large geometric obstacles	387
Outcrop	Hard and rough outcrop with loose rocks on top	250
Bedrock	Relatively flat exposed bedrock	270
Small rock	Do not interfere with wheeled rover and can easily be mistaken for large rock	268

enhancement techniques are performed on each image. These techniques include inversion, noise addition, rotation, grayscale conversion and brightness adjustment. The inversion technique uses 255 minus the corresponding RGB value to determine the RGB value of the new images.

Although the color of the target has changed, the color relationship between the object and the background still exists. The pictures obtained by the camera and the data processing process will be mixed with some noise. Adding salt and pepper noise can imitate the real data and increase the diversity of the data.

Rotation technology rotates images at various angles (90°, 180°, and 270°), and then the rotated images are resized and filled with black pixels. The rotated images need to be manually re-labeled. Grayscale conversion is the weighted summation of the RGB values. The phenomenon that camera is sensitive to brightness can be improved using brightness adjustment technology. Fig. 4 shows examples of data augmentation.

4. Experiment results

Two experiments were conducted to investigate the influence of various parameters and optimization methods on model performance. Fine-tuning and model hyperparameters (learning rate, number of iterations, mini-batch size, etc.) tuning through orthogonal method were carried out in experiment 1 to improve the performance of DL models. The performance of various DL models was then compared, and finally a relatively superior model was selected. Influence of various data augmentation techniques on the selected model performance was investigated in experiment 2.

The dataset was divided into training dataset, validation dataset, and testing dataset (70%, 15%, and 15%), while the augmented dataset was also divided into three parts (80%, 10%, and 10%). The

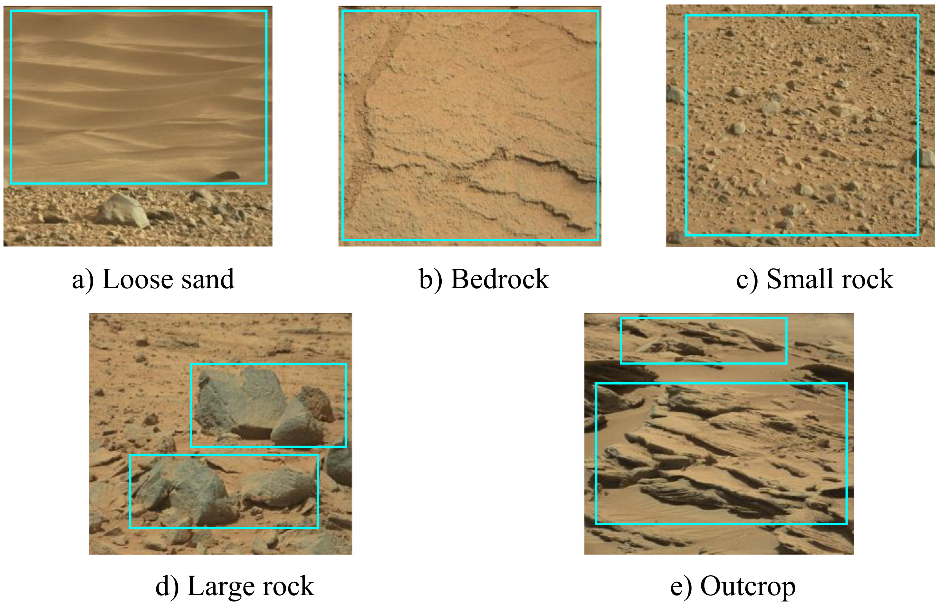

a) Loose sand
b) Bedrock
c) Small rock
d) Large rock
e) Outcrop

Fig. 3. Examples of manually labeled terrain.

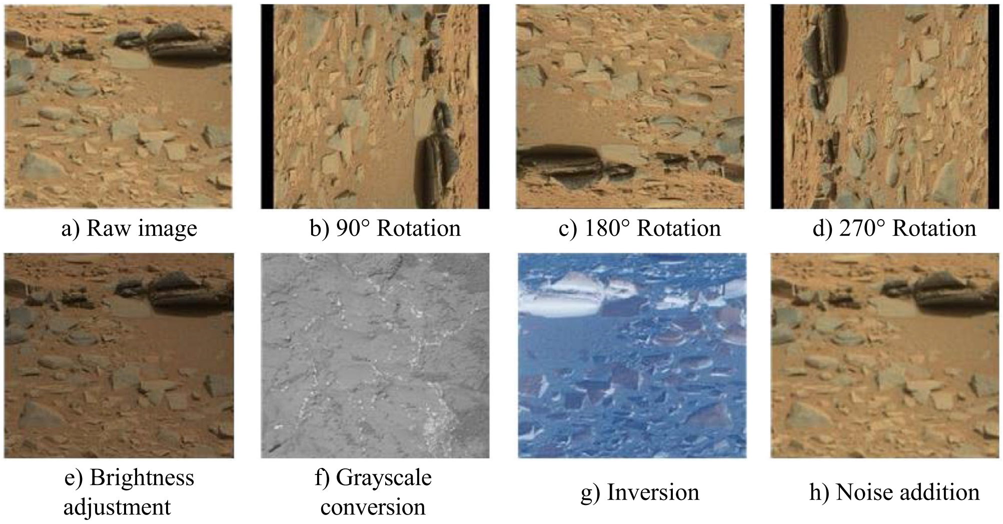

a) Raw image
b) 90° Rotation
c) 180° Rotation
d) 270° Rotation
e) Brightness adjustment
f) Grayscale conversion
g) Inversion
h) Noise addition

Fig. 4. Examples of image augmentation.

selected three DL models were deployed on Linux Ubuntu 18.04 LTS using Pytorch framework, and the experiments were carried out on NVIDIA Tesla P40 GPUs. The iteration number of each experiment was selected to ensure DL models' performance, and each experiment was performed twice to ensure repeatability.

4.1. Comparison of model performance using fine-tuning

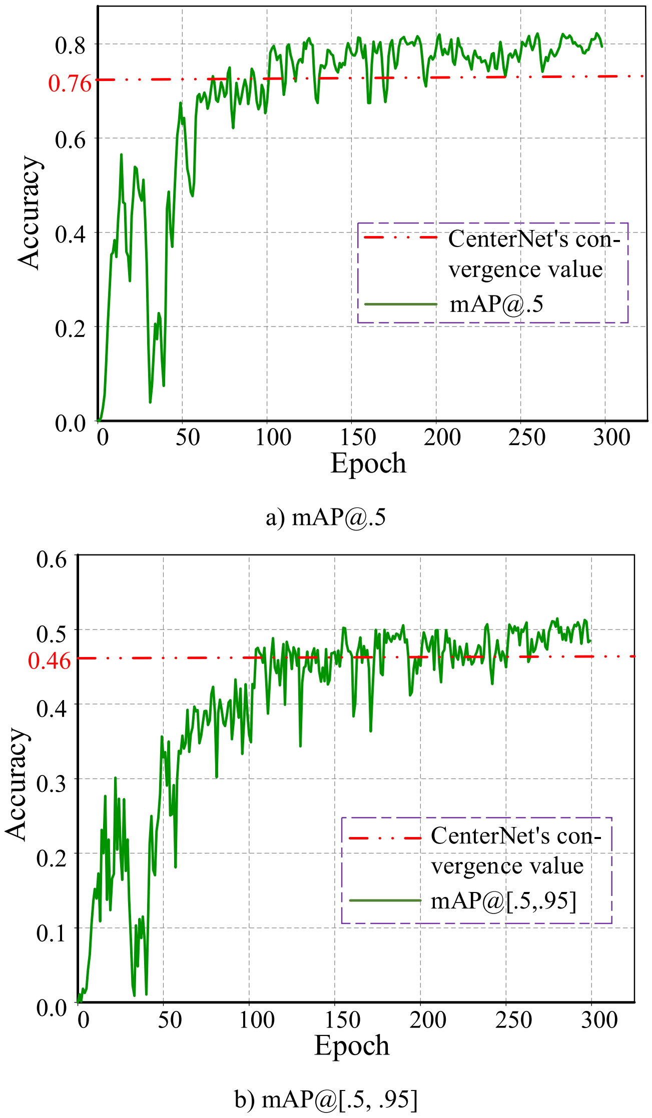
The fine-turned hyperparameters in this study contain learning rate, batch size, and image size. Metrics (mAP@0.5 and mAP@[0.5, 0.95]) and frames per second (FPS) were selected as performance indicators. In the field of object detection, it is generally believed that a good prediction result is obtained when IoU (Intersection over Union) between the predicted object position area and the actual area exceeds a set threshold. The mAP@0.5 (Mean Average Precision at Intersection over Union threshold 0.5) and mAP@0.5:0.95 (Mean Average Precision over different Intersection over Union thresholds, from 0.5 to 0.95, step 0.05) are COCO's standard metrics for all categories. They are model performance metrics obtained under various scenarios for the threshold values. FPS represents the number of predictable image frames per second of the trained model, aiming to compare the inference speed or real-time prediction ability of the model.

To compare their performance, each model was trained by running different iterations with a similar learning rate. The resolution of input images was about 600 × 600 pixels. During training process, the validation dataset was used to evaluate the intermediate model, and the model moved in a favorable direction.

Table 2 lists the performance of YOLOv5, CenterNet, and Faster-RCNN. In terms of averaged accuracy and FPS, YOLOv5 was selected to classify terrain. This is because YOLOv5 has a deeper structure and powerful design strategy. In addition, in terms of FPS, YOLOv5 and CenterNet perform much better than Faster-RCNN that is a two-stage model.

Table 2
Performance of YOLOv5, CenterNet, and Faster-RCNN.

Metric Model	mAP@0.5	mAP@[0.5, 0.95]	FPS
YOLOv5	0.776	0.485	76.9
CenterNet	0.771	0.479	45.3
Faster-RCNN	0.765	–	15.4



a) mAP@.5
b) mAP@[.5, .95]

Fig. 5. Accuracy of YOLOv5.

In terms of per unit training time, Faster-RCNN is about 3-times slower than CenterNet, and CenterNet is about 1.6-times slower than YOLOv5.

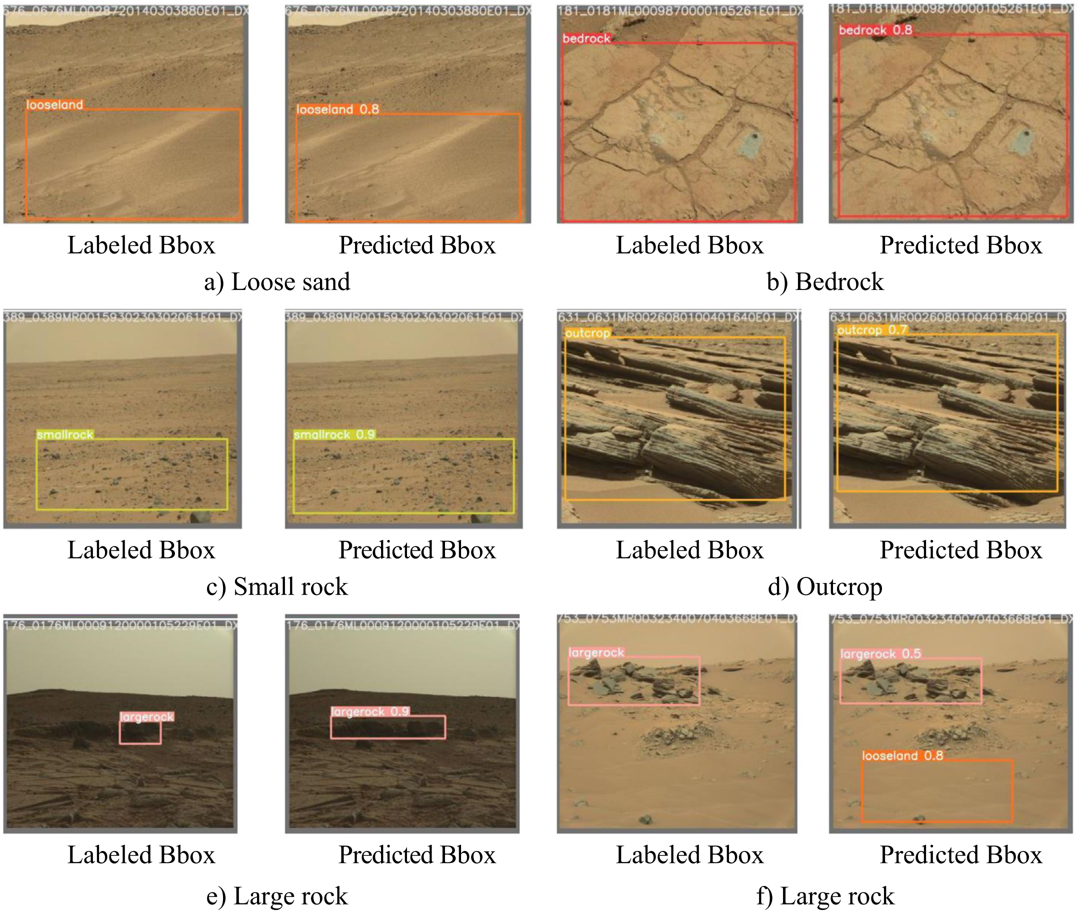
The curves of prediction accuracy versus epoch of YOLOv5 are shown in Fig. 5, and both mAP@0.5 and mAP@[0.5, 0.95] have similar convergence trends. The training epoch significantly affects the predictive accuracy. In the beginning of the training period, the mAP@0.5 fluctuates violently and even approaches zero, as illustrated in Fig. 5 a). The accuracy converges to a value of about 0.8 after 150 epochs, which indicates that the model converges. Fig. 5 b) is plotted under a comprehensive indicator. The mAP at IoU threshold 0.5 to 0.95 with a step size of 0.05 converges to around 0.5 after 150 epochs, as shown in Fig. 5 b).

The learning rate was adjusted through fine-tuning technology at the training stage. The initial value of learning rate plays a key role in the performance of DL model, while a comparatively high images resolution may obtain a better result.

Fig. 6 compares the labeled and predicted Bboxes using the optimal trained YOLOv5 model. Both terrain classes and Bboxes can be predicted correctly, and the confidence coefficient ranges from 0.5 to 0.9. The large rocks at far distance were confused with the small ones in near distance, and thus the Bbox of large rock in Fig. 6 f) was predicted with a confidence of 0.5. As shown in Fig. 6 e) and f), both the large rock and the loose sand that was not labeled can also be predicted, which demonstrates that YOLOv5 has powerful generation capacity.

4.2. Effect of data augmentation on model performance

Data augmentation extracts more information from the original dataset to build a more comprehensive dataset. Augmented dataset can be used to strengthen the generation capacity of a DL model in unknown dataset. Thus, experiment 2 was carried out to investigate the influence of data augmentation on terrain classification. As mentioned in Section 3.2, augmentation technologies were used to rebuild a larger dataset (containing 10,000 pictures). The experimental results are listed in Table 3, where Aft. Aug. stands for “After Data Augmentation”, and “Bef. Aug.” for “Before Data Augmentation”.



Labeled Bbox Predicted Bbox

a) Loose sand

Labeled Bbox Predicted Bbox

b) Bedrock

Labeled Bbox Predicted Bbox

c) Small rock

Labeled Bbox Predicted Bbox

d) Outcrop

Labeled Bbox Predicted Bbox

e) Large rock

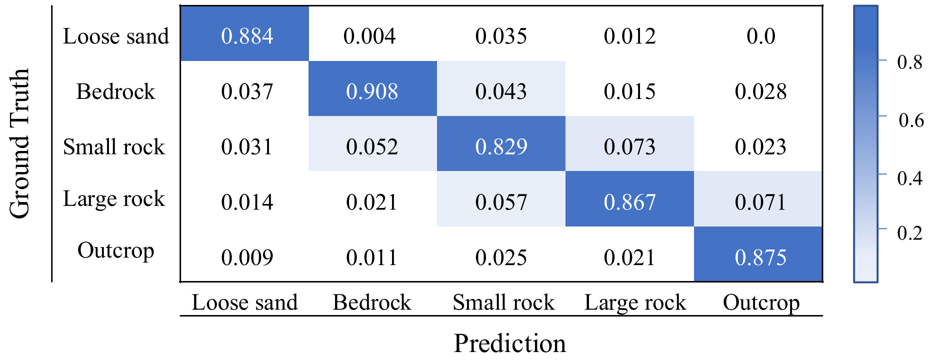
Labeled Bbox Predicted Bbox

f) Large rock

Fig. 6. Comparison of labeled and predicted Bboxes using the optimal trained YOLOv5 model.

Table 3
Prediction accuracy using data augmentation.

Classes Metric	Loose sand	Bedrock	Small rock	Large rock	Outcrop
AP@0.5 (Aft. Aug.)	0.945	0.928	0.852	0.872	0.900
AP@0.5 (Bef. Aug.)	0.853	0.858	0.697	0.731	0.741
AP@[0.5, 0.95] (Aft. Aug.)	0.743	0.682	0.484	0.678	0.653
AP@[0.5, 0.95] (Bef. Aug.)	0.634	0.553	0.343	0.385	0.510



Ground Truth \ Prediction	Loose sand	Bedrock	Small rock	Large rock	Outcrop
Loose sand	0.884	0.004	0.035	0.012	0.0
Bedrock	0.037	0.908	0.043	0.015	0.028
Small rock	0.031	0.052	0.829	0.073	0.023
Large rock	0.014	0.021	0.057	0.867	0.071
Outcrop	0.009	0.011	0.025	0.021	0.875

Fig. 7. Confusion matrix of terrain classification (after data augmentation).

The overall performance of terrain classification can be strengthened by data augmentation. After data augmentation, YOLOv5 can detect the classified five terrains with high accuracy. The AP@0.5 ranges from 0.852 to 0.945, while AP@[0.5, 0.95] ranges from 0.484 to 0.743.

Loose sandy terrain can be predicted with the highest accuracy, which can form a hazardous obstacle for WPRs. The detection accuracy of large rock is lower than most of the other terrains. Compared with other terrains, the size range of large rock is larger according to the classification criterion listed in Table 1, which increases the classification difficulty. Small rocks in near distance are frequently misclassified as large rock, and thus its prediction accuracy is the lowest. Confusion matrix of Mars terrain classification of YOLOv5 is shown in Fig. 7. After data augmentation, the prediction precision ranges from 0.829 to 0.908 for the five selected terrains. Data augmentation facilitates a significant improvement in generalization capacity of YOLOv5, but the misclassification rate between large rocks and small rocks is relatively high.

In conclusion, data augmentation can be used to improve the performance of DL models in terrain classification.

5. Conclusions and discussions

Real-time and accurate terrain identification can be used to prevent WPRs from getting trapped, and to slow down wheel wear. A Martian terrain classification method with DL models was proposed in this paper. (1) A dataset of Mars scenes was first developed; (2) Three typical DL models (YOLOv5, CenterNet, and Faster-RCNN) were then selected to demonstrate their performance in Martian terrain classification; (3) Transfer learning and data augmentation were finally deployed to strengthen model's generation capacity. Comprehensive experiments show that YOLOv5 is the best choice to conduct terrain classify-cation.

Though the five terrain classes can be identified correctly using YOLOv5, a more robust terrain recognition framework should be developed to identify terrain in real time with a higher accuracy by combining DL model with exteroceptive sensors and proprioceptive sensors. In addition, terrain recognition research based on unsupervised learning methods is also emerging, which will provide more possibilities for the development of intelligent terrain analysis and scene interpretation science.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study was supported in part by the National Natural Science Foundation of China [Grant No. 52275012, No. 51905119], Fundamental Research Funds for Central Universities [Grant No. HIT.NSRIF.2020090], and Self-Planned Task of State Key Laboratory of Robotics and System [Grant No. SKLRS202003C].

References

Bellutta, P., Manduchi, R., Matthies, L., Owens, K., Rankin, A., 2000. Terrain perception for DEMO III. Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat No 00TH8511): IEEE, pp. 326–331.

Brooks, C.A., Iagnemma, K., 2005. Vibration-based terrain classification for planetary exploration rovers. IEEE Trans. Rob. 21, 1185–1191.

Chavez-Garcia, R.O., Guzzi, J., Gambardella, L.M., Giusti, A., 2017. Image classification for ground traversability estimation in robotics. In: International Conference on Advanced Concepts for Intelligent Vision Systems. Springer, pp. 325–336.

Cheng, H., Chen, H., Liu, Y., 2014. Topological indoor localization and navigation for autonomous mobile robot. IEEE Trans. Autom. Sci. Eng. 12, 729–738.

Cowen, R., 2005. Opportunity rolls out of purgatory. Sci. News 167, 413.

Giguere, P., Dudek, G., 2009. Clustering sensor data for autonomous terrain identification using time-dependency. Auton. Robot. 26, 171–186.

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, p. 580–587.

Li, W., Hsu, C.-Y., 2020. Automated terrain feature identification from remote sensing imagery: a deep learning approach. Int. J. Geogr. Inf. Sci. 34, 637–660.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al., 2016. In: Ssd: Single shot multibox detector. European conference on computer vision. Springer, pp. 21–37.

Manduchi, R., Castano, A., Talukder, A., Matthies, L., 2005. Obstacle detection and terrain classification for autonomous off-road navigation. Auton. Robot. 18, 81–102.

Ojeda, L., Borenstein, J., Witus, G., Karlsen, R., 2006. Terrain characterization and classification with a mobile robot. J. Field Rob. 23, 103–122.

Rankin, A., Maimone, M., Biesiadecki, J., et al., 2020. Driving curiosity: Mars rover mobility trends during the first seven years. In: 2020 IEEE Aerospace Conference. IEEE, pp. 1–19.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Proces. Syst., 28

Sadhukhan, D., Moore, C., Collins, E., 2004. Terrain estimation using internal sensors. Proc. of the IASTED Int. Conf. on Robotics and Applications.

Valada, A., Spinello, L., Burgard, W., 2018. Deep feature learning for acoustics-based terrain classification. In: Robotics research. Springer, pp. 21–37.

Walas, K., Nowicki, M., 2014. Terrain classification using laser range finder. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 5003–5009.

Yu, C.W., Chen, Y.L., Lee, K.F. et al., 2019. Efficient intelligent automatic image annotation method based on machine learning techniques. In: 2019 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW). IEEE, pp. 1-2.

Zhang, S., Chi, C., Yao, Y., et al., 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9759-9768.

Zhou, X., Wang, D., Krähenbühl, P., 2019. Objects as points. arXiv preprint arXiv:190407850.