A Study on the Autonomous Detection of Impact Craters

Nour Aburaed[1] [0000-0002-5906-0249], Mina Alsaad[1] [0000-0003-0393-1700], Saeed Al Mansoori[2] [0000-0001-8499-2809], and Hussain Al-Ahmad[1] [0000-0002-5781-5476]

[1] University of Dubai, Dubai, UAE
[2] Mohammed Bin Rashid Space Centre, Dubai, UAE

Abstract. Planet surface studies are among the most popular research areas in planetary science, as they are useful to attain information about a planet's history and geology without directly landing on its surface. Autonomous detection of craters has been of particular interest lately, especially for Mars and lunar surfaces. This review study deals with the technical implementation, training, and testing of YOLOv5 and YOLOv6 to gauge their efficiency in detecting craters. YOLOv6 is the most recent member of the YOLO family, and it is believed that it outperforms all of its predecessors. In addition to comparing the aforementioned two models, the performance of the most widely used optimization functions, including SGD, Adam, and AdamW is studied as well. The methods are evaluated using mAP and mAR to verify whether YOLOv6 potentially outperforms YOLOv5, and whether AdamW is capable to generalize better than its peer optimizers.

Keywords: Object Detection, Convolutional Neural Networks, YOLO, Craters, optimizers

1 Introduction

Exploring the solar system has been a popular research interest since time immemorial. With the rapid and revolutionary improvement in technology that enables scientists to collect data and interact with outer space, the interest in this area of research continues to increase steadily throughout the years and across several industrial prospects. Mars has been the most studied planet after Earth due to their similar geological characteristics. Furthermore, there are already large archives of Martian numerical and visual datasets [5]. Mars has various surface features that are of particular importance, such as impact craters, valleys, and sand dunes. Impact craters (or simply craters) are considered as one of the most important features on Mars as well as the moon. They are a result of collision impacts, such as meteoroids, massive asteroids, and so on with Martian or lunar surfaces [25]. The interest in detecting craters rose especially after NASA launched and landed Mars Rover in 2020 to explore Mars surface, which uses autonomous navigation to cover larger distance. Craters present potential obstacles for autonomous navigation vehicles, and thus, scientists utilized image processing and Artificial Intelligence (AI) technology to investigate the size,

This is a peer-reviewed, accepted author manuscript of the following article: Aburaed, N., Alsaad, M., Mansoori, S. A., & Al-Ahmad, H. (2022). A study on the autonomous detection of impact craters. In N. El Gayar, E. Trentin, M. Ravanelli, & H. Abbas (Eds.), Artificial Neural Networks in Pattern Recognition - 10th IAPR TC3 Workshop, ANNPR 2022, Proceedings (pp. 181-194). (Lecture Notes in Computer Science; Vol. 13739). Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-3-031-20650-4_15

shape, and density of various craters on Mars and the moon. Additionally, studying craters helps in understanding the relative age, timescale, weathering, and geology of a planet's surface without the need to land on it [11].

The manual methods for detecting and counting craters from images rely heavily on human experts to annotate craters by visual interpretation of images, which is considered labor-intensive, time consuming, and subjective to human error. Consequently, autonomous and robust crater detection is very important to develop an automatic extraction approach for detecting and identifying crater features from images with minimal human interventions. Nowadays, AI, especially Deep Learning (DL) techniques, play a vital role in object detection generally, including the analysis of Martian and lunar surfaces [16]. Convolutional Neural Networks (CNNs) are a subset of Artificial Neural Networks (ANNs), which are in turn a subset of Machine Learning and AI. CNNs have proven their efficiency in various computer vision tasks, including segmentation [24], classification and object detection [1]. Hence, many CNN-based object detection methods are adopted to automatically extract the impact crater.

This paper presents an experimental review study on the autonomous detection of craters on Mars and lunar surfaces by comparing the performance of You Only Look Once (YOLO)-v5 and -v6 techniques. Specifically, the optimization technique is studied as a crucial hyperparameter in each network architecture. The studied optimization functions include Stochastic Gradient Descent (SGD), Adaptive Moment Estimation (Adam), and weight decay Adam (AdamW). There exists a consensus that YOLOv6 outperforms YOLOv5, and that AdamW has better ability to generalize than Adam and SGD. This consensus will be verified through this study. The efficiency is determined according to mean Average Precision (mAP) and mean Average Recall (mAR). The rest of the paper is organized as follows: Section 2 presents the necessary background related to object detection, YOLO, and the optimization functions, Section 3 showcases the dataset used in this study, Section 4 explains the experimental setup, Section 5 demonstrates the results and draws deductions based on them, finally, Section 6 concludes the paper by summarizing the findings and stating the future directions of this study.

2 Background

2.1 Object Detection

In computer vision and remote sensing, object detection is defined as the task of locating and identifying objects in an image or video by assigning the class to which this object belongs [4]. There exists a wide range of applications, such as traffic monitoring, autonomous vehicle navigation, and environmental monitoring [15]. However, object detection is considered as a challenging task due to the large variations in object scales and orientations, background complexity, and lack of labeled samples [19]. Object detection algorithms can be categorized into traditional approaches and Deep CNN (DCNN)-based approaches. Some of the most successful traditional methods for object detection include template matching [3], knowledge-based methods, Object-based Image Analysis (OBIA) [8], and Machine Learning (ML). The latter includes a feature extraction stage [6, 2, 23], followed by a classifier. The main drawbacks of the conventional approaches are their instability towards various object sizes, categories, and lighting conditions. It is also considered computationally complex. Thus, these approaches were diminished gradually and replaced by DCNNs techniques [9]. Nowadays, DCNN-based methods are state-of-the-art in computer vision and image processing tasks, particularly object detection problems due to the availability of robust labeled datasets and hardware with GPU and TPU processing capabilities to perform computation tasks in edge devices. In general, DCNNs-based object detection algorithms can be divided into two groups: one-stage and two-stage object detection approaches based on the model architecture. Two-stage algorithm often suffer from computational bottleneck that hinders real-time object detection. Thus, the models adopted for this research are one-stage object detectors, and the most prominent example of such type is the YOLO family of object detectors.

The first detector chosen from the YOLO family is YOLOv5. Recently, YOLOv5 gained a significant advantage in autonomous object detection task in terms of speed and detection accuracy [10]. This network is the latest official release of the YOLO model series and it is built on Pytorch rather than Darknet framework [12]. The basic architecture of YOLOv5 consists of three parts: Backbone, Neck, and Head, as shown in Figure 1 [13]. Focus structure and Cross Stage Partial Network (CSP) are introduced in the Backbone block. The focus layer was added to reduce layers, parameters, FLOPS, and CUDA memory, and to improve forward and backward speed. CSP Backbone reduces the repetitive gradient information during the optimization process, which improves the inference speed and accuracy and decreases the total number of parameters, which in turn reduces the model size. The Neck block in YOLOv5 contains Path Aggregation Network (PANet) and Spatial Pyramid Pooling (SPP) modules. PANet is used to obtain feature pyramids, which adopts a Feature Pyramid Network (FPN) that includes several bottom ups and top down layers to enhance the propagation of low level features in the model. PANet also improves the localization in lower layers, which also improves the localization accuracy of the object. SPP module is used to increase the receptive field of the network and capture different scale features. The Head of YOLOv5 is exactly the same as YOLOv4 and YOLOv3 Heads that generates three different sizes (18×18, 36×36, 72×72) of feature maps to achieve multi-scale prediction that enables the model to deal with small, medium, and big objects [22]. The major improvements during the training stage includes mosaic data augmentation and auto learning bounding box anchors. Also, Generalized Intersection Over Union (GIOU) loss function is introduced during training instead of the IOU loss adopted in YOLOv3 [21]. The overall architecture of YOLOv5 is shown in Figure 1.

The second autonomous detector chosen for this study is the most recent version of the YOLO family, which is YOLOv6, a single-stage object detection

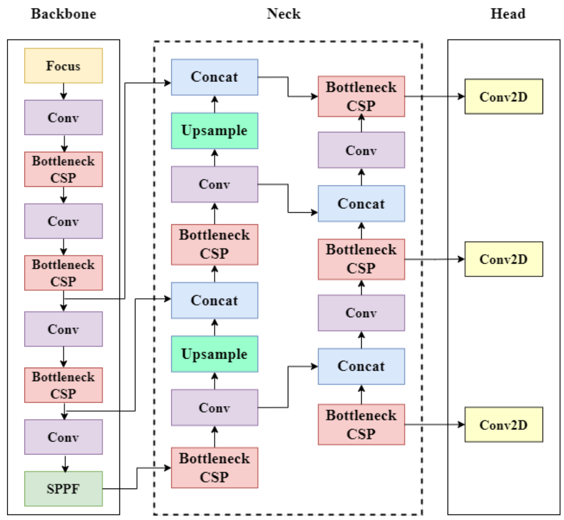


Fig. 1: Yolov5 Architecture

framework dedicated to industrial applications. This model is not part of the official YOLO series, but according to [20], it outperforms YOLOv5 in terms of detection accuracy and inference speed, which is more efficient for industrial applications. YOLOv6 includes many improvements in the Backbone, Neck, Head blocks and also training strategies. For instance, the Neck and Backbone in YOLOv6 have been redesigned by using Rep-PAN and EfficientRep structures, respectively, based on the idea of hardware-aware neural network. EfficientRep Backbone can make use of hardware computing power, such as GPU, and it also has strong representation capabilities. Rep-Pan Neck, is more accurate and faster then PANet and SPP. Yolov6 Head is decoupled by adding a layer between the network and the final Head, which in turn improves the performance. During the training process, RepVGG style [7] is introduced, which is a re-parameterizable structure that adopts a multi-branch topology and can be equivalently fused into a single 3×3 convolution. This fusion utilizes the processing power and memory. For efficient inferencing, the training strategy also includes an anchor-free paradigm, Simplified Optimal Transport Assignment (SimOTA) label assignment policy, and Scale-Sensitive IOU (SIOU) Bounding Box regression loss. The overall architecture of YOLOv6 is shown in Figure 2.

2.2 Optimization Functions

Optimization algorithms are used to minimize the loss of predictive models, such as DCNNs, with respect to a certain training dataset by adjusting the model's parameters, such as the weight and/or learning rate. Detection of craters can be considered as an object detection problem that consists of only one class

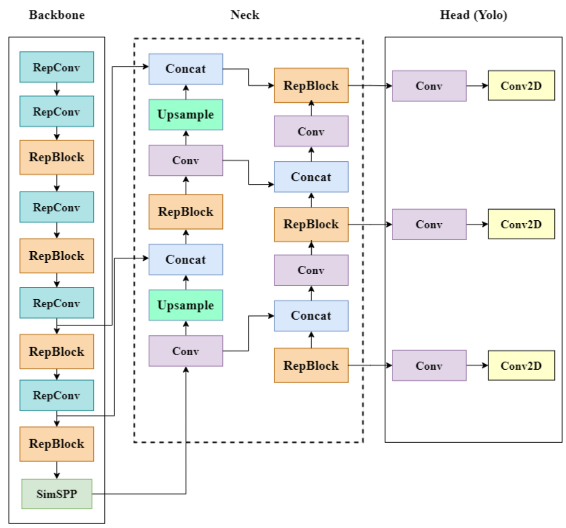


Fig. 2: Yolov6 Architecture

of objects. For an ith sample of a crater object denoted x with label y, SGD optimizer is defined as:

$$\Theta = \Theta - \alpha \cdot \nabla_{\Theta} J(\Theta; x^{(i)}; y^{(i)}) \tag{1}$$

such that Θ represents the model's parameters, $\nabla_{\Theta} J$ is the objective function, and α is the learning rate. SGD's advantage is that it is generally fast, and its fluctuations enable it to potentially escape from one local minimum to a better one, but it makes finding the global minimum more complicated.

Adam is another optimization algorithm that was first proposed in [14]. It divides the learning rate by an exponentially decaying average of squared gradients, which is an idea that was first seen in earlier optimization functions, such as RMSProp and Adadelta. Additionally, Adam uses a mechanism known as *momentum*, such that it keeps an exponentially decaying average of past gradients. This gives Adam greater advantage in terms of adaptive learning. The decaying averages at a certain time step t are computed as follows:

$$\hat{m}_t = \frac{\beta_1 m_{t-1} + (1 - \beta_1) g_t}{i - \beta_1^t} \tag{2}$$

$$\hat{v}_t = \frac{\beta_2 v_{t-1} + (1 - \beta_2) g_t^2}{1 - \beta_2^t} \tag{3}$$

g_t is the gradient at time t, and m_t and v_t are the estimates of mean and uncentered variance, respectively. Consequently, Adam updates the DCNN's parameters as follows:

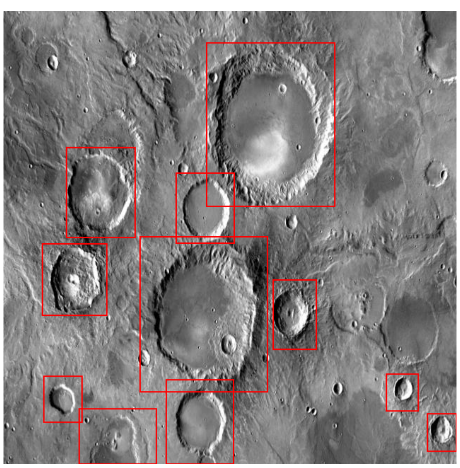
$$\Theta_{t+1} = \Theta_t - \frac{\alpha}{\sqrt{\hat{v}} + \epsilon} \hat{m}_t \quad (4)$$

The most recent version of Adam is AdamW [18]. Loshchilov and Hutter argue that Adam does not have the ability to generalize, and they solve this issue by controlling the weight decay, such that it is performed after controlling the parameter-wise step size. AdamW is expressed as follows:

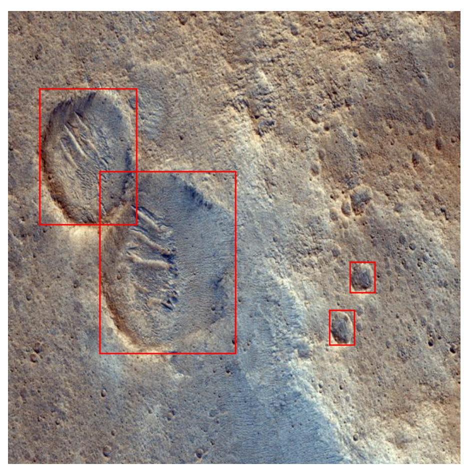
$$\Theta_{t+1} = \Theta_t - \alpha \frac{\beta_1 - m_t + (1 - \beta_1)(g_t + w\Theta_t)}{\sqrt{\hat{v}} + \epsilon} \quad (5)$$

3 Dataset

The dataset used in this study is the Martian/Lunar Crater Detection Dataset provided by Kaggle [17]. The dataset mainly contains images of Mars and moon surfaces, which may contain crater features. The data is from different sources. For Mars, the images are mainly from Arizona State University (ASU) and United States Geological Survey (USGS), while all moon images are from NASA Lunar Reconnaissance Orbiter mission. All images are pre-processed with RoboFlow to remove EXIF rotation, and they are resized to 640×640. The annotation work was performed by a group from Tongji University, Shanghai, China in YOLO text format. Samples from the dataset are shown in Figure 3.



(a) Groundtruth for lunar image



(b) Groundtruth for Mars image

Fig. 3: Sample from mars/lunar crater dataset

Table 1: Training parameters for YOLOv5 and YOLOv6 experiments.

Parameter	Value
Epochs	500
Loss	MSE
Optimization	SGD/Adam/AdamW
Scheduler	Cosine annealing

4 Experimental Setup

The dataset discussed in Section 3 is divided into 98 images for training, 26 for validation, and 19 for testing. YOLOv5 is tested 3 times, each time with a different optimization function; SGD, Adam, and AdamW. The same experiments are repeated for YOLOv6. In order to ensure fair comparison, the same training parameters are applied to both YOLOv5 and YOLOv6 across all experiments. Furthermore, an early stopping strategy is deployed, such that if the network does not improve for 100 consecutive epochs, then it stops training prematurely. Also, the evaluation is performed based on the best parameters obtained, and not based on the most recent epoch. The training parameters are summarized in Table 1.

5 Results

Whether a detection is considered as True Positive (TP), True Negative (TN), False Positive (FP), or False Negative (FN), this is decided depending on the Intersection Over Union (IoU), which assesses the extent of overlap between the detected bounding box and the groundtruth bounding box. IoU is defined as follows:

$$IoU = \frac{Area\ of\ overlap}{Area\ of\ union} \tag{6}$$

For this study, the threshold value of IoU is 0.5, such that if the detected bounding box overlaps with the groundtruth one by at least 50%, then the detection is considered as TP. Detections with less than 50%, where 0% indicates failure to detect the object altogether, are depicted as FN. If the model predicts a crater where no crater exists, the detection is regarded as FP, and if it successfully avoids this prediction, then it is a TN case.

The results of the experiments are evaluated using mAP and mAR. Precision indicates the model's ability to find TP among all TP and FP cases. mAP measures precision across all classes c. Since there is only one class in this study, mAP and precision are calculated the same way and can be used interchangeably, such that:

$$mAP = P = \frac{1}{c}\sum_{1}^{c}\frac{TP}{TP+FP} \tag{7}$$

Recall indicates the model's ability to find TP among all TP and FN cases. Similar to mAP, mAR measures recall across all classes, such that

$$mAR = R = \frac{1}{c}\sum_{1}^{c}\frac{TP}{TP+FN} \tag{8}$$

Table 2 summarizes the results of the experiments in terms of mAP and mAR. Overall, YOLOv5 shows better performance than YOLOv6 in terms of mAP. At the same time, YOLOv6 shows better performance than YOLOv5 in terms of mAR. AdamW shows better performance compared to other optimization algorithms in both YOLOv5 and YOLOv6 for mAP, but not for mAR. AdamW shows the best mAP with YOLOv5, while SGD shows the best mAR for the same network. For YOLOv6, Adam shows the best mAP and mAR. It is worth mentioning that there are cases where YOLOv6 with AdamW successfully detects craters in complicated scenes, whereas it fails to do so with other optimization functions, and YOLOv5 generally fails to detect them as well. Figure 4 depicts such case, where Figures 4(a-e) all show FN, FP, and redundant detections, which are also regarded as FP. On the other hand, Figure 4(f), does not show such cases. Conversely, there are cases where YOLOv6 with SGD shows the best result, as seen in Figure 5.

It seems that despite the strengths of each model and optimization function, there is still no optimal way to avoid all the shortcomings and cover all the scenarios, as each model and each optimization function are successful in specific cases. This opens room for improvement and draws a new direction in this area of research. Furthermore, YOLOv6 did not outperform YOLOv5 in all scenarios, and AdamW did not outperform Adam and SGD in all scenarios, which contradicts the general belief about YOLOv6 and AdamW.

6 Conclusion and Future Work

In this study, an experimental review of YOLOv5 and YOLOv6 models with SGD, Adam, and AdamW optimization functions has been conducted using Martian/Lunar Crater Detection Dataset. The general belief is that YOLOv6 is an upgraded version of YOLOv5, and that AdamW optimizer outperforms SGD and Adam. The networks and optimization functions are tested systematically to verify these claims. Experiments show inconsistent outcomes, as there are cases where YOLOv5 performs better than YOLOv6, and vice versa, and cases where each optimizaion function outshines others under different conditions. In the future, YOLOv6 model will be extended to the wavelet domain in order to

Table 2: Results Summary

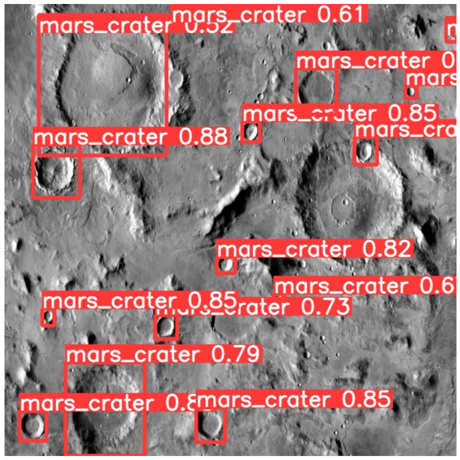
Model	SGD		Adam		AdamW	
	mAP	mAR	mAP	mAR	mAP	mAR
YOLOv5	**0.72**	0.54	**0.73**	0.48	**0.74**	0.49
YOLOv6	0.62	**0.64**	0.62	**0.65**	0.55	**0.47**

detect objects using frequency analysis. This can potentially mitigate the aforementioned inconsistencies and shortcomings faced in YOLOv6's performance. Also, the model will be trained and tested using Mars data collected by Emirates Hope Probe.

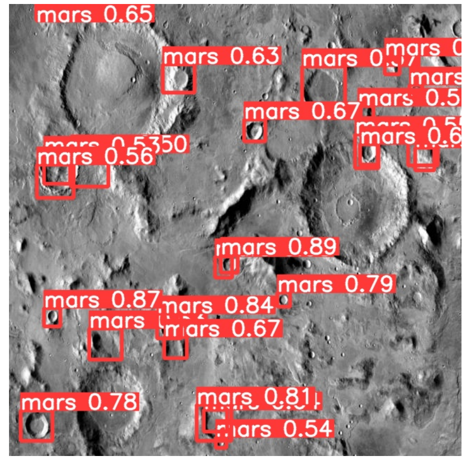
References

1. Aburaed, N., Al-Saad, M., Chendeb El Rai, M., Al Mansoori, S., Al-Ahmad, H., Marshall, S.: Autonomous object detection in satellite images using wfrcnn. In: 2020 IEEE India Geoscience and Remote Sensing Symposium (InGARSS). pp. 106–109 (2020). https://doi.org/10.1109/InGARSS48198.2020.9358948
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer vision and image understanding **110**(3), 346–359 (2008)
3. Chantara, W., Ho, Y.S.: Object detection based on fast template matching through adaptive partition search. In: 2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE). pp. 1–6. IEEE (2015)
4. Cheng, G., Han, J.: A survey on object detection in optical remote sensing images. ISPRS Journal of Photogrammetry and Remote sensing **117**, 11–28 (2016)
5. Christoff, N., Manolova, A., Jorda, L., Mari, J.L.: Morphological crater classification via convolutional neural network with application on mola data. In: ANNA '18; Advances in Neural Networks and Applications 2018. pp. 1–5 (2018)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). vol. 1, pp. 886–893. Ieee (2005)
7. Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13733–13742 (2021)
8. Fatima, S.A., Kumar, A., Pratap, A., Raoof, S.S.: Object recognition and detection in remote sensing images: a comparative study. In: 2020 International Conference on Artificial Intelligence and Signal Processing (AISP). pp. 1–5. IEEE (2020)
9. Guo, J.M., Yang, J.S., Seshathiri, S., Wu, H.W.: A light-weight cnn for object detection with sparse model and knowledge distillation. Electronics **11**(4), 575 (2022)
10. Guo, Z., Wang, C., Yang, G., Huang, Z., Li, G.: Msft-yolo: Improved yolov5 based on transformer for detecting defects of steel surface. Sensors **22**(9), 3467 (2022)

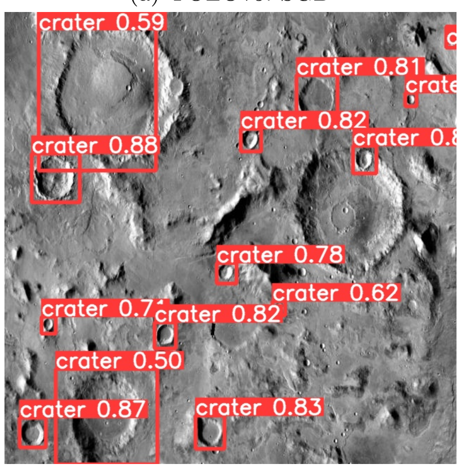
11. Hsu, C.Y., Li, W., Wang, S.: Knowledge-driven geoai: Integrating spatial knowledge into multi-scale deep learning for mars crater detection. Remote Sensing **13**(11) (2021). https://doi.org/10.3390/rs13112116, https://www.mdpi.com/2072-4292/13/11/2116
12. Jindal, M., Raj, N., Saranya, P., Sundarabalan, V.: Aircraft detection from remote sensing images using yolov5 architecture. In: 2022 6th International Conference on Devices, Circuits and Systems (ICDCS). pp. 332–336. IEEE (2022)
13. Jung, H.K., Choi, G.S.: Improved yolov5: Efficient object detection using drone images under various conditions. Applied Sciences **12**(14), 7255 (2022)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015), http://arxiv.org/abs/1412.6980
15. Li, K., Cao, L.: A review of object detection techniques. In: 2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT). pp. 385–390. IEEE (2020)
16. Lin, X., Zhu, Z., Yu, X., Ji, X., Luo, T., Xi, X., Zhu, M., Liang, Y.: Lunar crater detection on digital elevation model: A complete workflow using deep learning and its application. Remote Sensing **14**(3) (2022). https://doi.org/10.3390/rs14030621, https://www.mdpi.com/2072-4292/14/3/621
17. LincolnZh: Martian/lunar crater detection dataset (Feb 2022), https://www.kaggle.com/datasets/lincolnzh/martianlunar-crater-detection-dataset
18. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=Bkg6RiCqY7
19. Mahmoud, A., Mohamed, S., El-Khoribi, R., Abdelsalam, H.: Object detection using adaptive mask rcnn in optical remote sensing images. Int. J. Intell. Eng. Syst **13**(1), 65–76 (2020)
20. meituan: Yolov6 (June 2022), https://github.com/meituan/YOLOv6
21. Miao, J., Zhao, G., Gao, Y., Wen, Y.: Fire detection algorithm based on improved yolov5. In: 2021 International Conference on Control, Automation and Information Sciences (ICCAIS). pp. 776–781. IEEE (2021)
22. Nepal, U., Eslamiat, H.: Comparing yolov3, yolov4 and yolov5 for autonomous landing spot detection in faulty uavs. Sensors **22**(2), 464 (2022)
23. Norman, M., Shahar, H.M., Mohamad, Z., Rahim, A., Mohd, F.A., Shafri, H.Z.M.: Urban building detection using object-based image analysis (obia) and machine learning (ml) algorithms. In: IOP Conference Series: Earth and Environmental Science. vol. 620, p. 012010. IOP Publishing (2021)
24. Talal, M., Panthakkan, A., Mukhtar, H., Mansoor, W., Almansoori, S., Ahmad, H.A.: Detection of water-bodies using semantic segmentation. In: 2018 International Conference on Signal Processing and Information Security (ICSPIS). pp. 1–4 (2018). https://doi.org/10.1109/CSPIS.2018.8642743
25. Wang, J., Ding, W., Fradkin, B., Pham, C.H., Sherman, P., Tran, B.D., Wang, D., Yang, Y., Stepinski, T.F.: Effective classification for crater detection: A case study on mars. In: 9th IEEE International Conference on Cognitive Informatics (ICCI'10). pp. 688–695 (2010). https://doi.org/10.1109/COGINF.2010.5599824



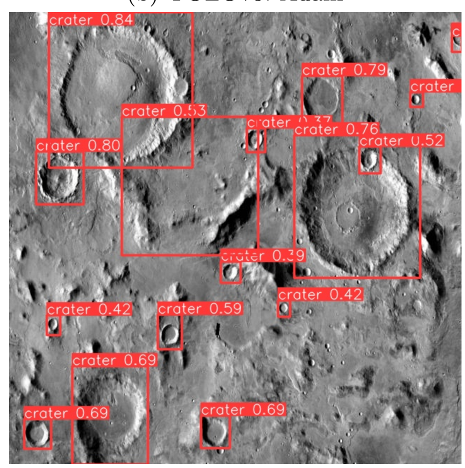
(a) YOLOv5: SGD



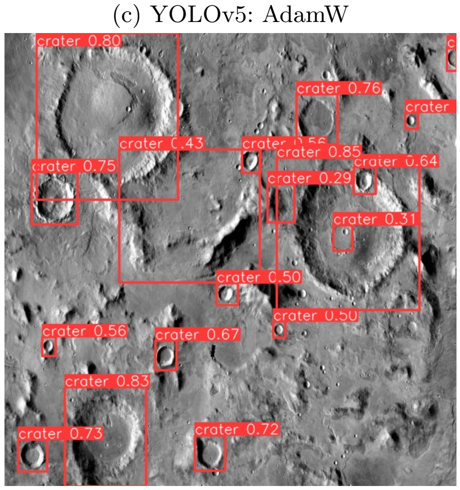
(b) YOLOv5: Adam



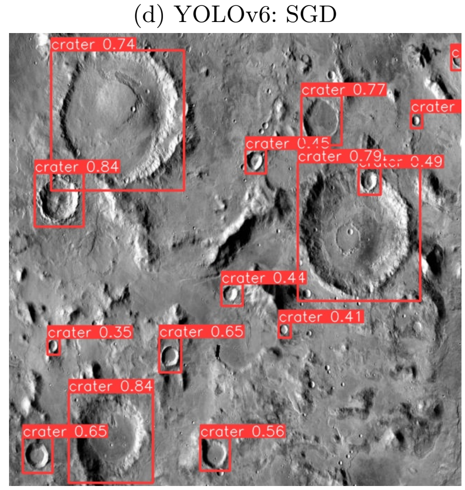
(c) YOLOv5: AdamW



(d) YOLOv6: SGD

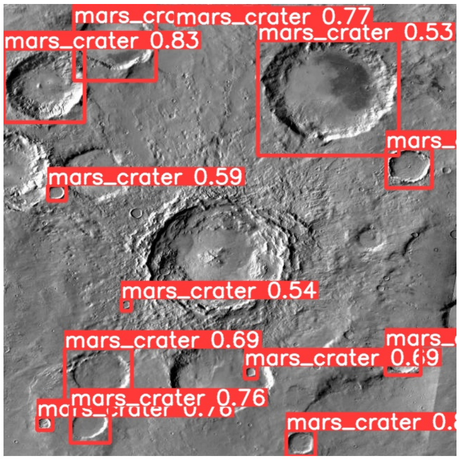


(e) YOLOv6: Adam

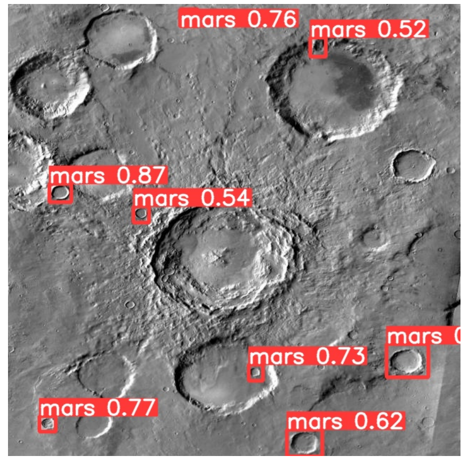


(f) YOLOv6: AdamW

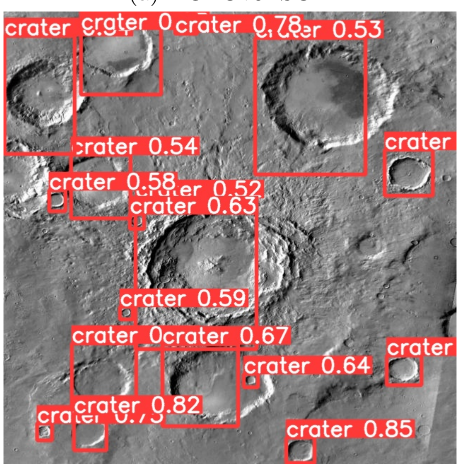
Fig. 4: Sample results from YOLOv5 with (a) SGD, (b) Adam, and (c) Adam w, and sample results from YOLOv6 with (d) SGS, (e) Adam, (f) AdamW. YOLOv6 with AdamW shows the best results.



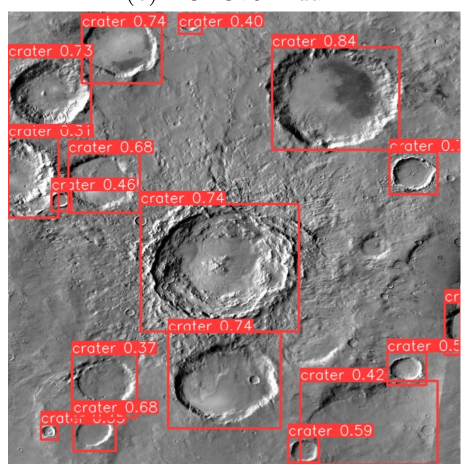
(a) YOLOv5: SGD



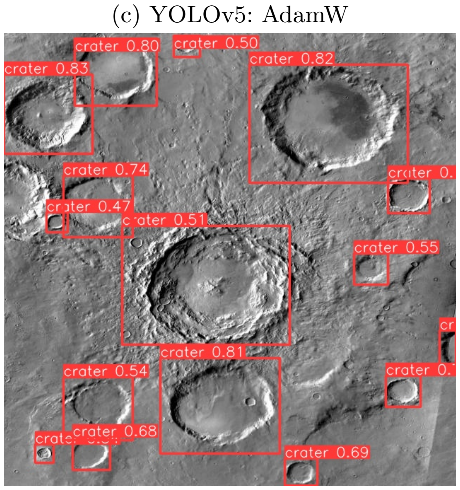
(b) YOLOv5: Adam



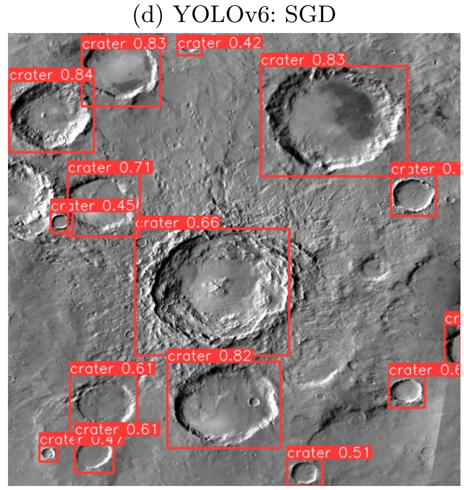
(c) YOLOv5: AdamW



(d) YOLOv6: SGD



(e) YOLOv6: Adam



(f) YOLOv6: AdamW

Fig. 5: Sample results from YOLOv5 with (a) SGD, (b) Adam, and (c) Adam w, and sample results from YOLOv6 with (d) SGS, (e) Adam, (f) AdamW. YOLOv6 with AdamW shows the best results.