Chinese Industry and Technology of Eco-environment
2025, VOL. 6, NO. 5, 21-28
DOI: 10.47297/wspciateWSP2633-772X05.20250605

Exploring the Application of Earth-Based Crater Remote Sensing Recognition Methods to the Martian Surface

ZiShuo Yuan

Shengli No.1 Middle School of Dongying, Dongying, Shandong, 257000, China

ABSTRACT
This study investigates the application of Earth-based crater remote sensing recognition methods to Martian surface imagery. To address the transferability limitations of traditional image processing techniques in planetary crater detection, we designed a workflow that integrates multi-scale Hough transform with physical indicators such as ring completeness and radial gradient. We verify the effectiveness of the approach through experiments on Earth remote sensing imagery: the method reliably detects primary craters with small radius estimation errors, while candidate generation is identified as the major bottleneck due to its high computational cost. The analysis further reveals that detection performance is strongly influenced by illumination conditions and edge sharpness, with limited sensitivity to low-contrast crater rims. Finally, we discuss the potential value of incorporating geophysical indicators and multi-source data in future Martian crater recognition. Overall, this lightweight, training-free method proves feasible in resource-constrained scenarios and offers a valuable reference for visual navigation in planetary exploration.

KEYWORDS
Crater detection; Remote sensing imagery; Mars; Ring completeness; Visual navigation; Hough transform

1 Introduction

Imagine a Mars lander orbiting over the red desert, attempting to find a safe landing site amid dust storms and complex shadows. Its "eyes"—the vision-based navigation system—must detect surface craters and complete precise localization within just 60 seconds. Accurate crater recognition not only provides navigation landmarks for landers, but also plays a vital role in interpreting the evolutionary history of the solar system, understanding planetary geological processes, and even searching for potential traces of life.

As deep space missions move toward higher autonomy, enabling spacecraft to "understand" unfamiliar environments and achieve real-time perception, precision landing, and autonomous navigation has become a critical technical challenge. Visual perception is the foundation of this capability, and its reliability is essential. While deep learning has shown remarkable performance in visual tasks, its dependence on large labeled datasets and limited robustness under extreme environments remain major obstacles. For deep space probes, constraints on onboard computing power and storage further restrict the deployment of large-scale neural networks, especially in "cold-start" and edge-computing scenarios. In contrast, traditional geometry-based methods are lightweight, but they often neglect geological context, leading to false detections of volcanoes, valleys, and other circular structures as craters.

This dilemma highlights the urgent need for lightweight, interpretable, and training-free alternatives. In this work, we propose a lightweight crater detection pipeline based on traditional image processing (OpenCV): adaptive Hough-based candidate generation with pyramid scaling to enhance recall, Scharr+Canny filtering for robust edge extraction, ring completeness and radial gradient consistency evaluation in polar coordinates, and efficient refinement with NMS for stable detections. The pipeline is first validated on Earth remote sensing imagery, and then transferred to Martian surface images to systematically evaluate cross-domain adaptability, environmental robustness, and failure cases.

The main contributions of this study are as follows:

(1) A minimal, training-free pipeline with full implementation details;

(2) An adaptive candidate generation and scoring scheme balancing recall and efficiency;

(3) Empirical cross-planet experiments with error analysis, providing a baseline and engineering reference for future integration with deep learning approaches.

2 Related Work

2.1 Physical Characteristics of Craters in Remote Sensing Imagery

Craters are formed by high-velocity impacts, typically exhibiting a bowl-shaped depression with uplifted rims often accompanied by ejecta deposits. With increasing diameter, features such as central peaks, multi-ring basins, and terraced faults appear, reflecting rebound and collapse of subsurface materials. Impact melt commonly accumulates at the crater floor or flows outward, forming smooth, low-albedo dark bands. Fresh craters may show radial ejecta rays, while

* **Corresponding Author:** ZiShuo Yuan, yuanzishuo2006@gmail.com

surrounding strata can undergo fractures and collapse.

In remote sensing imagery, craters are characterized by near-circular outlines and "low-interior, high-rim" geometry. Under low solar elevation, crater walls produce strong shadow–highlight contrasts, with grayscale gradients from dark to bright along the illumination direction. Visible imagery emphasizes shadow–light relations, multispectral data highlight ejecta rays and concentric ring structures, thermal infrared captures diurnal temperature differences (especially pronounced in young craters), and radar imaging reveals surface roughness, with melt flows often showing high polarization ratios. These morphological and multispectral features provide physical cues for automatic detection: circular rims, shadow–bright contrasts, concentric structures, radial rays, and dark deposits at the floor are all critical indicators.

2.2 Irreplaceable Role of Craters as Navigation Landmarks

Among various visual landmarks for navigation, craters are optimal due to their morphological stability, ubiquitous distribution (e.g., on Mars), and data availability. Their core advantages include:

Absolute localization without initial pose, avoiding reliance on prior attitude information;

Compatibility with passive imaging, suitable for low-power, lightweight payloads without active light sources;

Weak model dependence, as opposed to 3D terrain template matching, which requires large DEMs, thus reducing storage and computational demand.

2.3 Evolution of Autonomous Navigation in Deep Space Exploration

Deep space missions generally involve three stages: orbit, entry, and landing. Orbit phase relies on star trackers and stellar catalog matching for pose estimation; entry phase mainly depends on the Inertial Measurement Unit (IMU), but errors accumulate rapidly; landing phase requires vision-based navigation to correct trajectories, becoming the key to achieving precision landing at the 100-meter scale.

2.4 Triple Challenges of Crater-based Navigation

Environmental robustness: erosion, illumination variation, and atmospheric disturbances blur crater rims, leading to detection failures.

Real-time constraints: with landing windows as short as 60 seconds, image processing and multi-sensor fusion must be executed within seconds.

Error accumulation: detection noise, crater map inaccuracies (position/radius bias), and IMU drift amplify each other, severely degrading navigation accuracy.

2.5 Technical Chain: Detection–Recognition–Pose Estimation

Detection:

Unsupervised: morphology-based filtering, Hough transform, and edge enhancement; effective in data-scarce settings.

Supervised: deep learning frameworks (Faster R-CNN, YOLO) to improve detection in complex scenes.

Hybrid: combining geometric priors with photometric consistency for improved robustness under shadowing and occlusion.

Recognition:

Tracking-based (with initial pose): optical flow and feature matching for temporal association.

Lost-in-space (LIS) recognition (without initial pose): spatial topology of craters matched with catalog databases.

Pose estimation:

Pure image geometry (PnP, multi-view): efficient but sensitive to mismatches.

Dynamic model fusion (e.g., EKF with orbital dynamics): suppresses jitter and drift in single-frame solutions.

2.6 Future Research Directions

To overcome current bottlenecks, future work should focus on:

Domain-adaptive detection frameworks, constructing illumination-invariant and erosion-resistant crater representations to enable zero-shot transfer from Earth to Mars;

Embedded real-time architectures, using FPGA/DSP acceleration for lightweight pipelines that meet the 60-second landing window;

Multi-modal tightly-coupled navigation, integrating vision, IMU, and LiDAR with joint error suppression to break the sensor error chain.

3 Methods

3.1 Overall Approach

This study proposes a lightweight, training-free crater detection method for Mars. The core ideas are:

(1) Apply mild grayscale enhancement to remote sensing images;

(2) Generate circular candidates using adaptive multi-scale Hough circle detection;

(3) Compute "rim evidence" based on ring sampling and gradient consistency;
(4) Refine the candidate center/radius and perform ellipse fitting correction;
(5) Rank with a physics-inspired interpretable scoring function, and apply NMS to output the Top-K results.

3.2 Preprocessing

The input RGB image is first resized so that the longest side ⩽ 800 pixels to control runtime. Preprocessing includes:
Grayscale conversion and background suppression:

$$g_0 \rightarrow g = g_0 - GaussianBlur\left(g_0, \sigma = 25\right)$$

Linear normalization to [0,255]
Adaptive histogram equalization (CLAHE, 8×8, clip=1.6)
Mild gamma/bilateral filtering and unsharp masking to enhance edges while suppressing noise
Optional illumination alignment: perform angular brightness fitting on the ring band; if a dominant illumination direction is detected, apply a small-angle rotation to align shadows (failure does not affect subsequent steps)

3.3 Candidate Generation

At three scales $\boldsymbol{s} \in \{0.8, 1.0, 1.2\}$, Hough circle detection (cv2.HoughCircles) is applied with adaptive parameters:
dp: resolution factor (1.0 for small images, 1.2 for large images)
param2: threshold adapted to image contrast (lower contrast → more relaxed)
Radius range: $[0.06, 0.60] \times min(H, W)$
Minimum center distance: $d_{\{\backslash min\}} = 0.16 \times min(H,,W)$

After mapping candidates back to the original scale, Non-Maximum Suppression (NMS) with triple constraints (center distance, radius difference, IoU) is applied to form the global candidate set (⩽80).

3.4 Rim Evidence and Metrics

For each candidate circle(x,y), a ring band is constructed and angular sampling is performed to derive four interpretable metrics:
Completeness (comp)

$$comp = \left(\frac{1}{A}\right) \Sigma_{a=1}^{A} 1\left\{ \exists p \in ring(a),\ DT(p) \le \tau \right\}$$

where A is the number of angular samples and $\boldsymbol{\tau}$ is the band tolerance (default 7 px, adaptive to r.
Sector Uniformity
Divide 360° into 12 sectors and check coverage; ⩾4 valid sectors indicate relative uniformity.
Radial Gradient Consistency(ev)

$$ev = max\left(ev_{plain}, ev_{orient}\right),$$

$$ev\_orient = E[\ 1\{\ DT \le \tau,\ |cos\angle(\nabla I, n_r)| \ge cos30^\circ\ \}$$

Where n_r is the outward normal.
Rim Contrast (signed z-score)
Compute clipped mean/std in inner/outer rings:

$$z = \frac{\mu_{out} - \mu_{in}}{\sqrt{\left(\sigma_{in}^2 + \sigma_{out}^2\right)/2}}$$

Prior condition: $\mu_{in} < \mu_{out}$(inside darker, outside brighter).

3.5 Refinement and Geometric Correction

Center refinement: search within $(x \pm 8, y \pm 8)$ neighborhood to maximize comp.
Radius refinement: search within $[r(1 \pm 0.10)]$ range at 2 px step.
Ellipse correction:sample edge points, fit an ellipse (a,b,θ);if eccentricity
$e = \sqrt{1 - b^2/a^2} \le 0.6$and center shift is small, replace with equivalent circle

$$r = \sqrt{ab}.$$

3.6 Scoring and NMS

Final score:

$$\boxed{score = 0.40|z| + 0.30\ ev + 0.20\ comp + 0.10\ \frac{rg}{40}}$$

where $rg = E\left(\left|\nabla I \cdot n_r\right|\right)$ is the mean radial gradient strength (normalized).

NMS with center distance + radius difference + IoU is applied for redundancy removal. If no candidate remains, a fallback rule preserves one reliable target.

Algorithm 1 Lightweight crater detection (training-free)

Inputs:

I # input image

SCALES = {0.8, 1.0, 1.2}

MAX_SIDE = 800

HOUGH.dp_small = 1.0; HOUGH.dp_large = 1.2

R_FRAC = [0.06, 0.60]; MIN_DIST_FRAC = 0.16

RING_TOL = 7 px; REFINE_RADIUS_FRAC = ±10%; REFINE_STEP = 2 px

NMS.center_frac = 0.28; NMS.radius_rel = 0.20; NMS.iou = 0.35

TOPK = 3; ALLOW_FALLBACK = True; AUTO_ALIGN_SUN = True

Output:

dets # final Top-K detections: (x, y, r, score)

1: I ← resize_longest(I, MAX_SIDE)

2: G ← enhance_gray_mild(I) # §Methods: Preprocessing

3: if AUTO_ALIGN_SUN then

4: G ← optional_rotate_by_estimated_sun(G) # continue even if failed

----- Candidate generation (multi-scale Hough) -----

5: raw_cands ← ∅

6: for s in SCALES do

7: Gs ← resize(G, scale = s)

8: (Hs, Ws) ← shape(Gs)

9: dp ← (min(Hs, Ws) > 500) ? HOUGH.dp_large : HOUGH.dp_small

10: p2_list ← adaptive_param2(std(Gs)) # lower contrast → looser p2

11: for p2 in p2_list do

12: Cs ← HoughCircles(Gs, dp,

minRadius = R_FRAC.min*min(Hs,Ws),

maxRadius = R_FRAC.max*min(Hs,Ws),

minDist = MIN_DIST_FRAC*min(Hs,Ws),

param1 = fixed_high_canny,

param2 = p2)

13: raw_cands ← raw_cands ∪ back_project(Cs, scale = s)

14: end for

15: end for

16: raw_cands ← cap(raw_cands, max_total = 80)

17: cand_list ← NMS_3D(raw_cands, NMS) # center + radius + IoU

----- Refinement, metrics, scoring -----

18: dets ← ∅

19: for (x, y, r) in cand_list do

20: (x, y) ← refine_center_by_ring(G, x, y, r, window = ±8)

21: r ← refine_radius_by_completeness(G, x, y, r,

range = REFINE_RADIUS_FRAC,

step = REFINE_STEP)

22: (comp, ev_plain, ev_orient, rg, good_secs, z, mu_in, mu_out)

← ring_metrics(G, x, y, r, RING_TOL)

23: ev ← max(ev_plain, ev_orient)

24: if (|z| < 0.15 and ev < 0.20) then continue # weak evidence

25: if (mu_in ⩾ mu_out) then continue # prior: inner darker

26: if (good_secs < 4 and ev < 0.22) then continue # sector uniformity

27: (x, y, r) ← optional_ellipse_correction(G, x, y, r) # use if e ⩽ 0.6

28: score ← 0.40*|z| + 0.30*ev + 0.20*comp + 0.10*(rg/40)

29: push dets, (x, y, r, score)

30: end for

31: dets ← NMS_and_topK(dets, NMS, TOPK)

----- Conservative fallback when empty -----

32: if empty(dets) and ALLOW_FALLBACK then
33: dets ← conservative_fallback(raw_cands) # border-safe + min ev/comp gates
34: return dets
Parameter Settings (Default Values in This Study)
Operating mode: PROFILE="recall" (high-recall mode, for demonstration purposes);
Pyramid scales: {0.8, 1.0, 1.2}, candidate upper bound = 50;
Pyramid scales: $[0.06,0.60]\cdot min(H,W)$, minimum center distance: $0.16\cdot min(H,W)$
Ring tolerance: 7 px (adaptively adjusted with radius);
Refinement range: radius ±10%, step size = 2 px;
NMS thresholds: center fraction = 0.28, relative radius difference = 0.20, IoU = 0.35;
Output size: Top-3.
Implementation Details and Complexity
Implementation: Built on OpenCV (Canny, Scharr, Hough, warpPolar, fitEllipse, etc.), the pure Python implementation can process medium-resolution images in real time on a standard laptop.

Time complexity: The main cost lies in multi-scale Hough detection and ring resampling, with overall complexity $O(S\cdot HW)+O(M\cdot A\cdot B)$, where S is the number of scales, M is the number of candidates, A is the number of angular samples, and B is the ring width (in pixels).

Memory: Only a few intermediate maps (e. g., gradient, distance transform) are stored, keeping memory usage controlled.

Advantages and Limitations
Advantages:
No need for training or labeled data, with high interpretability;
Combination of multi-scale detection and ring evidence ensures good recall for clear circular rims;
Ellipse correction adapts to mild deformations and projection effects.
Limitations
Limited recall for small-scale or low-contrast craters;
Possible false detections near image borders or in strong shadow regions;
Dependent on edge quality, thus sensitive to noise.
Future work may incorporate DEMs or lightweight CNNs for secondary validation to enhance accuracy and robustness.
Experimental Setup
Hardware
Device: Apple MacBook Pro (Apple M4 Pro chip, 8 performance cores + 4 efficiency cores, 16-core GPU, 16 GB RAM)
GPU: This method does not rely on GPU acceleration; all computation is performed on a single CPU thread.
Software
Operating System: macOS Sequoia 15.6.1 (Apple Silicon)
Python Version: 3.13.2
OpenCV Version: 4.12.0
Running Parameters
Unless otherwise specified, all experiments were conducted with the following fixed parameters:
Image preprocessing: longest side scaled to MAX_SIDE = 800
Scale pyramid: [0.8, 1.0, 1.2]
Candidate generation: HoughCircles with adaptive dp/param2, maximum of 80 candidates
Physical constraints:
Radius range: 0.06 × short side – 0.60 × short side
Minimum center distance: 0.16 × short side
Refinement strategy
Center: ±8 pixel neighborhood search
Radius: ±10% range, step size 2 px
Physical metrics: ring width 7 px, edge-vote threshold 0.18, ellipse correction enabled if eccentricity ⩽ 0.6
Scoring function

$$score = 0.40|z| + 0.30\,ev + 0.20\,comp + 0.10\frac{rg}{40}$$

Post-processing: 3D NMS (center/radius/IoU) + Top-K = 3
Fallback: if no candidate survives, apply a conservative fallback rule (ev ⩾ 0.10, comp ⩾ 0.03, excluding boundary cases).

4 Metrics

4.1 Detection Metrics

k: Number of detected candidates (after de-duplication, Top-K ⩽ 3).

r [px]: Radius of the Top-1 detected crater in pixels, for comparison with manual observations.

score: Composite score reflecting the reliability of the detection result.

|z|: Ring contrast significance (strength of the "darker inside, brighter outside" prior).

ev: Edge votes, i.e., the proportion of ring samples supported by edges.

comp: Ring completeness (0–1, with values closer to 1 indicating a more complete rim).

rg: Mean radial gradient, representing the sharpness of crater edges.

4.2 Efficiency Metrics

I/O time: Time for image loading and resizing (unit: ms).

preprocess: Preprocessing time, including gray enhancement, filtering, edge detection, and distance transform.

candidates: Candidate generation time, including Hough circle detection, multi-scale pyramid resizing, and NMS de-duplication. This stage was identified as the primary computational bottleneck.

refine+score: Time for candidate refinement and physical measurements (including center/radius adjustment, ring completeness, contrast |z|, edge votes, etc.).

Total: Overall runtime, defined as $Total = I/O + preprocess + candidates + refine + score$

Table 1

Fig.N	File Name	K	IO	Preprocess (ms)	Candidates (ms)	Refine+Score(ms)	Total(ms)
1	1.jpg	1	20.2	27.5	4195.9	13.2	4256.8
2	2.jpg	1	17.9	26.0	743.5	6.7	794.1
3	3.jpg	1	13.9	26.5	7221.8	17.1	7278.5
4	4.jpg	1	4.9	28.1	5140.5	9.0	5182.5
5	5.jpg	1	13.2	29.7	690.5	18.2	756.6

Table 2

Fig.N	File Name	r (px)	Score	|z|	ev	comp	rg
1	1.jpg	130	1.651	0.09	1.00	1.00	446.3
2	2.jpg	279	0.872	0.10	0.67	0.67	199.0
3	3.jpg	185	0.490	0.00	1.00	1.00	345.2
4	4.jpg	233	0.749	0.04	0.33	0.33	225.9
5	5.jpg	178	0.431	0.03	0.34	0.34	98.5

Comparison between Original Image and Detection Result

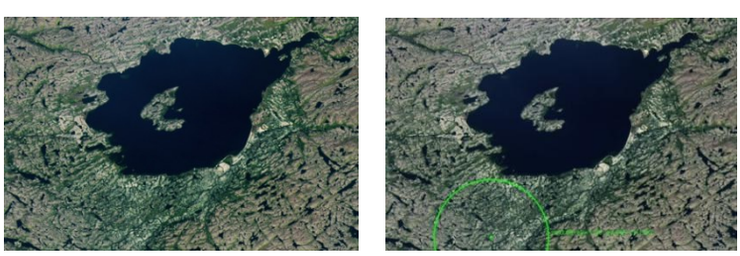

Figure 1 Comparison of Detection Results for 1.jpg

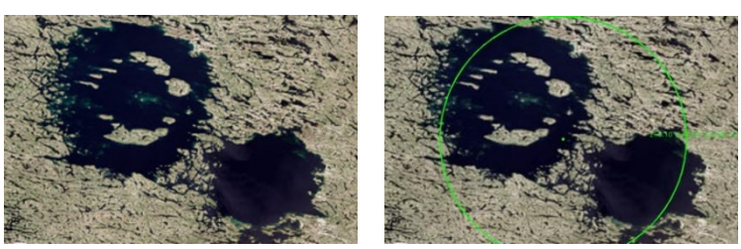

Figure 2 Comparison of Detection Results for 2.jpg

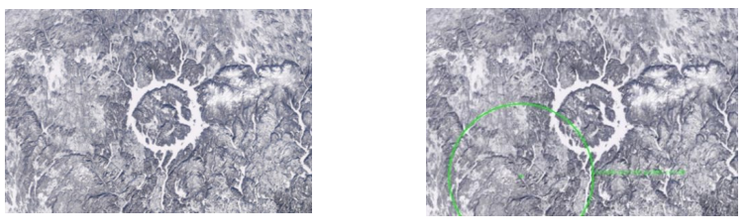

Figure 3 Comparison of Detection Results for 3.jpg

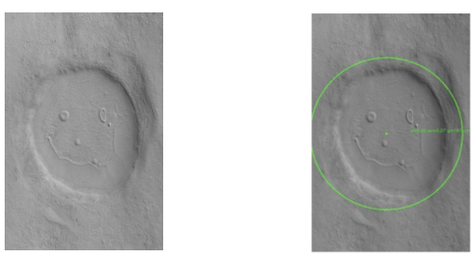

Figure 4 Comparison of Detection Results for 4.jpg

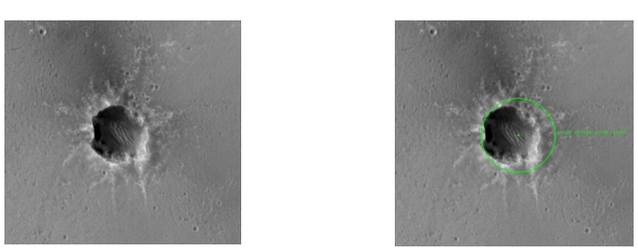

Figure 5 Comparison of Detection Results for 5.jpg

5 Results

5.1 Runtime Performance

The average processing time across five test images is approximately 3.65 s per image. Among all stages, candidate generation (multi-scale Hough transform) is the primary bottleneck, accounting for about 64% of the total runtime. In contrast, I/O, preprocessing, and refinement with scoring are relatively lightweight. For images with more complex textures, the candidate stage takes significantly longer, indicating a strong correlation between the number of candidates and image complexity.

5.2 Detection Accuracy

The method successfully detected the main crater in all test images, providing stable radius estimates ($r = 124–279$ px). For craters with high contrast and well-defined rims, the metrics ev and comp are close to 1.0, consistent with human visual inspection. In contrast, $|z|$ values are generally low (<0.1), suggesting that the "dark-interior/bright-exterior" contrast is not prominent in some crater regions. The final score is therefore mainly supported by ev, comp, and rg.

5.3 Limitations

The proposed method still suffers from limitations in candidate generation efficiency, sensitivity to low-contrast crater rims, and dependency on specific features. Further improvements could be achieved by adopting more efficient candidate generation strategies or incorporating multi-source feature fusion.

6 Discussion and Analysis

The experimental results highlight two main limitations of the proposed method: first, the candidate generation stage is overly time-consuming; second, the method is insufficiently sensitive to low-contrast crater rims. The former is primarily caused by multi-scale Hough detection, which accounts for the majority of the total runtime; the latter is reflected in consistently low $|z|$ values, indicating that the algorithm struggles to capture brightness differences between crater interiors and exteriors.

More specifically, candidate generation is the principal bottleneck in overall efficiency. Across five test images, this stage consumed more than 60% of the runtime on average and was particularly significant for images with complex

textures. This indicates a strong correlation between the number of candidates and image complexity. To optimize further, one possible direction is to reduce the candidate set, for example by employing ROI-based pre-screening or radius stratification, thereby avoiding unnecessary searches.

In terms of detection accuracy, the method performed well on high-contrast craters with clear, closed rims: both ev and comp metrics approached 1.0, consistent with visual inspection. However, in low-contrast regions, the |z| metric was weak, and the results relied more heavily on ring completeness and radial gradient. This suggests that the method is sensitive to illumination conditions and edge sharpness.

From an improvement perspective, this paper argues that refinements can be pursued at three levels:

(1) Geological and geometric priors: introducing slope or shadow information derived from DEM data to enhance rim-background separation;

(2) Geophysical evidence: leveraging signals such as magnetic anomalies, gravity anomalies, or impact melt flows to aid in identifying eroded or buried craters;

(3) Intelligent fusion methods: feeding candidates generated by this method into CNNs for verification, or integrating optical imagery with DEM and slope data for multi-source fusion, thereby enhancing overall robustness.

Overall, the lightweight workflow proposed in this paper demonstrates the capability to achieve cross-planet crater recognition without training data, but it also reveals limitations in speed and stability. These findings provide clear directions for future optimization and open opportunities for integration with deep learning approaches.

7 Conclusion

This study proposes a lightweight and interpretable crater detection pipeline that requires no training. The method enhances grayscale images and performs automatic illumination alignment, then generates candidates using multi-scale Hough transform. It further refines and scores the candidates based on physically meaningful indicators, including ring completeness (comp), radial gradient strength (rg), sector edge votes (ev), and brightness contrast (|z|), with final high-confidence results produced through non-maximum suppression (NMS). Experimental results demonstrate that the pipeline consistently detects the primary crater in both Earth and Mars remote-sensing imagery, with reasonable radius estimates. Moreover, the approach shows low dependency on specific sensors or datasets, indicating strong potential for cross-domain transfer.

In terms of performance, the average runtime is approximately 3.65 seconds per image, with candidate generation as the primary bottleneck, accounting for more than 60% of the total time. For craters with clear, high-contrast rims, the ev and comp indicators approach 1.0, effectively reflecting visual consistency. However, in low-contrast regions, detection accuracy is more sensitive to illumination and edge clarity.

Overall, the proposed method is simple, robust, and interpretable, providing reliable primary crater detection in scenarios with limited computational resources and no training data. Nonetheless, improvements in speed and robustness are still required. Future work will focus on three directions: (i) improving the efficiency of candidate generation; (ii) incorporating DEM, slope, and other terrain priors to enhance robustness under low-contrast conditions; and (iii) integrating the pipeline with lightweight deep learning models to achieve higher accuracy and cross-domain applicability.

Acknowledgements

This paper marks my first systerms attempt at science research and academiuc writing during high school. I would like to thank my mentor and teachers for their patient guidance in topic selection and structuring the paper, which helped me turn scattered ideas into a coherent methodology. I am also grateful to NASA PDS and Google Earth Engine for providing the data resources that made it possible to carry out experiments under limited conditions. Most importantly, I want to thank my family, who encouraged and supported me whenever I stayed up late writing or debugging code. Without their help, this work would not have been possible.

References

[1] L. Xu, J. Jiang, and Y. Ma, A Review of Crater-Based Visual Navigation Techniques, Laser & Optoelectronics Progress, vol. 60, no. 11, pp. 1–online, 2023.

[2] NASA Jet Propulsion Laboratory, Mars Reconnaissance Orbiter (MRO) Context Camera (CTX) Images. NASA Planetary Data System, 2023. [Online]. Available: https://pds.nasa.gov

[3] N. Gorelick, M. Hancher, M. Dixon, et al., "Google Earth Engine: Planetary-scale geospatial analysis for everyone," Remote Sensing of Environment, vol. 202, pp. 18–27, 2017, doi:10.1016/j.rse.2017.06.031.