

Article

A Real-Time Sinkage Detection Method for the Planetary Robotic Wheel-on-Limb System via a Monocular Camera

Baochang Liu ¹, Lihang Feng ²  and Dong Wang ^{1,*} 

¹ School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; 220213627@seu.edu.cn

² College of Electrical Engineering and Control Science, Nanjing Tech University, Nanjing 211816, China; lfeng8@njtech.edu.cn

* Correspondence: kingeast16@seu.edu.cn; Tel.: +86-159-5173-9989

Abstract: When traversing soft and rugged terrain, a planetary rover is susceptible to slipping and sinking, which impedes its movement. The real-time detection of wheel sinkage in the planetary wheel-on-limb system is crucial for enhancing motion safety and passability on such terrain. Initially, this study establishes a measurement of wheel sinkage under complex terrain conditions. Subsequently, a monocular vision-based wheel sinkage detection method is presented by combining the wheel–terrain boundary with the wheel center position (WTB-WCP). The method enables the efficient and accurate detection of wheel sinkage through two-stage parallel computation of the wheel–terrain boundary fitting and wheel center localization. Finally, this study establishes an experimental platform based on a monocular camera and the planetary rover wheel-on-limb system to experimentally validate and comparatively analyze the proposed method. The experimental results demonstrate that the method effectively provides information on the wheel sinkage of the planetary rover wheel-on-limb system, and the relative errors of the method do not exceed 4%. The method has high accuracy and reliability and is greatly significant for the safety and passability of planetary rovers in soft and rugged terrain.

Keywords: sinkage detection; planetary rover; monocular vision



Citation: Liu, B.; Feng, L.; Wang, D. A Real-Time Sinkage Detection Method for the Planetary Robotic Wheel-on-Limb System via a Monocular Camera. *Appl. Sci.* **2024**, *14*, 2319. <https://doi.org/10.3390/app14062319>

Academic Editor: DaeEun Kim

Received: 11 February 2024

Revised: 5 March 2024

Accepted: 8 March 2024

Published: 9 March 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As an important device for extraterrestrial exploration, a planetary rover helps to study extraterrestrial life, water resources, and minerals [1–3]. Despite its significance, operating and conducting exploration missions with planetary rovers presents formidable challenges. This is mainly attributed to the rugged terrain and the unpredictable variations in the softness and hardness of the sandy soil found on both the Martian and lunar surfaces. For instance, the U.S. Mars rover Spirit became stranded in the Troy sands after sinking into soft soil, and all attempts to free it were unsuccessful. This eventually led to it becoming a stationary observation platform [4]. Utilizing the planetary rover wheel-on-limb system to detect the soil environment of the local path in front of traveling, the planetary rover can avoid dangerous paths during travel planning. This is an important method to improve the passability of the planetary rover through soft soil [5–7]. Therefore, the real-time sinkage detection of the wheel-on-limb system is of great significance to ensure the safety of planetary rover operation [8–10].

Numerous studies have addressed the detection of wheel sinkage on planetary rovers [11,12]. Currently, as an important device to ensure the driving safety of planetary vehicles [13], vision sensors are often used to capture wheel sinkage images for sinkage detection [14]. Iagnemma et al. [15] proposed a vision algorithm that utilizes the grayscale values at the edges of the wheel image to estimate wheel sinkage. The algorithm assumes a significant difference in grayscale between the wheel and the terrain. The interface between the terrain and the wheel is identified as the area with the greatest variation

in grayscale. Despite its computational efficiency and simplicity, it is notably affected by variations in lighting and shading. Reina et al. introduced the visual sinkage estimation (VSE) algorithm in 2006 [16]. The algorithm first converts the image of the wheel–ground contact surface into a grayscale map and then determines the contact angle and sink value using equally spaced black concentric circles on the wheel and edge detection. Brook and colleagues implemented Reina’s principles for sinkage detection and employed robust image processing techniques to identify contact points, thereby enhancing the VES algorithm. However, despite its simplicity, feasibility, and efficiency, this method does not consider image noise, which can result in lower precision when calculating sinking [17]. These methods only work well if the difference in grayscale between the wheel and the terrain is large enough.

To overcome the limitations of high grayscale differences, Hegde et al. [18] proposed a method to detect wheel sinking based on the Normalized Cuts image segmentation method. This method recognizes the wheel–soil boundary line by segmenting the wheel–soil image and thus determines the wheel sinkage value. The method mentioned offers the benefits of providing highly accurate and detailed data. However, its main limitations include noise insensitivity and high image quality requirements. In 2015, Lv Fengtian [19] introduced a wheel–soil boundary extraction method based on machine vision and saturation. The method extracts the wheel contour by transforming the image space and image binary processing. The algorithm enhances traditional image enhancement by considering the interaction between the wheel and soil. Additionally, a calculation model has been developed for wheel sinkage under complex ground conditions. Combining the wheel contour and the morphological features of the wheel–terrain contact, the wheel sinkage depth, entry angle, and departure angle are effectively calculated based on the wheel sinkage model. This method requires high image illumination and exhibits low accuracy at small values of wheel sinkage. In order to avoid the influence of environmental factors such as illumination, some scholars have begun to use methods other than image processing for wheel sinkage detection. Hige et al. [20] acquire the depth information of the soil area beside the wheel by placing a TOF camera on the side of the wheel and subsequently calculating the wheel sinkage. Unlike image processing methods, this method is minimally affected by the conditions of the target environment, such as lighting or reflectance. In 2017, Comin et al. [2] developed a distance-based sinking detection method. The method attaches an infrared rangefinder to the underside of the SR chassis and models the ground clearance based on the rangefinder’s measurements to estimate wheel leg sinkage. The method has low estimation error and high computational efficiency. However, it is sensitive to errors on significantly sloping terrain.

In order to effectively detect the front probe wheel sinkage of the planetary rover wheel-on-limb system and to improve the real-time accuracy of the detection, this paper proposes a monocular vision sinkage detection method combining the fitted wheel–terrain boundary with the wheel center position (WTB-WCP) for a planetary rover. Accurate values are provided for wheel sinkage calculation using parallel computation by WTB and WCP. The second section introduces the wheel sinkage calculation model and the monocular vision sinkage detection method. The third section details the hardware configuration of the experimental platform. The fourth section performs experimental validation to substantiate the proposed monocular visual sinkage detection method, and comparative experiments are analyzed. Finally, the fifth section concludes the paper.

2. Measurement Principles and Methods

2.1. Wheel Sinkage Calculation Model

In practical scenarios [21], the ground tends to be rough and uneven. To simulate a realistic driving environment and facilitate calculations and analysis, it is assumed that there is a height difference between the point where the wheel flange contacts the ground (entry point A) and the point where the wheel flange loses contact with the ground (departure point B). Figure 1 illustrates the establishment of a cartesian coordinate system on a surface,

with the center of the circle labeled as the origin. The point where the wheel flange makes contact with the ground, known as the entry point, is labeled as point A. Conversely, the point where the wheel flange leaves the ground, known as the departure point, is labeled as point B. The angle θ_1 formed by the line OA and the vertical direction, is referred to as the entry angle. Similarly, the angle θ_2 included between the line OB and the vertical direction is termed the departure angle [22]. The formula for calculating the entry angle and departure angle is presented as follows.

$$\theta_1 = \frac{\pi}{2} - \arctan\left(\frac{y_A}{x_A}\right) \tag{1}$$

$$\theta_2 = \frac{\pi}{2} - \arctan\left(\frac{y_B}{x_B}\right) \tag{2}$$

Since the wheel–terrain boundary is an irregular curve in rugged terrain, a fitted line was used as the wheel–terrain boundary. The wheel sinkage depth is $z = R - OC$. The wheel–terrain boundary can be expressed as

$$y = ax + b \tag{3}$$

where a and b are the parameters obtained from the fit. Thus, wheel sinkage can be expressed as Equation (4).

$$z = R - \frac{|b|}{\sqrt{a^2 + 1}} \tag{4}$$

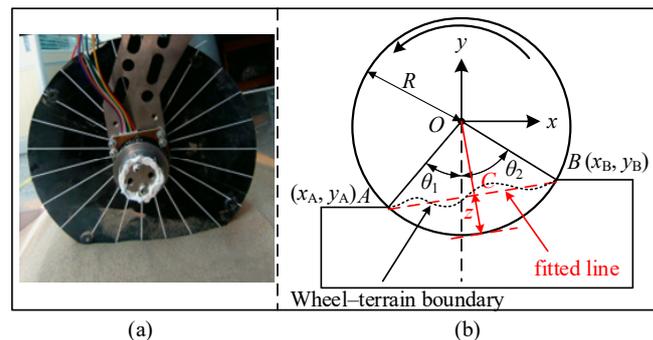


Figure 1. (a) Physical Image of wheel sinkage; (b) calculation model for wheel sinkage.

2.2. Monocular Vision Sinkage Detection Method

In order to correctly detect wheel sinkage, the camera’s exposure parameters and camera focal length need to be adjusted before starting. Camera imaging distortion and installation angle deviation can cause wheel distortion in the collected images. The actual circular wheel appears as an elliptical image in the initial photograph. Thus, it is necessary to project the image of the wheel onto a vertical surface using perspective transformation. Assuming that (u, v) is the pixel coordinates of the original image and (x, y) is the pixel coordinates of the corresponding perspective transformed image, the relationship between the two can be expressed as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = M \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{5}$$

where M is the perspective transformation matrix. The pixel coordinates of the image after perspective transformation are calculated as Equation (6).

$$\begin{cases} x = (\alpha_{11} \times u + \alpha_{12} \times v + \alpha_{13}) / (\alpha_{31} \times u + \alpha_{32} \times v + \alpha_{33}) \\ y = (\alpha_{21} \times u + \alpha_{22} \times v + \alpha_{23}) / (\alpha_{31} \times u + \alpha_{32} \times v + \alpha_{33}) \end{cases} \tag{6}$$

Since the perspective transformation matrix has 8 degrees of freedom, M can be obtained by providing the original pixel coordinates of the four points on the wheel surface and the corresponding pixel coordinates after perspective transformation.

Canny edge detection is performed on the image to obtain pixel information at the wheel–terrain boundary. The image is first converted to the grayscale image, and Gaussian filtered using the Gaussian kernel in Equation (7).

$$G = \begin{bmatrix} 1/16 & 2/16 & 1/16 \\ 2/16 & 4/16 & 2/16 \\ 1/16 & 2/16 & 1/16 \end{bmatrix} \tag{7}$$

Subsequently, the magnitude and direction of the gradient are computed using the Sobel operator. Assuming that $H_{(i,j)}$ is the pixel point to be computed, the gradients in the x -direction and y -direction are, respectively, expressed as

$$\begin{cases} G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} A_{(i-1,j-1)} & A_{(i-1,j)} & A_{(i-1,j+1)} \\ A_{(i,j-1)} & H_{(i,j)} & A_{(i,j+1)} \\ A_{(i+1,j-1)} & A_{(i+1,j)} & A_{(i+1,j+1)} \end{bmatrix} \\ G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} A_{(i-1,j-1)} & A_{(i-1,j)} & A_{(i-1,j+1)} \\ A_{(i,j-1)} & H_{(i,j)} & A_{(i,j+1)} \\ A_{(i+1,j-1)} & A_{(i+1,j)} & A_{(i+1,j+1)} \end{bmatrix} \end{cases} \tag{8}$$

where $A_{(i-1,j-1)} \dots A_{(i+1,j+1)}$ represent the pixel value around point $H_{(i,j)}$. Thus, the gradient strength G_H and gradient direction θ_H at point $H_{(i,j)}$ can be expressed as

$$\begin{cases} G_H = \sqrt{G_x^2 + G_y^2} \\ \theta_H = \arctan\left(\frac{G_y}{G_x}\right) \end{cases} \tag{9}$$

Finally, the edge image is obtained by non-maximum suppression and hysteresis thresholding. Fitting the pixels at the wheel–terrain boundary is based on the least squares method. Combined with Equation (3), the pixels at the wheel–terrain boundary are fitted by

$$\begin{bmatrix} a \\ b \end{bmatrix} = \left(\begin{bmatrix} x_1 & \dots & x_n \\ 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} x_1 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} x_1 & \dots & x_n \\ 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \tag{10}$$

where $(x_1, y_1) \dots (x_n, y_n)$ represent coordinates of the wheel–terrain boundary points extracted from Canny edge detection. The wheel–terrain boundary (WTB) can be obtained from Equations (3) and (10).

Hough circle detection is used to identify the wheel center position and wheel radius. Assume that point H is a pixel point of the transformed image. The image is first grayed out, and its gray value H_Y can be calculated by

$$H_Y = 0.299 \times H_R + 0.587 \times H_G + 0.114 \times H_B \tag{11}$$

where $H_R, H_G,$ and H_B are color channel values of point H . Assuming that the coordinates of point H are (i, j) , a Gaussian filter is applied to reduce the noise, and the process can be expressed as

$$H_Y = \begin{bmatrix} 1/16 & 2/16 & 1/16 \\ 2/16 & 4/16 & 2/16 \\ 1/16 & 2/16 & 1/16 \end{bmatrix} \times \begin{bmatrix} A_{Y(i-1,j-1)} & A_{Y(i-1,j)} & A_{Y(i-1,j+1)} \\ A_{Y(i,j-1)} & H_{Y(i,j)} & A_{Y(i,j+1)} \\ A_{Y(i+1,j-1)} & A_{Y(i+1,j)} & A_{Y(i+1,j+1)} \end{bmatrix} \tag{12}$$

where $A_{Y(i-1,j-1)} \dots A_{Y(i+1,j+1)}$ represent the pixel value around point $H_{(i,j)}$. Subsequently the image is subjected to edge detection and the center and radius of the circle are detected using the Hough gradient method. Suppose the Equation of the circle is

$$(x - m)^2 + (y - n)^2 = r^2 \tag{13}$$

where (m, n) represents the center coordinate and r represents the radius. From the above equation and Figure 2, it can be seen that the gradient direction of any point P on the circle is directed to the center of the circle. In other words, the line in the direction of the gradient through any point on the circle must pass through the center of the circle.

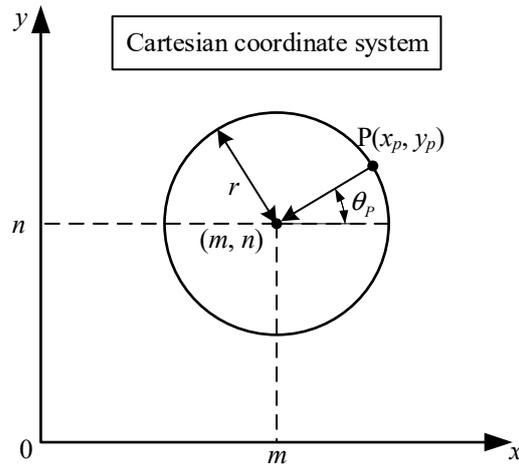


Figure 2. The gradient direction of a point on the circle.

Establishing a straight line in the gradient direction for each pixel point (x_i, y_i) in the edge detection according to Equation (9) can be expressed as

$$y = \frac{G_{y_i}}{G_{x_i}} x + y_i - \frac{G_{y_i}}{G_{x_i}} x_i \tag{14}$$

where G_{x_i} and G_{y_i} are the gradients in the x-direction and y-direction. As shown in Figure 3a, the weight of the non-zero pixel points is added one in the above line, and ultimately, the point (x_0, y_0) with the largest weight on the entire image is the wheel center. According to Figure 3b, the distance of all non-zero pixel points of the edge image to the wheel center is calculated according to Equation (15), and the number of times this distance has occurred is recorded using C_{r_i} . The Hough transform utilizes the transformation relationship between two spaces to convert a curve or straight line with identical shape in one space into a point in another coordinate space, creating a peak. This transformation translates the circle detection problem in an image into a statistical peak problem.

$$\begin{cases} r_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2} \\ C_{r_i} = C_{r_i} + 1 \end{cases} \tag{15}$$

The distance r_0 that occurs most is the radius of the wheel. In order to reduce the amount of computation, the radius is limited to between 350 and 400 pixels based on the wheel information. The wheel center position (WCP) (x_0, y_0) and the wheel pixel radius r_0 can be obtained from the results.

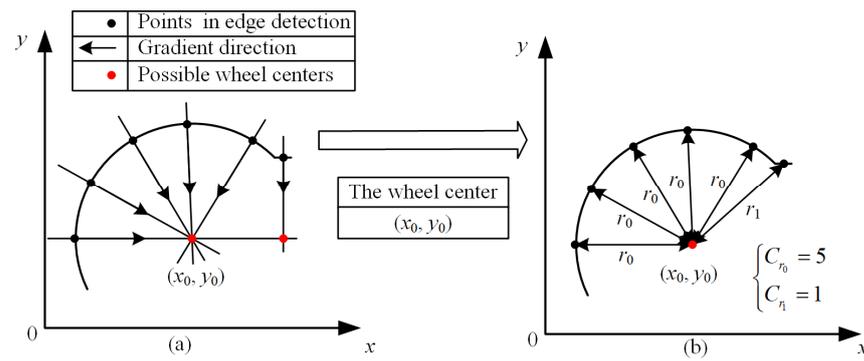


Figure 3. (a) The process of identifying the wheel center; (b) the process of identifying the wheel pixel radius.

Following the aforementioned processing steps, we obtain the pixel radius of the wheel, the coordinates of the wheel center, and the fitted line that represents the wheel–terrain boundary. Combined with Equations (3) and (4), the wheel sinkage z can be calculated by

$$\begin{cases} l_p = \frac{|ax_0 - y_0 + b|}{\sqrt{a^2 + 1}} \\ z = \frac{r_0 - l_p}{r_0} R \end{cases} \quad (16)$$

In the Equation, r_0 is the pixel radius of the wheel, R is the actual radius of the wheel, and l_p is the pixel distance from the wheel center to the fitted line.

2.3. Flow and Implementation of the Method

According to the principle of the monocular visual subsidence detection method proposed in Section 2.2, the specific flow of the method is shown in Figure 4. The method consists of three parts: wheel–terrain boundary (WTB) fitting, wheel center position (WCP) acquisition, and wheel sinkage value calculation. Among them, the acquisition of the wheel center position requires Hough circle detection and identification of the wheel center and radius to the image; wheel–terrain boundary requires Canny edge detection and wheel–terrain boundary fitting to the image. Finally, the wheel sinkage value is calculated according to the wheel sinkage model with WTB-WCP. In addition, a specific implementation of the monocular vision sinkage detection method is shown in Algorithm 1.

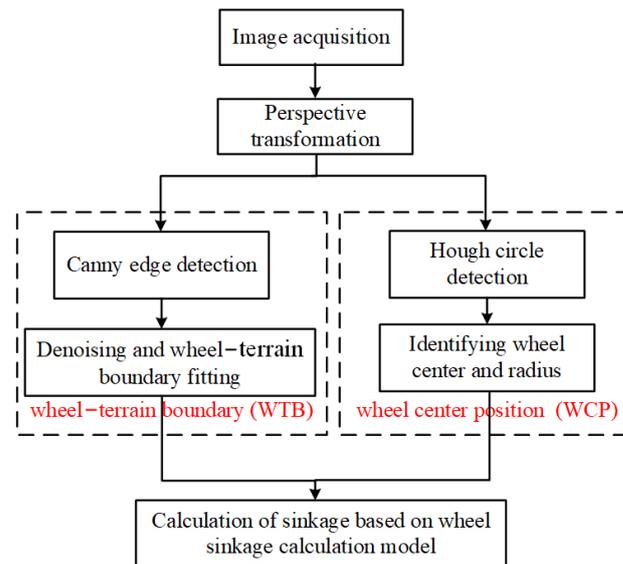


Figure 4. Flow chart of the monocular vision sinkage detection method.

Algorithm 1: Monocular visual sinkage detection method

Data: capture image $frame$, original image coordinates $psrc$ [4], transform image coordinates $pdst$ [4], wheel–terrain junction ROI

Result: wheel sinkage z

```

1.   $transM \leftarrow \text{getPerspectiveTransform}(psrc [4], pdst [4]);$ 
2.   $frame \leftarrow \text{warpPerspective}(transM);$ 
3.   $grayImage \leftarrow \text{cvtColor}(frame);$ 
4.   $grayImage \leftarrow \text{GaussianBlur}(\text{Size}(3,3));$ 
5.   $circle \leftarrow \text{HoughCircles}(grayImage, CV\_HOUGH\_GRADIENT);$ 
6.   $dst \leftarrow \text{Canny}(frame);$ 
7.   $points \leftarrow \text{dst}(ROI);$ 
8.   $n \leftarrow \text{points.size}();$ 
9.  for  $i \leftarrow 0$  to  $n$  do
10.      $sumx \leftarrow sumx + \text{points}_i.x;$ 
11.      $sumy \leftarrow sumy + \text{points}_i.y;$ 
12.      $sumx2 \leftarrow sumx2 + \text{points}_i.x^2;$ 
13.      $sumxy \leftarrow sumxy + \text{points}_i.x \times \text{points}_i.y;$ 
14.  end
15.   $b \leftarrow (n \times sumxy - sumx \times sumy)/(n \times sumx2 - sumx^2);$ 
16.   $a \leftarrow (sumy - b \times sumx)/n;$ 
17.   $d \leftarrow (a \times circle.center.x - circle.center.y + b)/(a^2 + b^2)^{0.5}$ 
18.   $z \leftarrow (circle.r - d) \times 80/circle.r;$ 

```

3. Experimental Platform

As shown in Figure 5, the planetary rover wheel-on-limb system consists of a mechanical arm control subsystem, an experimental wheel sensing subsystem, and a wheel sinkage detection subsystem. Among them, the mechanical arm control subsystem is used for terrain following and constant force tracking; the experimental wheel sensing subsystem is used to sense the wheel–terrain contact force; the wheel sinkage detection subsystem is used to detect the wheel sinkage depth in real time to judge the planetary rover passability [7]. The actual wheel radius in this paper is 80 mm. Since a fixed load is applied to the wheels during the movement of the planetary rover, the wheel-on-limb system simulates a real wheel-passing scenario by applying the same load to the perceived wheels via the mechanical arm. Detecting the sinkage of the experimental wheel in real time provides an important basis for the planetary rover’s passability. The wheel sinkage detection subsystem consists of three parts: a vision acquisition module, a data processing module, and auxiliary tools. To capture real-time images of the detection wheel and the wheel–terrain interface, the visual acquisition module must cover the entire wheel and the surrounding area on the wheel side. This coverage should extend at least as far as the diameter, length, and radius width of the wheel. The data processing module processes the collected visual image information in real-time to output the values of wheel sinkage, as well as the entry and departure angles of the wheel–terrain contact. During data processing, auxiliary tools are employed to determine the installation position of the visual acquisition module and its related parameters. The distortion-free USB camera ov5640 was chosen as the vision acquisition module based on its characteristics, as shown in Table 1. Measurement results confirm that the camera meets the requirements for measuring wheel sinkage.

Table 1. Detailed parameters of USB camera ov5640.

Maximum Resolution	Pixel	Focal Length	Field of View	Capture Rate
4K	8 million	2.2 mm	140°	30 fps

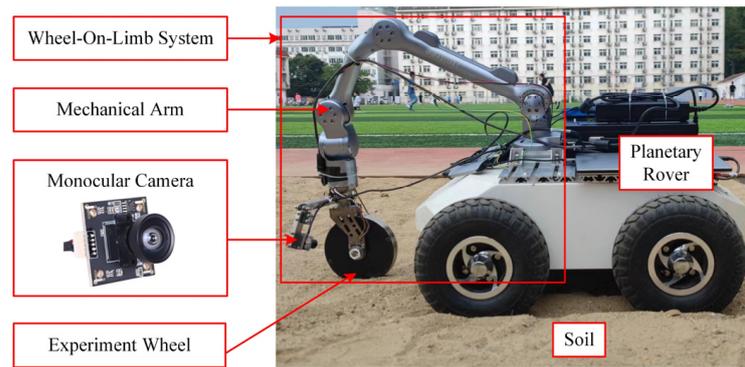


Figure 5. Planetary rover wheel-on-limb system.

4. Results and Discussion

4.1. Experimental Results of Wheel Sinkage Detection Method

The method itself was tested in a soil tank using the experimental platform described in the previous section. Figure 6 shows the resultant images corresponding to each processing step of the wheel sinkage detection method. The wheel–terrain boundary obtained by canny edge detection and the wheel–terrain boundary fitting is shown as the green straight line at the top of Figure 6. The Equation of the fitted straight line is $y = -0.13x + 739$. The wheel contour, obtained by perspective transformation and Hough circle detection, is shown as a pink circle at the bottom of Figure 6, with the wheel center represented by a green point. The coordinates of the wheel center are obtained as (409, 381), and the pixel radius of the wheel r_0 is 378 pixels based on the Hough detection result. The wheel sinkage can be calculated by the pixel coordinates of the center of the wheel, the fitted straight line, and Equation (16). By using the wheel center coordinates and the pixel coordinates at the ends of the wheel–terrain boundary, the entry angle and departure angle of the wheel can be calculated according to Equations (1) and (2). Table 2 shows the detection results in the experiment. In order to verify the accuracy of the monocular visual sinkage detection method proposed in this paper, eight experiments were conducted for wheel pictures with different sinkage depths. The specific experimental process diagram is shown in Figure 7. The wheel center data and sinkage data collected during the experiments will be used in the subsequent analysis of Sections 4.2 and 4.3.

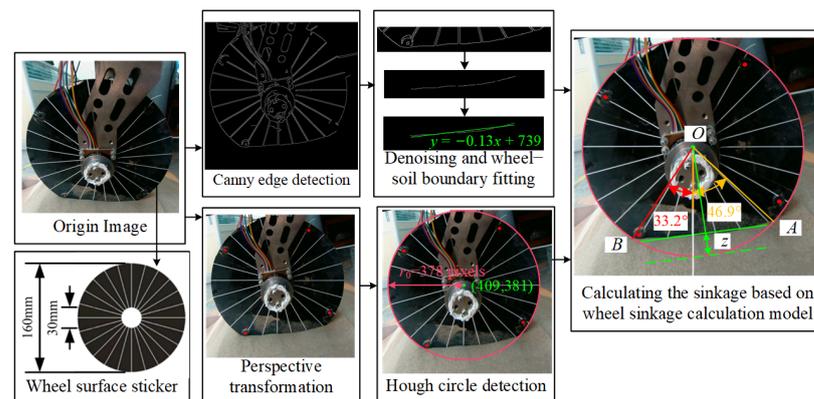


Figure 6. Wheel sinkage detection results.

Table 2. Wheel sinkage detection results.

Detection Parameters	r_0 (px)	l_p (px)	Sinkage z (mm)	Entry Angle θ_1 (°)	Departure Angle θ_2
Results	378	76	16.08	33.2	46.9



Figure 7. Results of eight different wheel sinkage depth tests: (a) Origin image; (b) WCP results; (c) WTB results; (d) detection results.

4.2. Reliability Verification Results

The reliability of the wheel sinkage detection method was verified by conducting qualitative and quantitative analyses of the perspective transformation and Hough circle detection results. To analyze the result of Hough circle detection, a circular sticker with spokes that corresponds to the wheel surface’s dimensions is employed, as depicted in the lower-left corner of Figure 6. The precise pixel coordinates of the wheel center should be determined by analyzing the wheel surface stickers and spokes and then compared

with the results obtained from the Hough circle detection, which are shown in Figure 7b. Table 3 displays the results, and Figure 8 illustrates the pixel error, actual error, and error as a percentage of the radius for each test. According to the results, the pixel coordinate error of the Hough circle detection result is found to be below 10 pixels, with the actual error falling within 2 mm. In comparison to an actual wheel with a diameter of 160 mm, the error as a percentage of radius does not exceed 2%. The average pixel error for the eight experiments was 3.62 pt, with an actual error of 0.77 mm and a percentage error of 0.96% relative to the wheel radius. The method exhibits a high degree of accuracy and reliability.

Table 3. Wheel sinkage detection results.

Test	Center Coordinates (Auxiliary Detection)	Center Coordinates (Hough Detection)	Pixel Error (pt)	Actual Error (mm)	Error as a Percentage of Radius
1	406, 378	410, 380	4.47	0.95	1.19%
2	408, 379	408, 380	1.00	0.21	0.26%
3	406, 380	408, 380	2.00	0.42	0.53%
4	406, 377	408, 379	2.83	0.60	0.75%
5	410, 379	408, 380	2.24	0.48	0.60%
6	410, 378	406, 380	4.47	0.94	1.18%
7	406, 376	408, 382	6.32	1.33	1.66%
8	410, 382	406, 378	5.66	1.21	1.51%
	Average error		3.62	0.77	0.96%

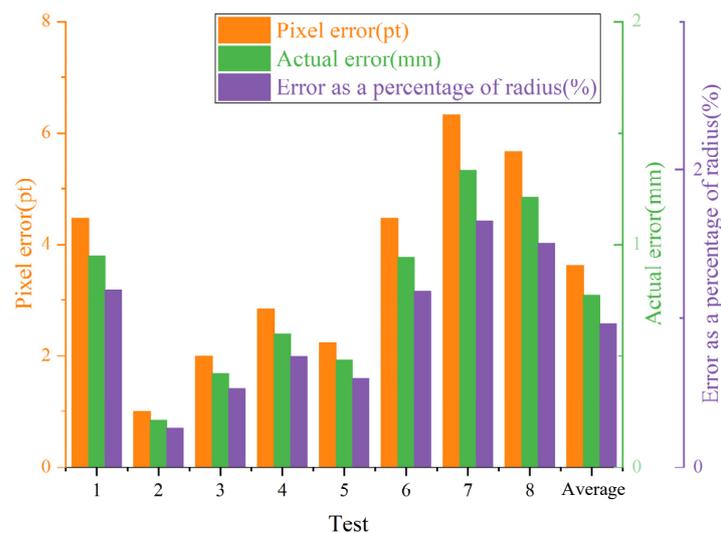


Figure 8. Error plots of Hough circle detection results.

4.3. Comparison Analysis

The conventional visual sinkage estimation (VSE) algorithm calculates the amount of wheel sinkage by capturing images from a monocular camera attached to one side of the wheel and utilizing equally spaced concentric circles on a white background. This algorithm detects wheel sinkage with good accuracy and is a superior traditional detection method. The monocular vision sinkage detection (MVSD) method proposed in this paper is compared with the VSE algorithm in the following experiment. The experiments were conducted to detect the amount of wheel sinkage using the VSE algorithm and MVSD method for the same wheel sinkage scenario, respectively, and the specific experimental procedure is shown in Figure 7. The experimental results and detection errors for the eight experiments are demonstrated in Table 4 and Figure 9. According to the experimental results, the errors of the sinkage values measured by the MSVD method are all smaller than the errors of the sinkage values measured by the VSE algorithm. The relative errors measured with the MSVD method do not exceed 4%, while the relative errors measured

with the VSE algorithm reach up to 8.66%. Therefore, the MSVD algorithm in this paper has higher accuracy and can provide more accurate wheel sinkage information for planetary rover passability prediction.

Table 4. Wheel sinkage detection results.

Test	Reference Sinkage Value (mm)	Sinkage Value Detected by MVSD (mm)	Sinkage Value Detected by VSE (mm)	Relative Error of MVSD	Relative Error of VSE
1	0.00	0.00	0.00	0.00%	0.00%
2	8.10	8.25	8.30	1.85%	2.47%
3	16.06	16.30	15.62	1.49%	2.74%
4	20.43	20.05	19.86	1.86%	2.79%
5	25.42	24.91	24.01	2.01%	5.55%
6	30.12	29.28	28.56	2.79%	5.18%
7	32.83	31.79	30.32	3.17%	7.65%
8	36.16	34.77	33.03	3.84%	8.66%

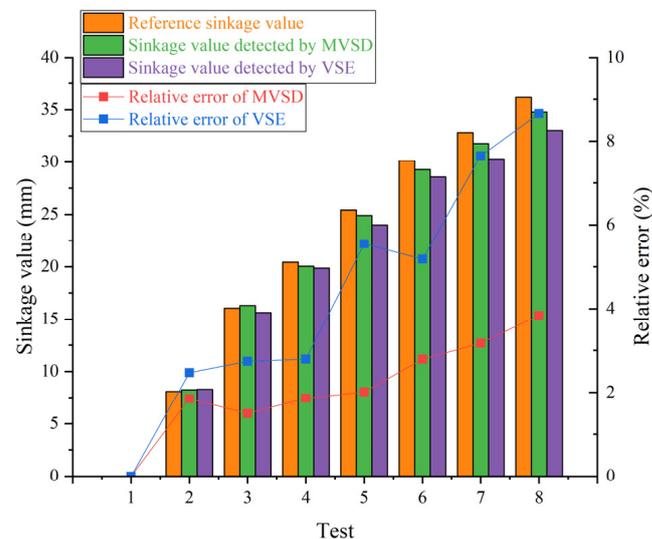


Figure 9. Comparative analysis test results.

5. Conclusions

This article presents a monocular visual sinkage detection method combining WTB and WCP for the planetary rover wheel-on-limb system. The wheel sinkage is captured using a single camera, and the original image is transformed into a frontal view of the wheel surface through a perspective transformation. Subsequently, the Hough circle detection algorithm is employed to obtain the pixel coordinates and radius of the wheel center in pixels. The Canny edge detection algorithm is utilized to capture image information at the wheel–terrain boundary. Next, the least squares fitting method is employed to obtain the best-fit straight line representing the boundary at the intersection. Finally, the settlement of the wheel should be calculated using the wheel settlement calculation model. The algorithm was ultimately validated using a soil tank experimental platform. The results indicated that the pixel coordinate error of the Hough circle detection result is below 10 pixels, and the actual error is less than 2 mm. The error as a percentage of radius does not exceed 2% when compared to the actual wheel size, which has a diameter of 160 mm. The relative errors of the monocular vision sinkage detection method do not exceed 4%, which exhibits a higher level of accuracy and reliability compared to the conventional visual sinkage estimation algorithm.

This method is able to work with the wheel-on-limb system to determine the passability of the planetary rover's traveling area, which is greatly significant for the driving safety of the planetary rover during exoplanet exploration. Similarly, the method can be

used for sinkage detection of wheeled vehicles in unknown terrain in the field, which is important for passability detection and vehicle traveling safety in the forward area. In addition, this sinkage detection method can be prepared for the study of terrain mechanics in rugged terrain to realize the calculation of wheel–soil interaction forces moving over rugged terrain.

Author Contributions: Conceptualization, B.L. and D.W.; methodology, B.L. and L.F.; validation, B.L. and D.W.; investigation, B.L. and D.W.; resources, D.W.; writing—original draft preparation, B.L.; writing—review and editing, L.F. and D.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 62103184.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: The authors acknowledge Zixu Guo for his contributions to the process of equipment debugging and field testing.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sanguino, T.D.M. 50 years of rovers for planetary exploration: A retrospective review for future directions. *Robot. Auton. Syst.* **2017**, *94*, 172–185. [CrossRef]
2. Comin, F.J.; Lewinger, W.A.; Saaj, C.M.; Matthews, M.C. Trafficability Assessment of Deformable Terrain through Hybrid Wheel-Leg Sinkage Detection. *J. Field Robot.* **2017**, *34*, 451–476. [CrossRef]
3. Gao, Y.; Chien, S. Review on space robotics: Toward top-level science through space exploration. *Sci. Robot.* **2017**, *2*, eaan5074. [CrossRef] [PubMed]
4. Mars Rover Escapes from the “Bay of Lamentation”. Available online: <http://space.newscientist.com/article9286> (accessed on 27 August 2018).
5. Allouis, E.; Marc, R.; Gancet, J.; Nevatia, Y.; Cantori, F.; Sonsalla, R.U.; Fritsche, M.; Machowinski, J.; Vögele, T.; Comin, F.; et al. FP7 FASTER Project—Demonstration of Multi-platform Operation for Safer Planetary Traverses. In Proceedings of the Symposium on Advanced Space Technologies in Robotics and Automation, Noordwijk, The Netherlands, 11–13 May 2015.
6. Zhang, W.Y.; Lyu, S.P.; Xue, F.; Yao, C.; Zhu, Z.; Jia, Z.Z. Predict the Rover Mobility Over Soft Terrain Using Articulated Wheeled Bevameter. *IEEE Robot. Autom. Lett.* **2022**, *7*, 12062–12069. [CrossRef]
7. Feng, L.H.; Miao, T.Y.; Jiang, X.; Cheng, M.; Hu, Y.; Zhang, W.G.; Song, A.G. An Instrumented Wheel to Measure the Wheel-Terrain Interactions of Planetary Robotic Wheel-on-Limb System on Sandy Terrains. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–13. [CrossRef]
8. Ye, P.J.; Sun, Z.Z.; Rao, W.; Meng, L.Z. Mission overview and key technologies of the first Mars probe of China. *Sci. China Technol. Sci.* **2017**, *60*, 649–657. [CrossRef]
9. Srividhya, G.; Sharma, G.; Kumar, H.N.S. Software for Modelling and Analysis of Rover on Terrain. In Proceedings of the Conference on Advances in Robotics, Pune, India, 4–6 July 2013; pp. 1–8. [CrossRef]
10. Guo, J.L.; Ding, L.; Gao, H.B.; Guo, T.Y.; Liu, G.J.; Peng, H. An Apparatus to Measure Wheel-Soil Interactions on Sandy Terrains. *IEEE-Asme Trans. Mech.* **2018**, *23*, 352–363. [CrossRef]
11. Spiteri, C.; Al-Milli, S.; Gao, Y.; de León, A.S. Real-time visual sinkage detection for planetary rovers. *Robot. Auton. Syst.* **2015**, *72*, 307–317. [CrossRef]
12. Xue, F.; Yao, C.; Yuan, Y.; Ge, Y.T.; Shi, W.J.; Zhu, Z.; Ding, L.; Jia, Z.Z. Wheel-Terrain Contact Geometry Estimation and Interaction Analysis Using Aside-Wheel Camera Over Deformable Terrains. *IEEE Robot. Autom. Lett.* **2023**, *8*, 7639–7646. [CrossRef]
13. Feng, L.H.; Wang, S.; Wang, D.; Xiong, P.W.; Xie, J.J.; Hu, Y.; Zhang, M.M.; Wu, E.Q.; Song, A.G. Mobile-DeepRFB: A Lightweight Terrain Classifier for Automatic Mars Rover Navigation. *IEEE T Autom. Sci. Eng.* **2023**. early access. [CrossRef]
14. Matsumura, R.; Ishigami, G. Visualization and analysis of wheel camber angle effect for slope traversability using an in-wheel camera. *J. Terramechanics* **2021**, *93*, 1–10. [CrossRef]
15. Iagnemma, K.; Kang, S.W.; Shibly, H.; Dubowsky, S. Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. *IEEE Trans. Robot. Autom.* **2004**, *20*, 921–927. [CrossRef]
16. Reina, G.; Ojeda, L.; Milella, A.; Borenstein, J. Wheel slippage and sinkage detection for planetary rovers. *IEEE-Asme Trans. Mech.* **2006**, *11*, 185–195. [CrossRef]

17. Brooks, C.A.; Iagnemma, K.D.; Dubowsky, S. Visual wheel sinkage measurement for planetary rover mobility characterization. *Auton. Robot.* **2006**, *21*, 55–64. [[CrossRef](#)]
18. Hegde, G.; Robinson, C.J.; Ye, C.; Stroupe, A.; Tunstel, E. Computer vision based wheel sinkage detection for robotic lunar exploration tasks. In Proceedings of the 2010 IEEE International Conference on Mechatronics and Automation, Xi'an, China, 4–7 August 2010; pp. 1777–1782. [[CrossRef](#)]
19. Lv, F.T.; Li, N.; Gao, H.B.; Ding, L.; Liu, Z.; Deng, Z.Q.; Song, X.G.; Yan, B. Planetary Rovers' Wheel Sinkage Detection Based on Wheel-Soil Boundary. In Proceedings of the 2015 International Conference on Electrical, Computer Engineering and Electronics; Advances in Computer Science Research. Atlantis Press: Amsterdam, The Netherlands, 2015; Volume 24, pp. 1156–1161. [[CrossRef](#)]
20. Higa, S.; Nagaoka, K.; Yoshida, K. Online estimation of wheel sinkage and slippage using a tof camera on loose soil. In Proceedings of the ISTVS 8th Americas Regional Conference, Detroit, MI, USA, 12–14 September 2016; pp. 1–9.
21. Chhaniyara, S.; Brunskill, C.; Yeomans, B.; Matthews, M.C.; Saaj, C.; Ransom, S.; Richter, L. Terrain trafficability analysis and soil mechanical property identification for planetary rovers: A survey. *J. Terramechanics* **2012**, *49*, 115–128. [[CrossRef](#)]
22. Gao, H.; Lv, F.; Yuan, B.; Li, N.; Ding, L.; Li, N.; Liu, G.; Deng, Z. Sinkage definition and visual detection for planetary rovers wheels on rough terrain based on wheel–soil interaction boundary. *Robot. Auton. Syst.* **2017**, *98*, 222–240. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.