AFRL-AFOSR-JP-TR-2024-0043

LANDrone: Developing robotic tools for adding non-planar surface landing capability to UAVs

Armagan elibol
JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY
1, ASAHIDAI 1-CHOME
NOMI-SHI, , 923-1211
JPN

01/19/2024
Final Technical Report

Air Force Research Laboratory
Air Force Office of Scientific Research
Asian Office of Aerospace Research and Development
Unit 45002, APO AP 96338-5002

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED	
20240119	Final	START DATE 20200918	END DATE 20230917

4. TITLE AND SUBTITLE
LANDrone: Developing robotic tools for adding non-planar surface landing capability to UAVs

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
	FA2386-20-1-4019	
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Armagan elibol

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY
1, ASAHIDAI 1-CHOME
NOMI-SHI 923-1211
JPN

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AOARD UNIT 45002 APO AP 96338-5002	AFRL/AFOSR IOA	AFRL-AFOSR-JP-TR-2024-0043

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Unmanned Aerial Vehicles (UAVs) have been attracting much attention and changing our daily lives. Recent technological advances in the development of UAVs have drastically increased both their general capabilities and areas of application. Among many others, one of the areas that benefit immediately from using UAVs could be remote inspection, since they can provide an alternative means of access to structures and collect data from locations difficult to reach for human inspectors. In this report, we present our advancements in developing a lightweight robotic landing gear for enabling UAVs to land on irregular surfaces. We have studied the limits and limitations of our novel design both theoretically and experimentally, leading to some further improvements. We have also studied improving the image-matching pipeline, aiming to integrate and use it while approaching and/or selecting a landing site in the long run.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
a. REPORT	b. ABSTRACT	c. THIS PAGE	SAR	33
U	U	U		

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
FUMIO KOJIMA	315-227-7007

Standard Form 298 (Rev.5/2020)
Prescribed by ANSI Std. Z39.18

Final Report
LANDrone: Developing Robotic Tools for Adding Non-Planar Surface Landing Capability to UAVs
FA2386-20-1-4019

PI: Dr. Armagan Elibol, aelibol@jaist.ac.jp
Japan Advanced Institute of Science and Technology
PO: Dr. Fumio Kojima, fumio.kojima.jp@us.af.mil
Asian Office of Aerospace R&D (AOARD)

September 19, 2023

Contents

1 Introduction **2**

2 Novel Design for Robotic Landing Gear **2**
2.1 Design of the Robotic Legs 5
2.2 Design of the Passive Decompression Device 8
2.3 The Operational Principle of Robotic Landing Gear 9

3 Enhancements for Feature-based Image Matching Pipeline **10**
3.1 Pre-Filtering Step for Outlier Reduction Method 11
3.2 Extension of Pre-Filtering Step for Multiple Motion Estimation 11

4 Experiments **12**
4.1 Experimental Results for Novel Robotic Landing Design 12
4.2 Experimental Results for Enhancements on Image Matching 17

5 Conclusions **19**
5.1 Scientific Outcomes 20

Abstract

Unmanned Aerial Vehicles (UAVs) have been attracting much attention and changing our daily lives. Recent technological advances in the development of UAVs have drastically increased both their general capabilities and areas of application. Among many others, one of the areas that benefit immediately from using UAVs could be remote inspection, since they can provide an alternative means of access to structures and collect data from locations difficult to reach for human inspectors. In this report, we present our advancements in developing a lightweight robotic landing gear for enabling UAVs to land on irregular surfaces. We have studied the limits and limitations of our novel design both theoretically and experimentally leading to some further improvements. We have also studied improving the image matching pipeline aiming to integrate and use while approaching and/or selecting a landing site in the long run.

1 Introduction

Nowadays robotic vehicles are increasingly used in a variety of applications and changing our daily life. With the improvements in their capabilities (*e.g.*, mobility, exploration, data collection, autonomy, and many others), they have been viable for the tasks that need to take place in hazardous environments. Periodic maintenance and inspection of man-made high-rise structures is one of such tasks. It is widely known that Wall Climbing Robots (WCRs) were proposed to be used for the inspection and cleaning of buildings, replacing long-reach fixed-based manipulators. However, the moving speed of WCRs is usually relatively slow, which often needs to be provided by roof cables and winches. This significantly limits the scope of applications of WCRs. Notably, the Unmanned Aerial Vehicle (UAV) technology has made astonishing progress in recent years, and its application areas are not only limited to aerial photography, entertainment, and similar others. An increasing number of tasks requiring UAVs to physically interact with their surroundings have been demonstrated (*e.g.*, inspection [14, 8] and agriculture [10, 7, 6]). In light of recent technological advances of UAVs in payload capacity, endurance, flight stability and control, and user interface, as well as the decline in the price of hardware platforms, UAVs begin to be used for civil operations under the related regulations and guidance. Therefore, UAVs are deemed an appropriate alternative for the maintenance and inspection tasks to overcome the aforementioned issues with WCRs. For this reason, we propose a lightweight robotic landing gear prototype that enables the UAV to attach ideally to any shape of the surface.

Different application-oriented platforms [15, 11] have been developed in response to the nature and needs of bridge inspection tasks. As mentioned above, off-the-shelf or custom-built UAV platforms can be potentially utilized for non-destructive inspection and maintenance of man-made high-rise structures, bridges, and bodies of airplanes. There have already been several attempts in remote inspection (*e.g.*, wall, dam, and many others [16, 19]), in which UAVs were an indispensable tool thanks to their ability to obtain data through optical sensors available on board. Specifically, for non-destructive inspection on high-rise buildings, there is always the risk of being affected by the wind. Therefore, it is of utmost importance to keep the UAV attached to the target surface securely and obtain the necessary data correctly.

In this report, we summarize our studies on (1) developing a novel robotic landing design for UAVs to land on arbitrarily shaped surfaces and (2) a simple yet efficient method for enhancing the performance of image matching.

2 Novel Design for Robotic Landing Gear

In this section, we provide details on the proposed robotic landing gear for off-the-shelf UAVs made up of 3 different parts; the Angle Control Part (ACP), the Counterweight Control Part (CCP), and the Environmental Sensor Part (ESP). As a part of its modular design, the distance between the front leg and the rear leg can be adjusted according to the user's needs and the size of the UAV. The installation of these parts is shown in Fig. 1. Specifically, in responding

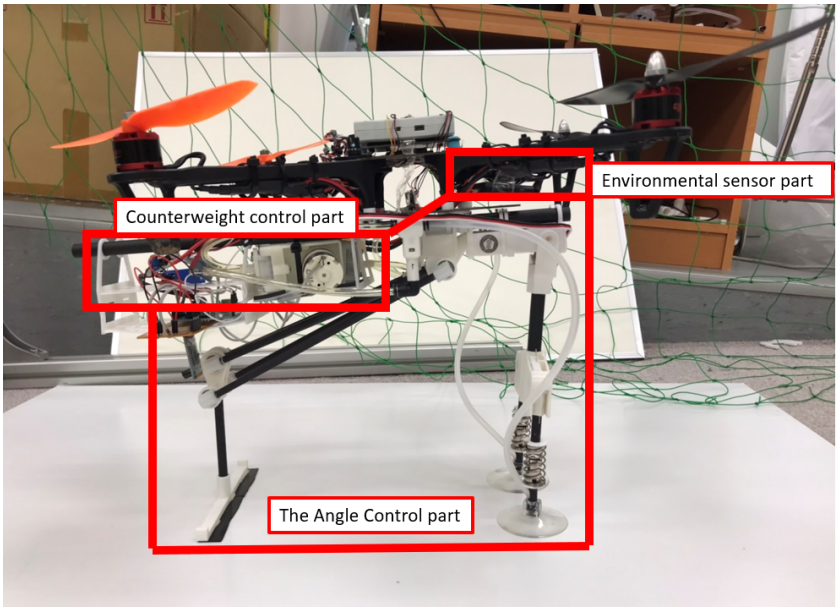


Figure 1: Installation positions of angle control, counterweight, and sensor parts.

to the mass distribution changes and the center of mass shifts, only the location of CCP needs to be adjusted backward and forward with respect to the servo motor position of the front leg prior to takeoff. This will allow the proposed landing gear to accommodate different-sized UAVs without any adverse impact on its landing capability. The type of UAV used is based on the DJI F450 frame with a brushless motor of 14.8V/9.5A-920rpm/V. It has a very limited payload of $1kg$. The motivation behind using the UAV with such a low payload is to enforce our design as minimum as possible so that it can be used by any type of UAV with a payload capacity of more than $1kg$. The proposed robotic landing gear weighs only $900g$. The robotic legs are designed to be attached to the surface with a vacuum system. The CCP is composed of the rather heavy and essential parts; a vacuum motor and a battery. Since the landing gear is mounted underneath the UAV, the weight reduction and the center of gravity change adjustment become particularly important. Depending on both payload and size of the UAV, the position of the CCP can be adjusted manually when the landing gear is being attached to the UAV in such a way that it keeps the center of gravity of the UAV with the landing gear as close as possible to the center of the original UAV. The ESP is the place to install various types of sensors for intended applications (*e.g.*, RGB-D camera, Lidar, Heading, IMU, and related others) in order to sense the target landing surface details, including the titling angle of the wall relative to the UAV approaching direction. We focus on the ACP and mechanical structure design of the landing gear. The ACP consists of a servo motor with $25kgf \cdot cm$ torque, two front robotic legs, and a non-slip leg. This design is inspired by the triangular landing gear designs used in commercial aircraft systems. Such triangular systems have been referred to as the most stable polygons. Two vacuum suction cups are mainly designed as a backup/emergency plan for any failure that might appear during landing and/or attaching to the surface. The CAD model of the prototype is shown in Fig. 2.

To better understand how the robotic landing gear can land on different shapes and inclinations of surfaces, we define the target surface in the world coordinate system (or the XYZ Cartesian axes) separately in the XY and XZ plane as illustrated in Fig. 3. While the robotic leg in the front accommodates the difference in surface shapes in the XZ plane, the front leg and the rear non-slip leg form a closed-chain structure by attaching to the surface in the XY plane.

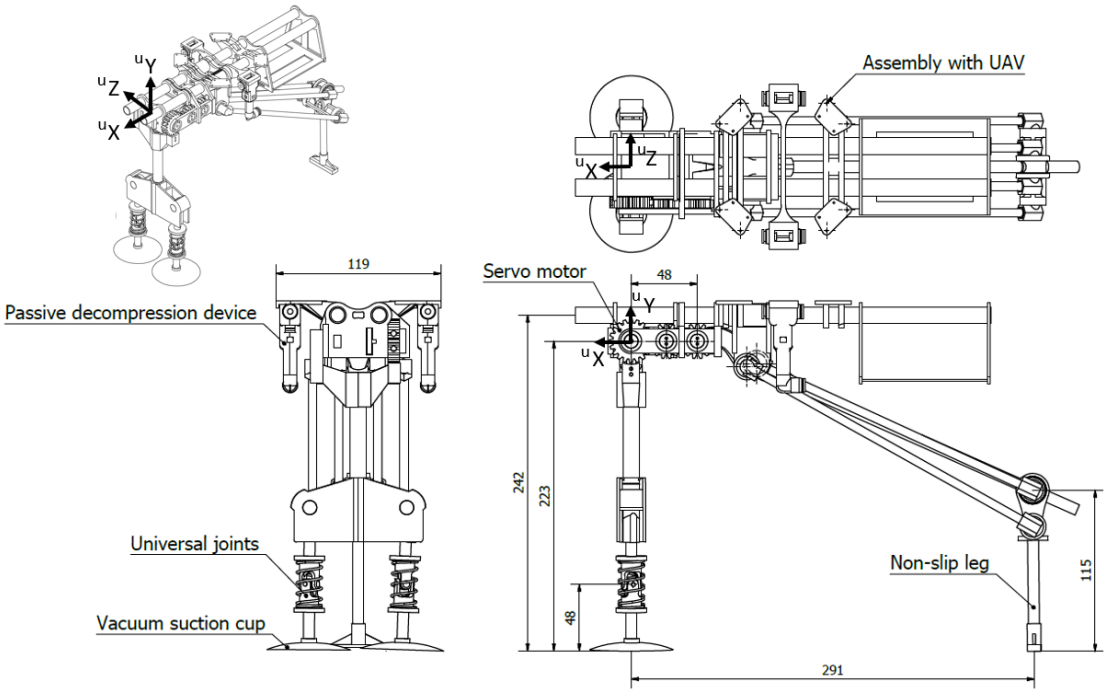


Figure 2: The CAD drawing of the robotic landing gear. A servo motor controls the angle between the front leg and ${}^{u}X$ axis of the UAV.

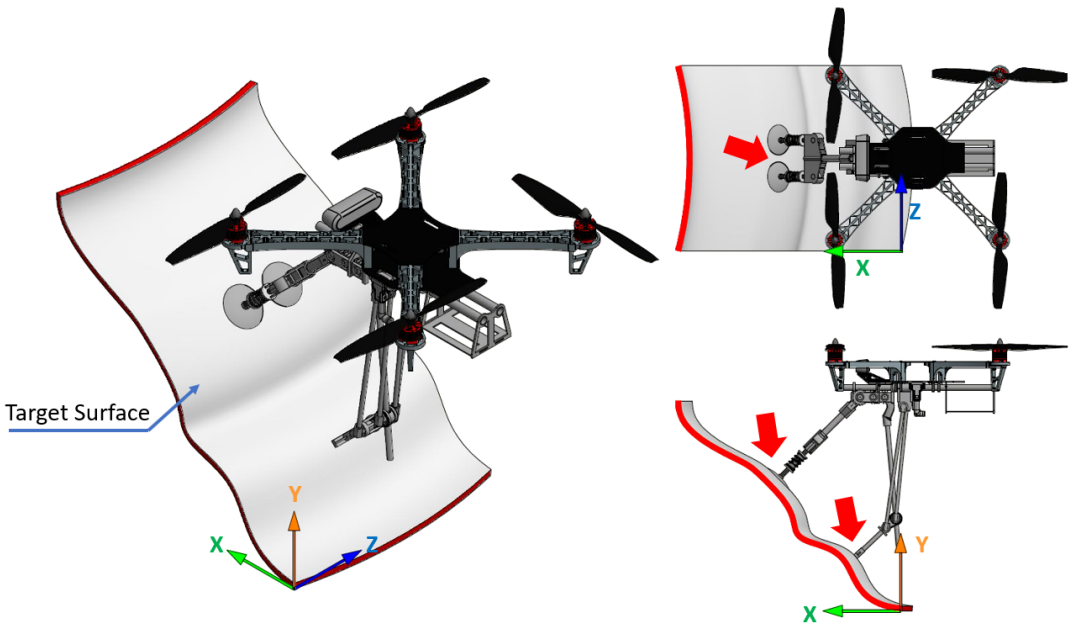


Figure 3: The projection of a 3D surface onto the XZ and XY planes. The varying surface curvature appears as a curved line, such as the red line in the figure, from which the shape of surface can be coarsely defined. Using this line, the landing gear is set up to its landing configuration.

2.1 Design of the Robotic Legs

Each robotic leg is made up of a vacuum suction cup, a set of universal joints, a compression spring, and a vacuum tube. For different angles of inclination of landing surface, robotic legs work passively on adjusting the vacuum suction cup's angle when contacting the surface as shown in Fig. 4 automatically.

Since the robotic landing gear uses a vacuum system to keep the UAV attached to the surface, the vacuum suction cup must be perpendicular to the target surface to maximize the adhesive force. In order to adapt to the different shapes of the surface, this design allows the robotic leg to be kept perpendicular adaptively to the surface and attached to it. Moreover, it can also correct slight angle errors when landing. In terms of structural design, the universal joint can provide the vacuum suction cup with about 45 degrees of steering in all directions. The steering degree of the universal joint is based on the shape of the target landing surface and the velocity of the UAV when approaching the target surface. The greater the velocity, the greater the rotation angle of the universal joint due to the compression spring used. The analysis of the usable steering angle of the actual universal joint will be presented in the experimental result section. When the UAV needs to leave the target surface, the compression spring will return the vacuum suction cup to the original position for the next landing.

The two robotic legs in the front are separated into two sides using a torsion spring to keep them parallel. In this way we can enable the robotic landing gear to land on different shapes of surface and achieve the cushioning effect by absorbing the shock during UAV landing. Even though the surface is not purely planar in the XZ plane, the torsion spring and the universal joint are flexible enough to accommodate different shapes of the surface. Some examples are shown in Fig. 5.

The mechanical structure of the front leg and the rear leg adapt to the surface in the XY plane. Since a single motor controls the angles of the front and rear leg configurations in ACP, in order to keep both legs parallel to the landing surface, we need to determine a gear ratio to make the non-slip rear leg rotate more than the front leg. Using the linkage method of multiple rods, the non-slip leg can change the angle and length with only a single motor input simultaneously. This design can make the front and rear legs have different angle changes, enabling them to adapt to the surface of different curvatures. Since the robotic landing gear uses a vacuum system to keep the UAV attached to the surface, it is important to ensure that the legs of the landing gear are perpendicular to the target surface. We define the ground plane as an angle of 0° surface, and a vertical plane perpendicular to the ground as an angle of 90° surface. Our novel design structure can make the landing gear land between 0° and 100°. Regardless of any angle of inclination of the target surface, it can keep the 3 legs perpendicular to the target surface.

We assume that the angle of inclination of the target surface is α, Θ_1 is the angle controlled by the servo motor, Θ_2 is the angular position of the rear leg, and Θ_3 is the rotation angle of the rear leg to the surface. For the cases where $\Theta_1 \in (0, \pi/2]$, we can make use of triangular representations as shown in Fig. 6. Then the angular relationship between the target surface and the servo motor (front leg) is given by

$$\alpha + \Theta_1 = \frac{\pi}{2}. \tag{1}$$

The front leg and the rear leg are driven by a gear with a ratio of $1 : 1.28$. This ratio is derived from the data obtained from simulations in the AutoDesk Inventor environment by modeling a linear relationship between Θ_2 and Θ_1. Therefore, the angle relationship between the front and the back leg is described by

$$\dot{\Theta}_2 = 1.28 \times \dot{\Theta}_1. \tag{2}$$

From the entire structure of the polygon with $n = 5$ sides, by using the formula for the sum of the interior angles $(n-2) \times \pi$, the relationship between the Θ_3 and other angles can be written as follows:

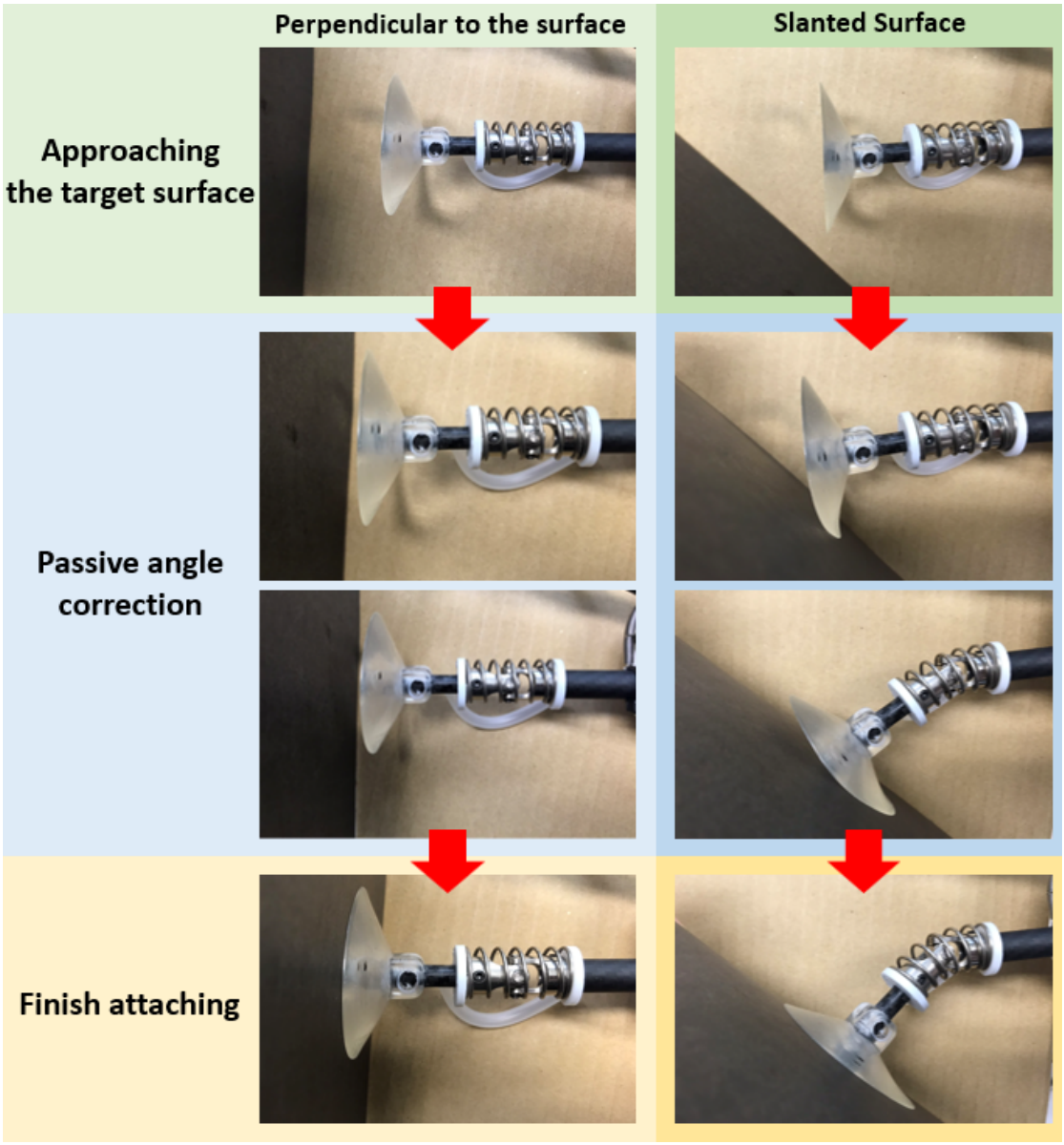


Figure 4: Attaching to the target surface: When the soft rubber part of the vacuum suction cup touches the landing surface, the friction between the rubber and the surface will steer the universal joint, keeping the vacuum suction cup perpendicular to the surface.

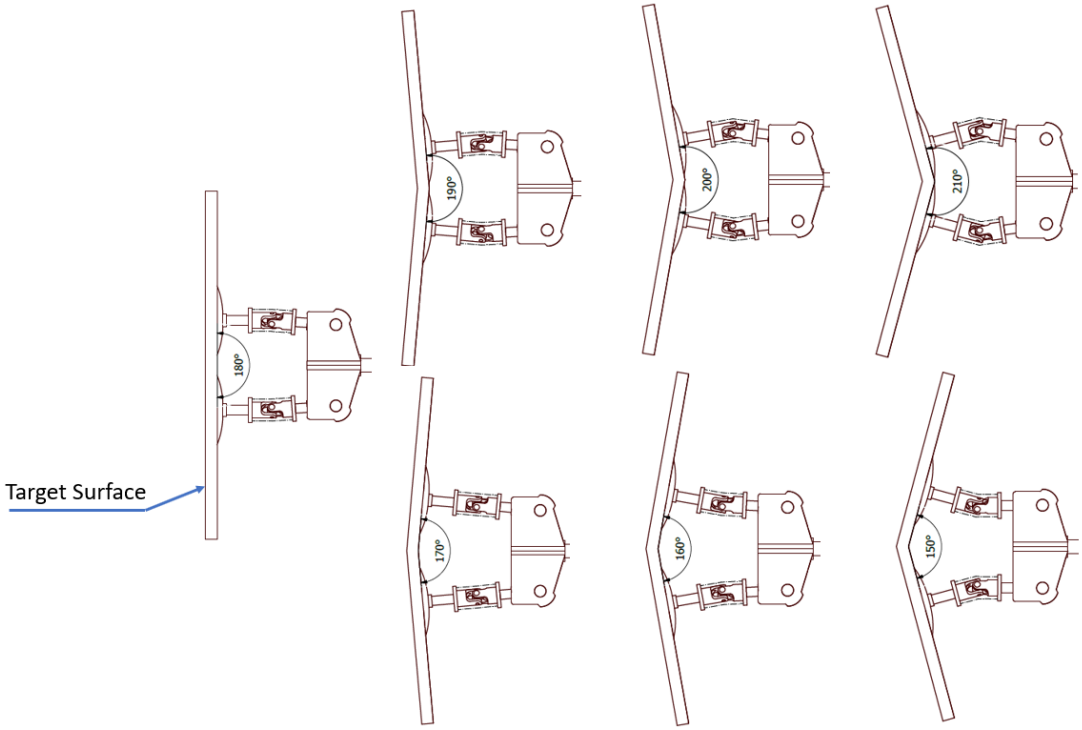


Figure 5: The two front robotic legs are flexible enough to accommodate any shape in the XZ plane. The back leg is only to support the entire landing gear, therefore it will not affect the difference in the XZ plane.

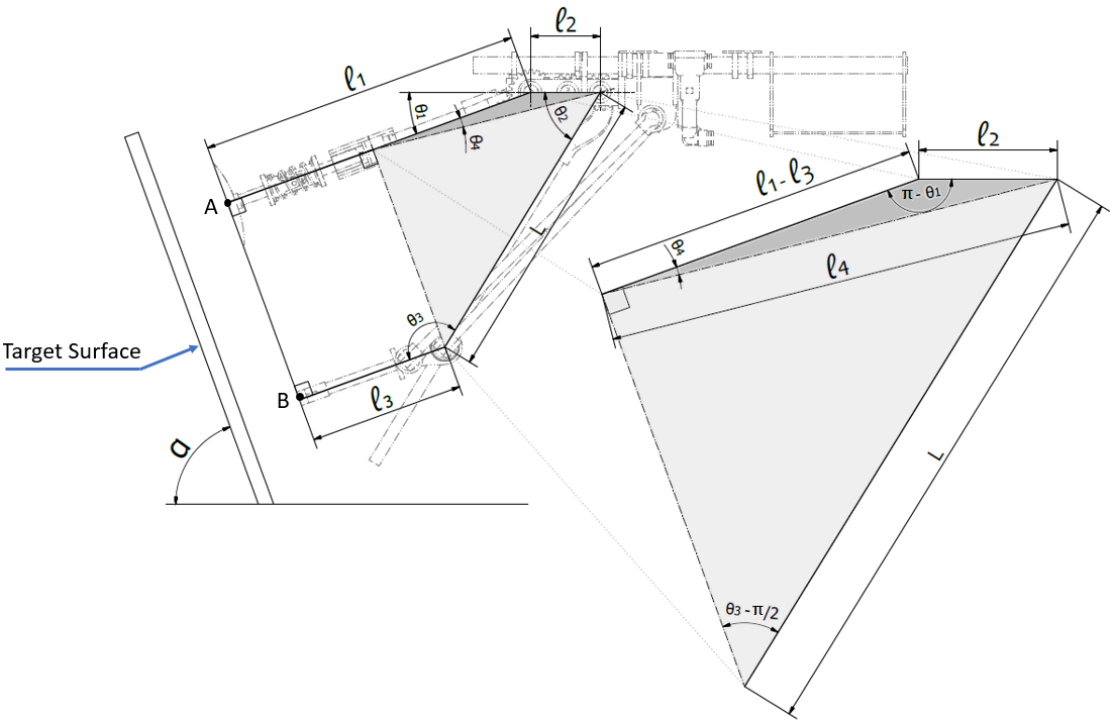


Figure 6: This figure is the setup of landing gear for an example of the $\alpha = 70$ degree target plane. It shows the definition and position of each angle within the main structure.

$$\begin{aligned} \frac{\pi}{2} + \frac{\pi}{2} + \Theta_3 + \Theta_2 + \pi - \Theta_1 &= (5-2) \times \pi \\ \Theta_3 + \Theta_2 - \Theta_1 &= \pi \end{aligned} \tag{3}$$

With different surface inclination angles, the angle Θ_1 (thus, Θ_2 and Θ_3) and length L will change at the same time accordingly. To calculate the length L, we can first calculate ℓ_4 using the law of cosines in the dark gray triangle in Fig. 6.

$$\begin{aligned} \ell_4^2 &= (\ell_1 - \ell_3)^2 + \ell_2^2 - 2\ell_2(\ell_1 - \ell_3)\cos(\pi - \Theta_1) \\ &= (\ell_1 - \ell_3)^2 + \ell_2^2 + 2\ell_2(\ell_1 - \ell_3)\cos(\Theta_1) \end{aligned} \tag{4}$$

where ℓ_1, ℓ_2, and ℓ_3 are leg segment lengths. After computing ℓ_4, by using the law of sines, Θ_4 can be calculated as follows:

$$\begin{aligned} \frac{\ell_2}{\sin\Theta_4} &= \frac{\ell_4}{\sin(\pi - \Theta_1)} \\ \Theta_4 &= \sin^{-1}\left(\frac{\ell_2 \sin\Theta_1}{\ell_4}\right) \end{aligned} \tag{5}$$

After calculating ℓ_4 and Θ_4, from the bigger triangle (colored light gray in Fig. 6), the length L can be calculated using the law of sines given by

$$\begin{aligned} \frac{L}{\sin(\frac{\pi}{2} - \Theta_4)} &= \frac{\ell_4}{\sin(\Theta_3 - \frac{\pi}{2})}, \\ L &= \frac{\ell_4 \cos\Theta_4}{-\cos\Theta_3}. \end{aligned} \tag{6}$$

With this design, for the surface in the XY plane to which the front and rear leg are positioned, any shape of the surface can be regarded as a plane with a different angle of inclination. The angle α with the ground plane can be calculated via a line segment connecting the landing points. Then Θ_1 can be set in such a way as to keep the front and rear leg parallel to the landing surface. The universal joint at the front end of the robotic leg can passively perform a slight angle correction, according to the actual shape of the surface at the landing point. However, instead of using a robotic leg with a vacuum suction cup, the rear leg is designed to be a non-slip leg. From our experiments, it was noticed that when all the legs in contact with the target surface are equipped with universal joints, the structural rigidity is seriously insufficient and it could cause the landing gear to fall off the target surface easily. It would also cause the UAV to be unable to land in a horizontal attitude after the power is turned off. Also, when leaving the target surface, it could lead to a control loss when the UAV propellers start to rotate again. It also limits the types of target surfaces. For discontinuous surfaces, the success rate of landing would decrease due to insufficient rigidity. The advantages of designing the rear leg with a non-slip foot are not only to solve the aforementioned issues but also not to affect the adsorption force of the vacuum system on the target surface. It has also a better landing effect for discontinuous surfaces.

2.2 Design of the Passive Decompression Device

When the UAV needs to leave the surface, the vacuum system needs to be decompressed to release the vacuum environment so that the UAV can take off smoothly. For this reason, we design a passive decompression device as shown in Fig. 7. Compared to the electronic decompression device, the passive decompression device is lighter and less power-demanding, while preserving the necessary functionality. This passive decompression device is composed of three parts: the air intake part, the outtake part, and air disperse housing. The air intake part is connected to the air disperse housing, the vacuum suction cup, and the air inlet part of the vacuum motor.

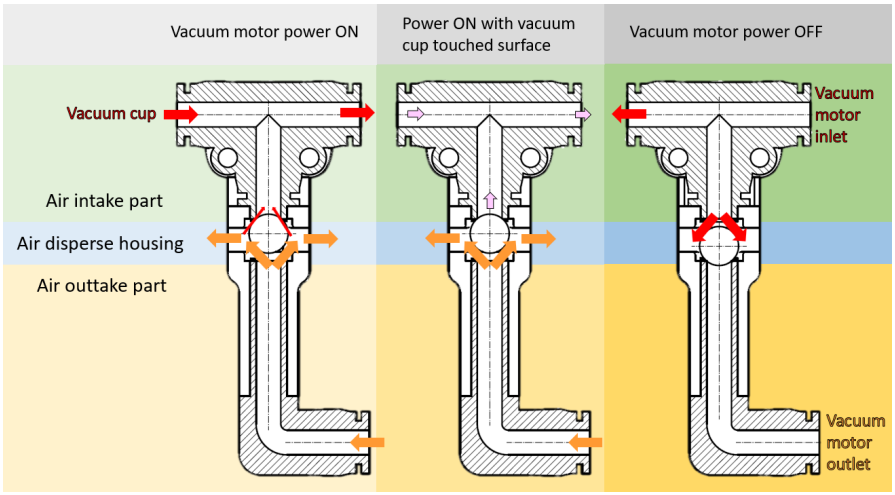


Figure 7: When the vacuum motor is turned on, the small steel ball is blown upward by the air from the outtake part and blocks the intake part. Since this air intake part is connected in parallel with the vacuum suction cup, the small steel ball will not be fully blocking the air intake part until the vacuum suction cup is attached to the surface.

The air outtake part is connected to the air disperse housing and the air outlet part of the vacuum motor. The air disperse housing connects the intake part and outtake part, and has a small steel ball in the center.

The outtake part will blow the small steel ball upward when the vacuum motor is turned on. Since the vacuum suction cup has not been attached to the target surface, the intake part will not become in a negative pressure state. Therefore, the part sucking the small steel ball has almost no suction for the small steel ball. This is because the vacuum suction cup and the sucking part are connected in parallel within the intake part. Therefore, until the vacuum suction cup is attached to the target surface, it will increase the suction power of the part that sucks the small steel ball and blocks it. And the intake part will be in negative pressure state to complete the suction step. When the UAV needs to leave the surface, it is only necessary to stop the vacuum motor. The small steel ball will quickly leave the air intake part, opening (the previously closed) air intake part. By doing so, the vacuum suction cup can be removed from the surface smoothly and quickly. While a small vacuum solenoid valve weighs around 50 grams, this passive pressure reduction unit weighs only 9 grams. Since every extra weight means more power consumption for the motor of UAVs, we opted to use a passive pressure-reducing device in order to keep the proposed robot landing gear as light as possible. The passive pressure-reducing device not only requires no additional control but also is lighter.

2.3 The Operational Principle of Robotic Landing Gear

A brief operation process is given in Fig. 8. Right after takeoff, the ACP rotates the robotic legs up and makes them face front (the direction of flight), and keeps them horizontal. When approaching the target surface, the 3D shape of the surface is detected using the data provided by the sensor part, and set up the robotic legs adjustments accordingly to start the vacuum system and be ready to approach the target surface. The decision of whether the robotic legs are fully attached to the surface or not is determined via the vacuum value. When taking off the surface again, the vacuum system is turned off and the power of the UAV is throttled to leave the surface.

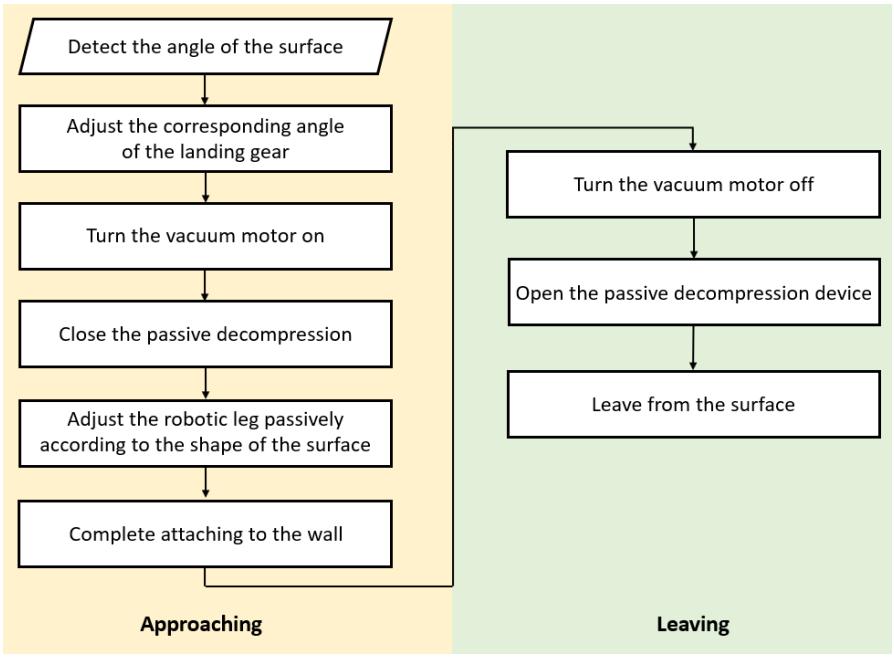


Figure 8: A brief operation process of the landing gear is divided into two parts: approaching and leaving. After leaving the current surface, return to the approaching part for the next target surface landing.

3 Enhancements for Feature-based Image Matching Pipeline

Most of the complicated and sophisticated vision-based robotic tasks usually build upon the image-matching step as matching images of the same scene can provide important information (e.g., camera motion). Image matching is generally done via extracting and matching some distinctive points via their feature vectors. This procedure generates some mismatched points due to imperfections. Mismatched points are called outliers and are identified via probabilistic methods. Since the probabilistic methods work in an iterative manner, they generally occupy a large portion of the computational cost of the whole image-matching pipeline. In this paper, we present a simple yet efficient algorithm that is employed for eliminating the outliers aiming at reducing the total number of iterations needed in the probabilistic methods. Our method is motivated by the common way of visualizing the established matches among images. We tile images together and search for the parallel lines connecting correspondences. We present extensive computational and comparative experiments using simulated data involving real images.

Feature-based image matching starts with detecting some salient points (regarded as features) in images. These feature points are represented with a vector of scalars obtained using pixel values in their neighborhood (*e.g.,* set of histograms of orientation gradients) and this vector is called a descriptor and this process is called feature description. Descriptors are matched by comparing the Euclidean distances between descriptor vectors. During this matching procedure, some feature points usually are not correctly matched. These mismatches are referred to as outliers. Probabilistic methods (*e.g.*, Random sample consensus (RANSAC) [5], Least median of squares regression (LMeds) [18] and similar others) have been employed in order to remove outliers and compute the transformation (or motion). Over the years, there have been several improvements proposed and presented for RANSAC, which is based on random sampling and using a threshold to identify inliers and outliers, in two directions namely, improving sampling methods and automatic threshold selection. PROSAC (Progressive Sample Consensus) [3] was proposed with an enhanced sampling algorithm based on ranked features according to similarity scores of their descriptors and samples were drawn through their ranking. In terms of decisive threshold selection for inlier-outlier separation, *automatic* RANSAC methods have been proposed [13, 4, 17]. Although they have improved the overall performance, due to their iterative

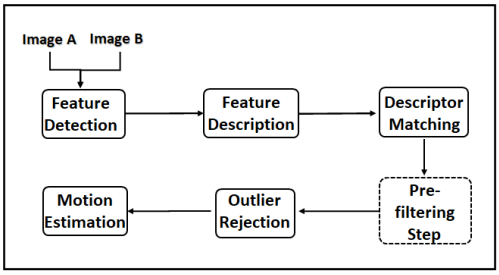


Figure 9: Image Matching Pipeline with the proposed pre-filtering step.

procedure nature, their computational cost can be still high.

3.1 Pre-Filtering Step for Outlier Reduction Method

Correspondences are usually visualized via tiling overlapping images together and drawing a line connecting each correspondence. Inliers mostly form a group of parallel lines in such visualization since differences in slope are preserved [2, 9] based on the geometrical invariants (specifically for certain type(s) of transformations). Motivated by this, we present a pre-filtering step to remove the outliers before applying a robust estimation method for outlier rejection and motion estimation. Our proposed approach applies a translation transformation to the feature point positions of the second image so that their positions with respect to the first image coordinate frame are obtained when they are tiled together. Then we compute the angles of the lines connecting correspondences. Once angles are computed, we find the peak group in the histogram created using the angle values. The lines, thus feature points forming those lines are considered as inliers and kept for further processing via robust estimation methods. An algorithmic representation of the proposed pre-filtering step is provided in Alg. 1 while the image matching pipeline with embedded our proposal step is illustrated in Fig. 9.

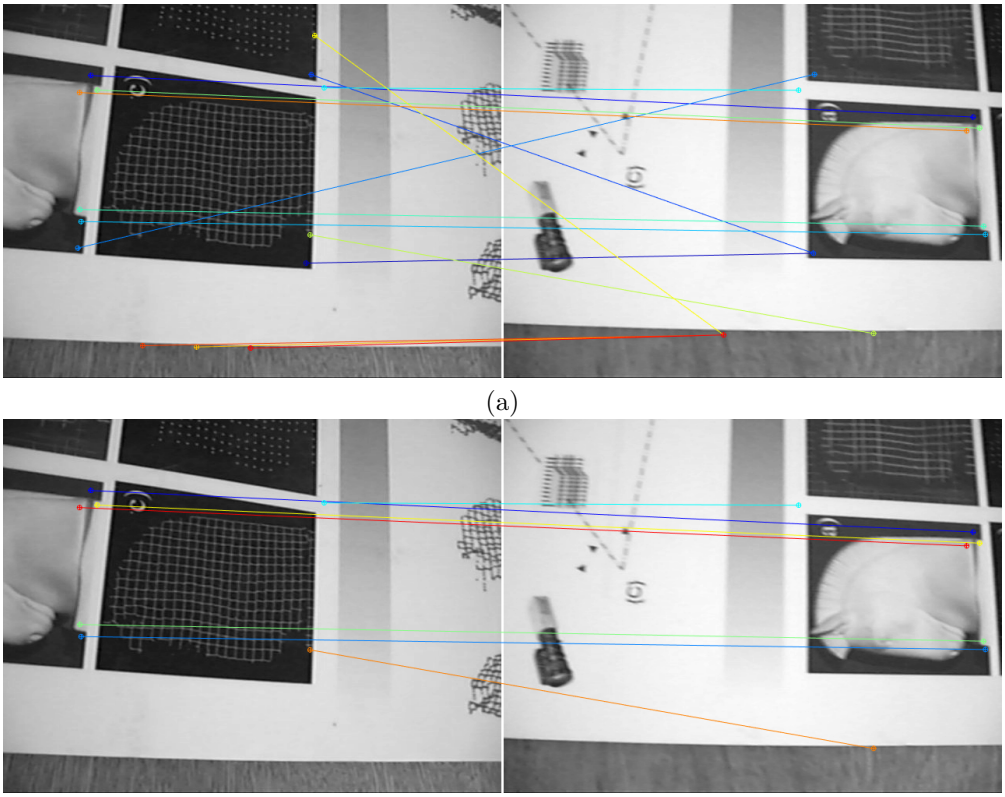
Algorithm 1: Algorithm for filtering correspondences via angle grouping

Input: Matched feature positions in the local image coordinate frames
$\mathbf{P} = (x_i, y_i) \quad i = 1, 2, \cdots, n$ and $\mathbf{M} = (x_i, y_i) \quad i = 1, 2, \cdots, n$,
Image size (u, v),
Bin-width for angle grouping, w
Output: a set of correspondences indices to be kept, Ind
foreach *feature correspondences* p *and* m **do**
    A1 $\leftarrow$ Compute the slope using $p = (x_p, y_p)$ and translated coordinate $m = (x_m + u, y_m)$
    A2 $\leftarrow$ Compute the slope using $p = (x_p, y_p)$ and translated coordinate $m = (x_m, y_m + v)$
Compute histograms over values in **A1** and **A2** using bin-width w
Keep correspondence indices in the bin width the maximum number of elements
$Ind \leftarrow$ Unify indices coming from two histograms

An example of applying the proposed filtering step tiling horizontally and vertically can be seen in Figs. 10 and 11. Before applying the filtering step, a total of 14 correspondences were established and only 5 inliers. After applying the filtering and unification step, a total of 7 correspondences remained including 5 inliers and 2 outliers.

3.2 Extension of Pre-Filtering Step for Multiple Motion Estimation

The objects in the scene can be moving, either rigidly or non-rigidly independent of the camera motion. This fact can cause a generation of multiple motions in two overlapping images. Therefore, we studied extending our pre-filtering step to be used in such cases. Our extension is based



(a)

(b)

Figure 10: (a) Established Correspondences. A total of 14 correspondences and only 5 inliers. (b) Remaining Correspondences after applying the filtering step. A total of 7 correspondences including 2 outlier and 5 inliers.

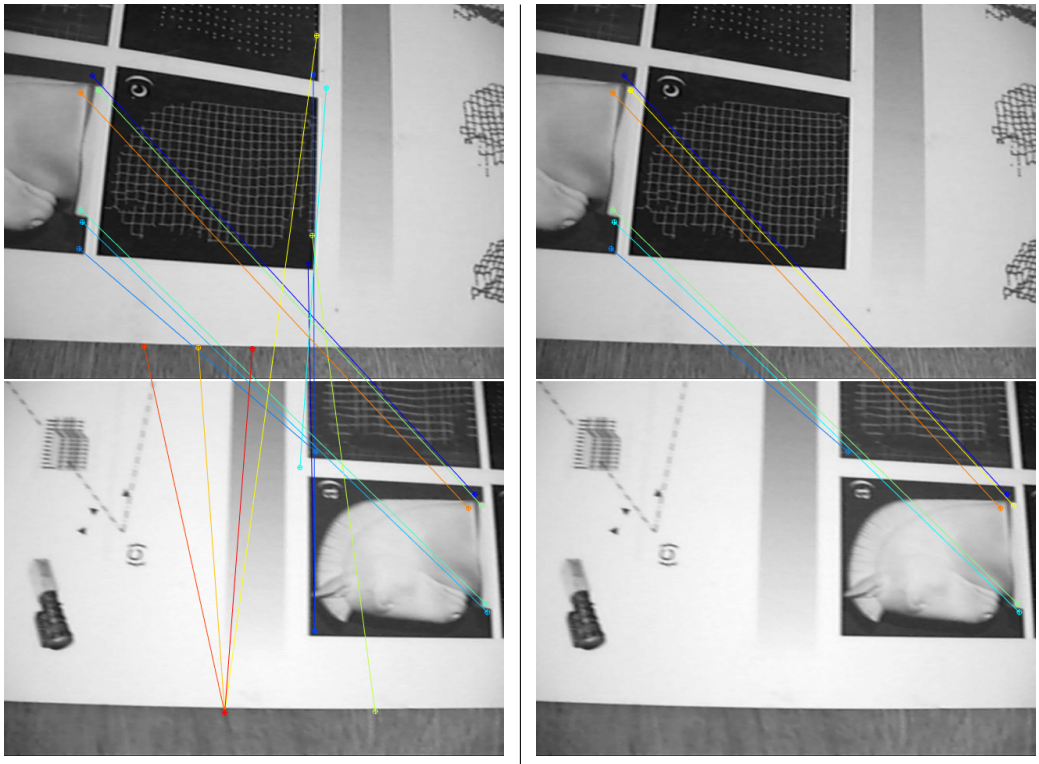
on investigating the histogram bins of slopes of lines generated by positions of correspondences. We find the matching correspondences that are in different histogram bins and compute the motion. The computed motions can be investigated further in order to understand and verify the cause of the motion such as moving object(s), points lying on different planes, and similar others. A combined algorithmic representation of the proposed multiple motion estimation extension of the pre-filtering step is provided in Alg. 2

4 Experiments

4.1 Experimental Results for Novel Robotic Landing Design

We carried out a preliminary landing experiment with an in-house built terrain generation system as shown in Fig. 12. This testbed allows the creation of different undulating surfaces by adjusting the angle of the landing plane. We also adjust test surfaces by calculating the distance between the front and rear legs. The limit curvature radius (R) of the surface can be calculated by using the angles (Θ_{FR}, Θ_{FL}, Θ_{Back}) and the distance (D).

During the experiments, we attached the robotic landing gear to a custom-built quadcopter based on a commercially available DJI F450. After experimenting with various landing trials (10 for each angle combination), the success rate of the robotic landing gear for different types of surfaces is presented in Table 1. The first two columns denote the angle configurations used in the landing terrain testbed, referring to Fig. 12, while the third column represents the contact area between the non-slip leg and the surface being tested. As a result of experimental tests, the landing limits for the robot were within 50° of the tangent point of the irregular surface and $R \geq 200mm$ obtained by the geometrical relation $D = 2 \times R \times sin(\Theta)$. D changes depending on



(a) (b)

Figure 11: (a) Established correspondences. A total of 14 correspondences and only 5 inliers. (b) Remaining correspondences after applying the filtering step. A total of 6 correspondences including 1 outlier and 5 inliers.

Algorithm 2: Algorithm for computing multiple motions

Input: Matched feature positions in the local image coordinate frames
$\mathbf{P} = (x_i, y_i) \quad i = 1, 2, \cdots, n$ and $\mathbf{M} = (x_i, y_i) \quad i = 1, 2, \cdots, n$,
Image size (u, v),
Bin-width for angle grouping, w
Output: a set of motions, **Hlist**
foreach *feature correspondences p and m* **do**
    A1 $\leftarrow$ Compute the slope using $p = (x_p, y_p)$ and translated coordinate $m = (x_m + u, y_m)$
    A2 $\leftarrow$ Compute the slope using $p = (x_p, y_p)$ and translated coordinate $m = (x_m, y_m + v)$
Compute histograms over values in **A1** and **A2** using bin-width w
foreach *bin* b_{a1} *with more than* 5 *correspondences* **do**
    foreach *bin* b_{a2} *with more than* 5 *correspondences* **do**
        $\mathbf{c_1} \leftarrow$ Find common correspondences between b_{a1} and b_{a2}
        H $\leftarrow$ Compute the motion using $\mathbf{c_1}$
        Remove $\mathbf{c_1}$ from the set of correspondences.
        Hlist $\leftarrow$ Save **H**.

the UAV size and also the small variation caused by passive joints (see Fig. 15). In our case, D was approximately $300mm$. During the experiments, failed landing attempts have been observed when the tilt angles of both the front and back planes are very large and in the same direction. The reason is the lack of enough friction force generated by the vacuum cup material. That leads to making it difficult to guide the universal joint when it touches the landing surface. It

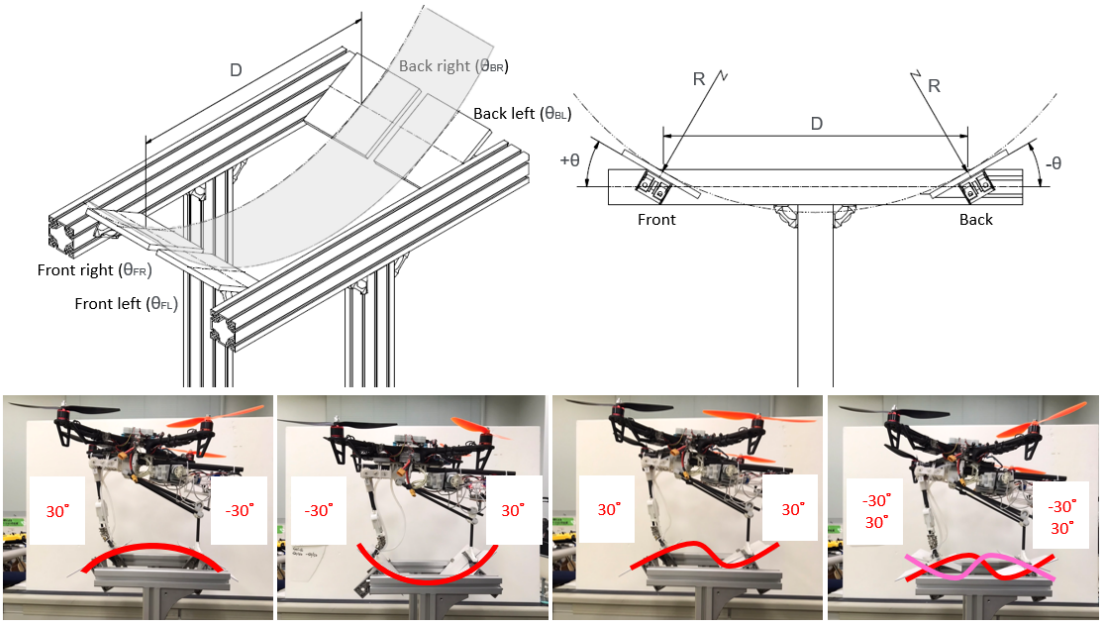


Figure 12: The uneven landing terrain testbed to simulate different types of surfaces by changing the surface undulation angle. We define the angle change in the clockwise direction as positive and counterclockwise as negative. A sample landing flight scenario can be seen at `https://youtu.be/fbsFBl-dzFs`.

was observed from the experiments that the rear non-slip leg design was able to land successfully regardless of the landing surface angle variations and unevenness of the landing surface.

We then proceed to the main experiments covering the entire flight and landing process. In our experiments, we use 6 different types of surfaces with smooth material (*e.g.*, plastics and metals) as shown in Fig. 13, including a vertical surface, a 45-degree slope surface, a discontinuous surface with elevation differences, and 3 types of curved surfaces with various radii of curvature, to test whether the robotic landing gear can successfully land on uneven surfaces. The whole operation process is divided into several steps: taking off, setting robotic legs to the initial position (defined by $\Theta_1 = 0$), approaching the target surface, contacting the target surface, and decreasing power. The overall pipeline of the experiment is given in Fig. 14.

We measured the Θ_3 and L distance during the landing experiments on different angled surfaces. We also computed the same values using Eqs. (1)-(6) and obtained values are given in Fig. 15. Some small error can be seen between values and this is mainly the right angle assumption between legs and the surface contact points. These errors are mostly compensated by the universal joints of front legs, demonstrating their importance in the proposed design. Since universal joints can compensate for some deviations from the right angle, the legs are not necessarily perpendicular to the surface all the time.

In order to make the universal joint automatically return to its original position, we use springs to perform the task of passive return. Among the ones we tested, we found that Ø1.4x19x55 and Ø1.2x20x40 were too soft, while Ø1.6x22x45 was too stiff to serve the intended purpose. We also tested torsion springs that did not provide a successful outcome in all angles and a set of 3 tension springs that were also failed. We use Ø1.4x19x43 that were empirically found to be the best for our purpose and design.

When robotic legs are facing front while flying, the rotation speed of the landing gear is likely to affect the stability of the UAV. We did different experiments with rotating speed of the servo motor. In the first experiment, when the servo motor was set to rotate 95° in 3 seconds, the UAV has lost its balance and fell down. When the selected rotation time is about 5 seconds, the balance was still adversely affected, and the UAV suffered a short-term control failure. However, it did not fall down. When the selected rotation time is 10 seconds, the balance was affected

Table 1: Success rate experimental trials for different type of surface

$\Theta_{FL} = \Theta_{FR}$ in degrees	$\Theta_{BL} = \Theta_{BR}$ in degrees	Contact area of non-slip leg	Success rate
0	0	full	100%
10	10	full	100%
20	20	full	100%
30	30	full	100%
40	40	full	70%
50	50	full	50%
10	-10	full	100%
20	-20	full	100%
30	-30	full	100%
40	-40	full	80%
50	-50	full	70%
-10	10	full	100%
-20	20	full	100%
-30	30	full	100%
-40	40	full	80%
-50	50	full	80%
-10	-10	full	100%
-20	-20	full	100%
-30	-30	full	100%
-40	-40	full	60%
-50	-50	full	50%

Θ_{FL}	Θ_{FR}	Θ_{BL}	Θ_{BR}	Contact area of non-slip leg	Success rate
-10	10	-10	10	full	100%
-20	20	-20	20	full	100%
-30	30	-30	30	full	100%
-40	40	-40	40	full	60%
-50	50	-50	50	full	70%
-10	10	-10	10	half	100%
-20	20	-20	20	half	100%
-30	30	-30	30	half	100%
-40	40	-40	40	half	60%
-50	50	-50	50	half	60%

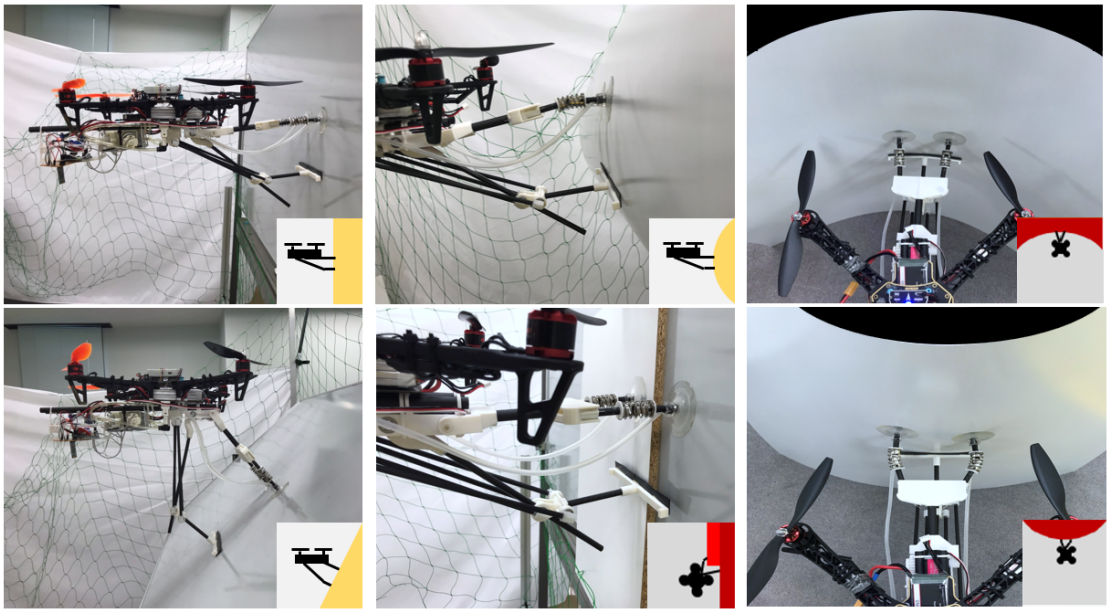


Figure 13: Some examples of landing on arbitrarily shaped surfaces. The landing gear can easily connect to a variety of surfaces with different curvatures through passive universal joints.

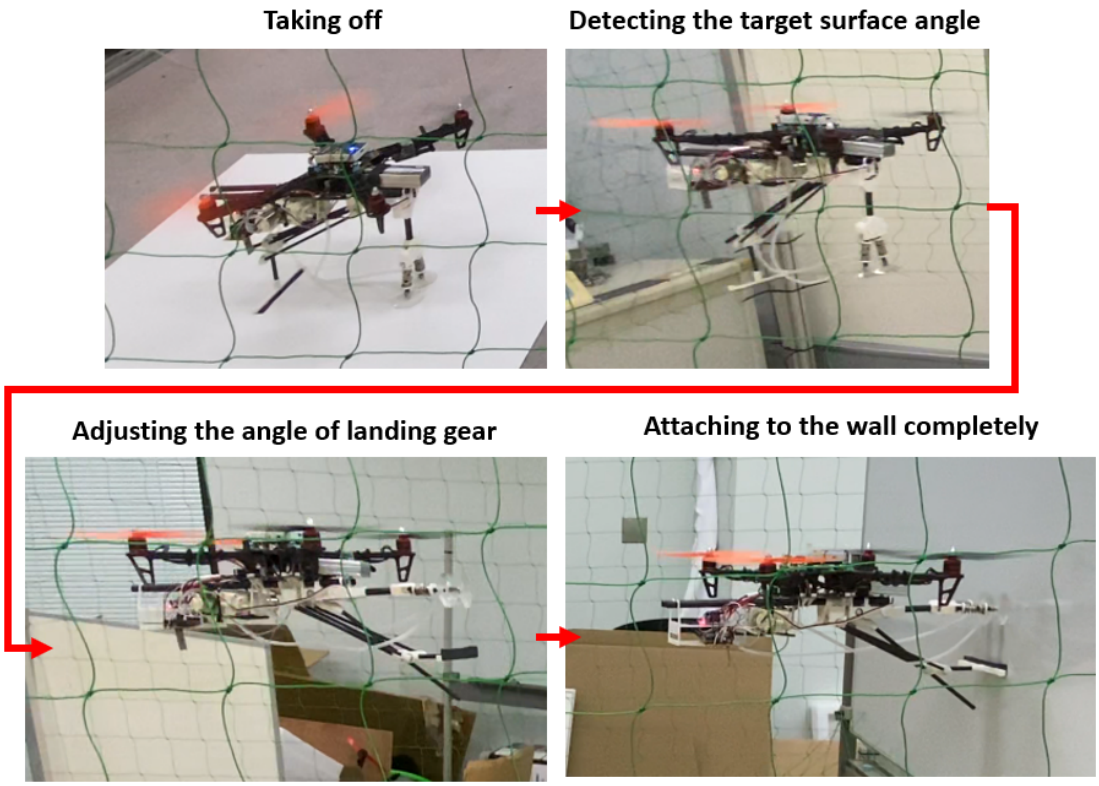


Figure 14: The figure shows the operational principle of robotic landing gear. According to the calculation of the center of gravity of the entire UAV system, given the counterweight position pre-adjusted while the robotic landing gear was being attached to the UAV, the angle of the robotic legs changes during the flight without affecting the flight stability and balance control. A sample flight scenario can be seen at `https://youtu.be/4HtYF2wJ1NQ`.

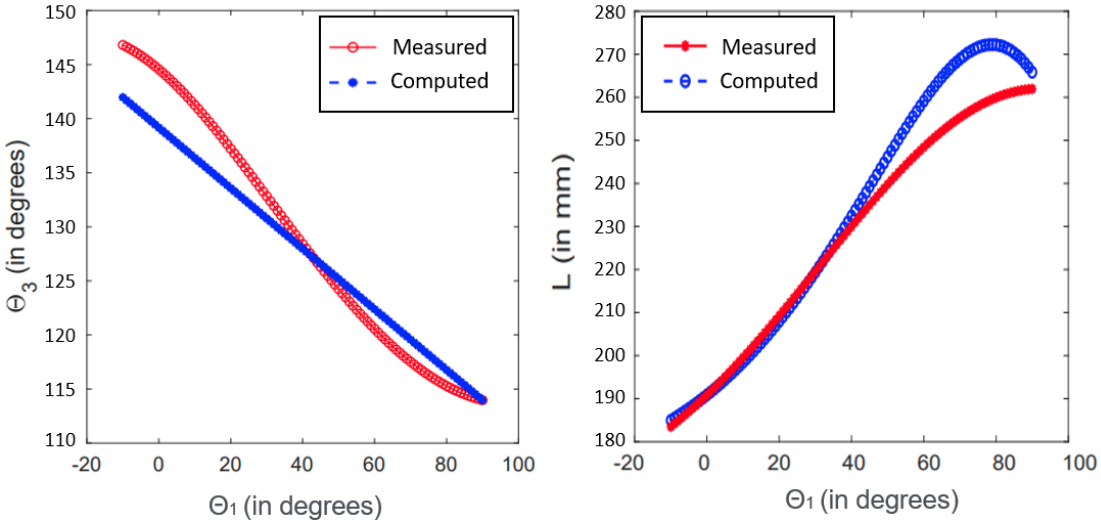


Figure 15: These graphs denote computed values using Eqs. (1)-(6) and measured values during experiments for Θ_3 and L over Θ_1. Small discrepancy between values are mainly the right angle assumption between legs and the surface (points A and B in Fig. 6) since this assumption does not hold always due to the universal joint.

slightly, but it did not affect the flight.

In order to test the landing ability on arbitrary surfaces, we set the landing surface angle in view of XZ plane between $[150°, 210°]$ with 5-degree intervals in the XZ plane simulating the cases depicted in Fig. 6. We performed 10 landing trials for each angle. We observed that the larger the angle, the more accurate speed and angle of the UAV control are required. The UAV control became very difficult when the angle is within the intervals $[150°, 160°]$ and $[195°, 210°]$. For the rest of the angles tested, the UAV has attached to the target surface easily. According to experiments, the front leg is designed for different angles and has an adaptive correction capability of about $\pm 20°$ of the target surface. The angle change greater than $\pm 20°$ from $180°$ will increase the difficulty of UAV control, therefore such a condition is normally not suitable for the proposed landing gear design.

4.2 Experimental Results for Enhancements on Image Matching

We have tested our proposed pre-filtering step via extensive simulations using 35 different homography matrices used in [12] covering most of the challenging situations. We used SURF [1] to extract features from an image and used homographies to generate their correspondences as ground truth. From this ground-truth data, for each simulation parameter summarized in Table 2, we randomly generated a set of correspondences that is composed of both inliers and outliers correspondingly and run M-estimator Sample Consensus (MSAC) [20] with the proposed pre-filtering step and without it. Moreover, we also corrupted correspondences positions by adding a zero-mean noise with different levels of standard deviations. If the mean error computed as in Eq. 7 is less than the distance threshold of 5 pixels, the estimated homography is considered correct and such a trial is counted as successful. The error is computed as a mean of distance between the total number (n) of correspondences (($\mathbf{p}, \mathbf{m}$)) when they are mapped with the estimated homography $\mathbf{H}$.

$$\mu = \frac{\sum_{i=1}^{n} ||\mathbf{p_i} - \mathbf{H} \times \mathbf{m_i}||_2}{n} \tag{7}$$

A threshold of 5 pixel is determined as the maximum error obtained by using ground-truth transformations with noise corrupted correspondences during our simulations. For each homography, we have $3 \times 5 \times 5$ different simulation parameter configurations and we repeated each configuration $1,000$ times leading to a total of $2,625,000 (35 \times 75 \times 1,000)$ trials.

Table 2: Parameters used in Experiments

Number of Transformations	35
Total Number of Correspondences (inliers + outliers)	$[100, 250, 500]$
Outlier Ratios	$[0.5, 0.6, 0.7, 0.8, 0.9]$
Noise Standard Deviations	$[0.0, 0.5, 1.0, 1.5, 2.0]$
Maximum Number of Iterations in RANSAC	$2,500$
Distance Threshold	5 pixels

We present the obtained results in the format of the confusion matrix in Table 3. In total, the proposed pre-filtering step was able to improve the result by approximately 10%. Moreover, using the pre-filtering step helped to reduce the number of iterations needed during robust estimation. This is mainly due to the fact that it reduces the total number of correspondences while increasing the inlier ratio and both are favorable for reducing the number of iterations. Statistically summarized results are given in Table 4.

Table 3: Summary of Obtained Results in Confusion Matrix Format

		Without Pre-filtering		
		Unsuccessful	Successful	Total
With Pre-filtering	Unsuccessful	0.066 (173,987)	0.038 (101,042)	0.105 (275,029)
	Successful	0.137 (359,405)	0.758 (1,990,566)	0.895 (2,349,971)
	Total	0.203 (533,392)	0.797 (2,091,608)	1.000 (2,625,000)

Since our pre-filtering step relies on counting parallel lines connecting correspondences when they are tiled together, the total number of inliers plays a more important role than the inlier ratio. In other words, our proposed filtering step performs better in the case of an outlier ratio of 0.9 and the total number of correspondences of 500 than in the case of 100 total number of correspondences with the same outlier ratio. For the total cases $(275,029)$ where our filtering step failed to provide accurate homography, we plotted the histogram for each tested number of correspondences with respect to inlier ratios used which is depicted in Fig. 16. As can be seen, the total number of failure cases using a total number of 100 correspondences is larger than 250 and 500 for each inlier (or outlier) ratio used in experimental tests.

Table 4: Number of Iterations in Robust Estimation

	min.	max.	mean	std.
With (overall)	3	2,500	644.88	908.69
Without (overall)	57	2,500	1,301.31	1061.55
With (succ.)	3	2,500	440.99	709.54
With (unsucc.)	21	2,500	2,386.97	434.94
Without (succ.)	57	2,500	995.79	977.06
Without (unsucc.)	219	2,500	2,499.34	32.20

We also present the total number of cases in which the proposed pre-filtering step was successful in increasing the initial inlier ratio and mean inlier ratio values in Table 5. The column *greater* denotes the number of cases where the inlier ratio has increased, while the column *less* provides the total number of cases where the inlier ratio has decreased after applying the pre-filtering step. Mean inlier ratios for both cases as well as overall were presented in the last three columns. It can be noted that there is a discrepancy between the total numbers presented in Table 3 and Table 5. This is due to the fact that in some cases although the pre-filtering step has decreased the inlier ratio, the robust estimation method was still able to obtain the correct motion and vice-versa.

In Figs. 17 and 18, We also present some preliminary experiments with the proposed extension of the prefiltering step in order to detect multiple motions. We generated a synthetic pair

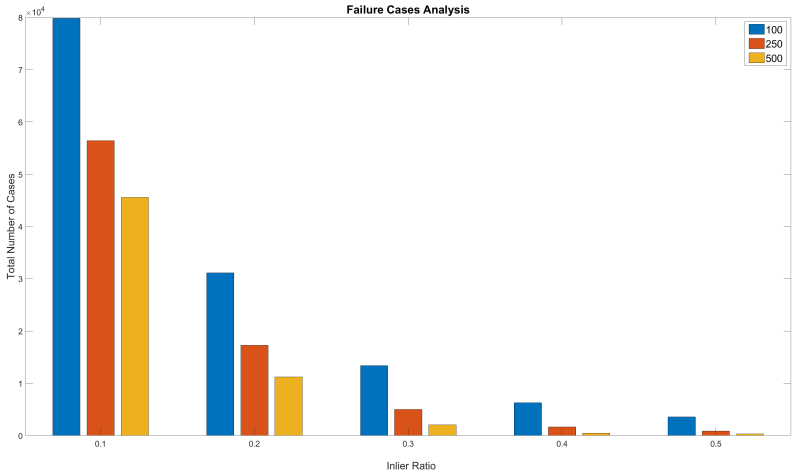


Figure 16: Histogram of the total failure cases using pre-filtering step with respect to inlier ratios. The standard image-matching procedure also failed in 63.26% of the total cases reported in this figure. In the case of a total of 100 correspondences have a larger number of failures in each inlier ratio tested leading to the conclusion that a total number of inliers plays a more important role than the ratio.

Table 5: Total Number of Cases Iterations in Robust Estimation

Inlier	Total Number of Trials			Mean Ratio		
Ratio	Greater	Less	Equal	Greater	Less	Overall
0.1	448,928	73,222	2,850	0.230	0.067	0.207
0.2	461,613	61,145	2,242	0.403	0.151	0.373
0.3	469,820	54,032	1,148	0.534	0.242	0.504
0.4	477,120	46,402	1,478	0.638	0.340	0.611
0.5	482,220	39,337	3,443	0.723	0.442	0.701
Total	2,339,701	274,138	11,161			

of images in which objects are moving differently than the camera motion. Our approach was able to identify the image regions that obey different motions.

5 Conclusions

We develop a novel robotic landing gear for off-the-shelf UAVs, which aims to enable them to land on any shape of surface. This design uses only one servo motor to reduce weight and power consumption, combined with a universal joint and multi-link design to achieve a high level of efficiency in irregular surface landing. This robotic landing gear with a mechanical structure weighs less than 1 kilogram. Using only one servo motor and two vacuum motors can allow a UAV to land on different shapes of surface. In general, if mobile robots are endowed with the capability of performing a large set of movements, it means that multiple motors are incorporated into the drive mechanism, which will cause an increase in total power consumption and weight. In order to keep the weight a minimum, we have adopted passive mechanical structures, minimizing the number of motors while keeping the expected function unaffected. This makes it possible for a UAV outfitted with the proposed landing gear to access complex and dangerous environments, such as industrial facilities, disaster sites after earthquakes, and similar others, and collect data in an energy-efficient manner.

We focused on the theoretical and geometrical analysis of the limitations of the proposed design extensively. Such studies provide useful and practical insights and aspects of the design as it can be scalable and adjustable to the UAV size. Numerical details for such purposes have

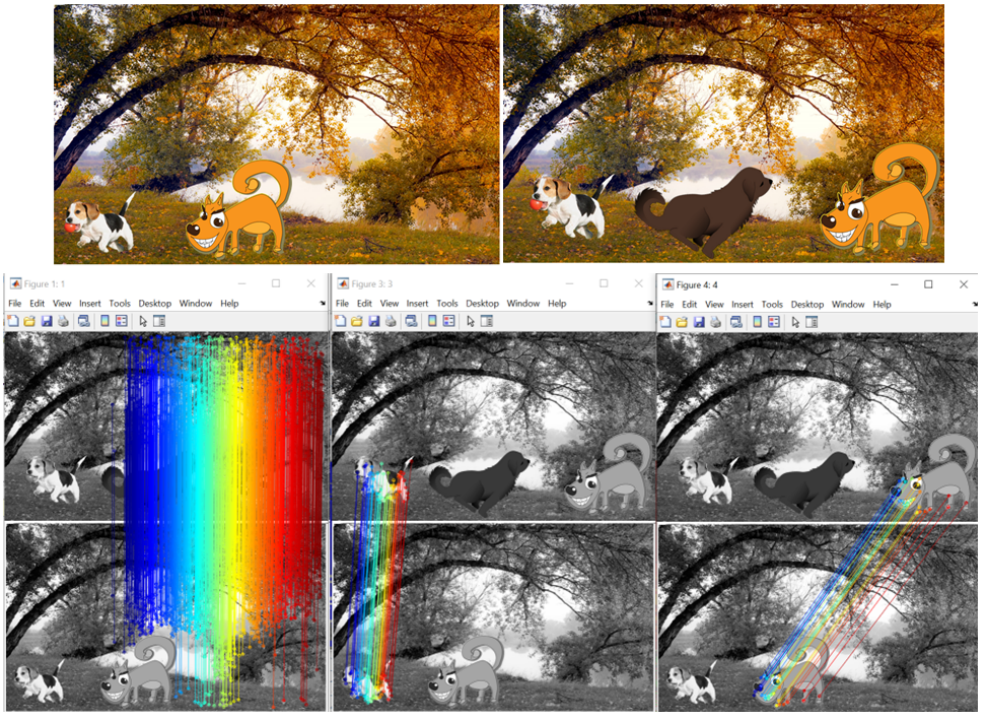


Figure 17: Synthetically generated pair of images including multiple motions. Three different motions were identified by the proposed method. While the first one is the camera motion, the second and the third one are the motions of the animals moved arbitrarily in the scene.

been studied and presented. We have also done several experiments on the landing performance. To do so, we designed an experiment bed that allows for configuring surfaces with different angles and we reported our experimental findings accordingly.

Optical sensors have been playing an important data source for robotic platforms performing different tasks. In order to select and/or recognize the landing site, we focus on improvements for the feature-based image-matching pipeline. We have developed a simple yet efficient pre-filtering step in order to reduce the outliers that occur due to the imperfections, noise, and thresholds used. This pre-filtering step is computationally light and it improves the overall pipeline both accuracy and computational cost. We also present an extension of this prefiltering step in order to detect and compute the multiple motions in the scene.

5.1 Scientific Outcomes

The developments made during the project were used and reported in the following scientific publications and we also present two videos of experiments carried out in the lab. space:

Videos

- A sample flight scenario demonstration in the Lab. `https://youtu.be/4HtYF2wJ1NQ`
- A sample video on landing different surfaces `https://youtu.be/fbsFBl-dzFs`

Peer-reviewed Scientific Publications:

- Elibol A, Chong NY. Developing a Novel Formulation for Team-Forming Problem of Coverage Problems with a Group of Heterogeneous Mobile Robots: Preliminary Results. In 2023 20th International Conference on Ubiquitous Robots (UR) 2023 Jun 25 (pp. 308-312). IEEE.

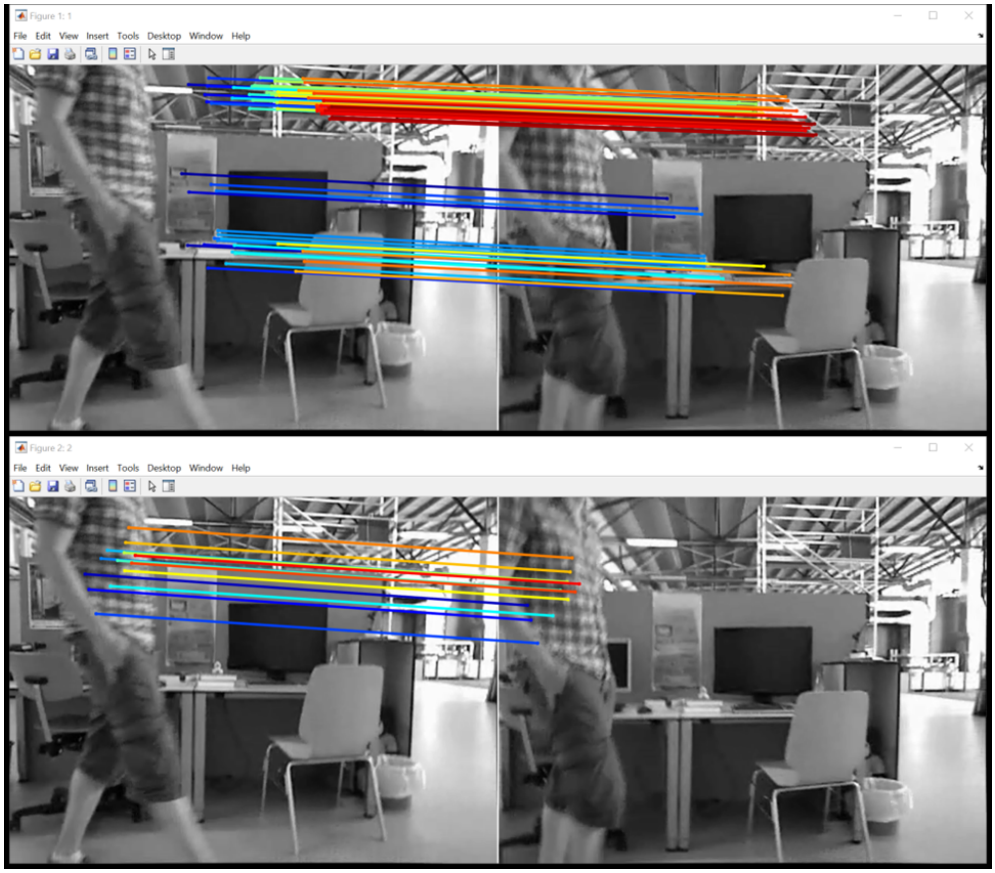


Figure 18: An example of results obtained by the proposed multiple motion detection method. Images containing a person moving in the scene are obtained from `https://cvg.cit.tum.de/data/datasets/rgbd-dataset/download`. Our approach was able to identify the camera's motion and the person's motion.

- Elibol A, Chong NY. Toward Computationally Efficient Path Generation and Push Planning for Robotic Nonprehensile Manipulation. In2023 9th International Conference on Automation, Robotics and Applications (ICARA) 2023 Feb 10 (pp. 44-48). IEEE.
- Huang, T., Elibol, A. & Chong, N.Y. Enabling landings on irregular surfaces for unmanned aerial vehicles via a novel robotic landing gear. Intel Serv Robotics 15, 231–243 (2022). `https://doi.org/10.1007/s11370-022-00420-y`
- Miyaura K, Elibol A, Chong NY. Performance Enhancement Step for Motion Estimation via Feature-based Image Matching. In2022 22nd International Conference on Control, Automation and Systems (ICCAS) 2022 Nov 27 (pp. 1161-1166). IEEE.
- Hsuan, T. C., Elibol, A., Chong, N.Y. A UAV-UUV Transformative Housing for Minimal Logistics Underwater Exploration, 2021 18th International Conference on Ubiquitous Robots (UR), July 12-14, Work-In-Progress Paper, 2021,
- Elibol A, Chong NY. Algorithmically Improved Framework for Image-only Robotic Mapping. In2021 18th International Conference on Ubiquitous Robots (UR) 2021 Jul 12 (pp. 515-520). IEEE.
- Huang, T., Elibol, A. & Chong, N.Y., A Design for UAV Irregular Surface Landing Capability, Proceedings of the 2020 17th International Conference on Ubiquitous Robots (UR), Kyoto, Japan, June 22-26, Work-In-Progress Paper, 2020.

References

[1] H. Bay, T. Tuytelaars, and L. J. Van Gool. SURF: Speeded up robust features. In *European Conference on Computer Vision*, pages 404–417, Graz, Austria, May 2006.

[2] Michael Bolt, Timothy Ferdinands, and Landon Kavlie. The most general planar transformations that map parabolas into parabolas. *Involve: A Journal of Mathematics*, 2(1):79 – 88, 2009.

[3] Ondrej Chum and Jiri Matas. Matching with prosac-progressive sample consensus. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 220–226. IEEE, 2005.

[4] Andrea Cohen and Christopher Zach. The likelihood-ratio test and efficient robust estimation. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2282–2290, 2015.

[5] M. A. Fischler and R. C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[6] Jorge Gago, Cb Douthe, REc Coopman, PPa Gallego, Mb Ribas-Carbo, Jb Flexas, Jb Escalona, and Hb Medrano. Uavs challenge to assess water stress for sustainable agriculture. *Agricultural water management*, 153:9–19, 2015.

[7] David Gómez-Candón, AI De Castro, and Francisca López-Granados. Assessing the accuracy of mosaics from unmanned aerial vehicle (uav) imagery for precision agriculture purposes in wheat. *Precision Agriculture*, 15(1):44–56, 2014.

[8] Norman Hallermann and Guido Morgenthal. Visual inspection strategies for large bridges using unmanned aerial vehicles (uav). In *Proc. of 7th IABMAS, International Conference on Bridge Maintenance, Safety and Management*, pages 661–667, 2014.

[9] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Harlow, UK, second edition, 2004.

[10] Mark Jarman, John Vesey, and Paul Febvre. Unmanned aerial vehicles (uavs) for uk agriculture: Creating an invisible precision farming technology. *White Paper, July*, 2016.

[11] AE Jimenez-Cano, G Heredia, and A Ollero. Aerial manipulator with a compliant arm for bridge inspection. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1217–1222. IEEE, 2017.

[12] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[13] Lionel Moisan, Pierre Moulon, and Pascal Monasse. Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers. *Image Processing On Line*, 2:56–73, 2012.

[14] Wancheol Myeong, Sungwook Jung, Byeongho Yu, Tirtawardhana Chris, Seungwon Song, and Hyun Myung. Development of wall-climbing unmanned aerial vehicle system for micro-inspection of bridges. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019.

[15] Photchara Ratsamee, Pakpoom Kriengkomol, Tatsuo Arai, Kazuto Kamiyama, Yasushi Mae, Kiyoshi Kiyokawa, Tomohiro Mashita, Yuki Uranishi, and Haruo Takemura. A hybrid flying and walking robot for steel bridge inspection. In *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 62–67. IEEE, 2016.

[16] Pere Ridao, Marc Carreras, David Ribas, and Rafael Garcia. Visual inspection of hydroelectric dams using an autonomous underwater vehicle. *Journal of Field Robotics*, 27(6):759–778, 2010.

[17] Clément Riu, Vincent Nozick, and Pascal Monasse. Automatic RANSAC by Likelihood Maximization. *Image Processing On Line*, 12:27–49, 2022.

[18] Peter J Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79(388):871–880, 1984.

[19] Young Kouk Song, Chang Min Lee, Ig Mo Koo, Duc Trong Tran, Hyungpil Moon, and Hyouk Ryeol Choi. Development of wall climbing robotic system for inspection purpose. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1990–1995. IEEE, 2008.

[20] P.H.S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000.