

Journal of Electronic Imaging

JElectronicImaging.org

Learning-based risk assessment and motion estimation by vision for unmanned aerial vehicle landing in an unvisited area

Hsiu-Wen Cheng
Tsung-Lin Chen
Chung-Hao Tien

Learning-based risk assessment and motion estimation by vision for unmanned aerial vehicle landing in an unvisited area

Hsiu-Wen Cheng,^a Tsung-Lin Chen,^a and Chung-Hao Tien^{b,*}

^aNational Chiao Tung University, Department of Mechanical Engineering, Hsinchu, Taiwan

^bNational Chiao Tung University, Department of Photonics, Hsinchu, Taiwan

Abstract. We proposed a vision-based methodology as an aid for an unmanned aerial vehicle (UAV) landing on a previously unsurveyed area. When the UAV was commanded to perform a landing mission in an unknown airfield, the learning procedure was activated to extract the surface features for learning the obstacle appearance. After the learning process, while hovering the UAV above the potential landing spot, the vision system would be able to predict the roughness value for confidence in a safe landing. Finally, using hybrid optical flow technology for motion estimation, we successfully carried out the UAV landing without a predefined target. Our work combines a well-equipped flight control system with the proposed vision system to yield more practical versatility for UAV applications. © The Authors. Published by SPIE under a Creative Commons Attribution 4.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JEI.28.6.063011](https://doi.org/10.1117/1.JEI.28.6.063011)]

Keywords: optical flow; self-supervised learning; unmanned aerial vehicle; vision-based landing.

Paper 190509 received May 28, 2019; accepted for publication Nov. 27, 2019; published online Dec. 20, 2019.

1 Introduction

Unmanned aerial vehicles (UAVs) are widely used in many fields, from military to civilian to commercial. For the sake of efficiency and convenience, a higher degree of autonomy is required to minimize human intervention. Many efforts have been made to develop vision-based technologies for UAV maneuvers. One challenge of this technique lies in how to land a UAV on a previously unvisited area. Generally, the issue can be divided into two parts: identifying a flat area for safe landing (landing risk assessment) and landing accurately on an unknown spot (motion estimation).

For the case of landing site selection, the goal is to find a planar surface with a small slope that is free of obstacles, which can be conducted through assessing the landing risk by either constructing an elevation map or evaluating the planarity of terrain appearance. The former approach shall build a full three-dimensional (3-D) geometry¹ with the corresponding coordinates of the environment by means of a sequence of images, e.g., structure from motion (SFM).²⁻⁵ As a consequence of estimated topographical information, a two-dimensional elevation map was thus obtained to comprehend the region flatness by, for instance, least median squares⁶ or plane fit.⁷ Incorporated with the coordinates about the surrounding, the information is useful for the likely landing site. SFM and related methods enable an image-based 3-D scene as well as determine the safety landing site, but with the cost of heavy computation. In order to directly obtain the absence of obstacles for the landing merit, alternatively, another incomplete but effective scheme was used. The planar area can be extracted through the homography estimation.^{8,9} Without the need for a region extraction process, the roughness estimation from the optical flow field was proposed to measure the planarity of the surface.¹⁰ In

addition, researchers addressed the learning process to yield more practical versatility for UAV applications, such as supervised learning for texture classification,¹¹ neural network policy for navigation,¹² and deep reinforcement learning for marker recognition.¹³ For the aspect of landing risk, a self-supervised learning (SSL) method was employed to overcome the constraint that significant movement is required for optical-flow-based roughness estimation.¹⁴

After determining an adequate landing site in an unvisited area, the follow-up is to complete the landing process guided by either the positioning system [e.g., global positioning system (GPS)] or the vision-based motion estimation system. In terms of the vision-based landing scheme, several patterns are designed as markers to tackle the close range and nighttime detection problem during UAV descent.¹⁵⁻¹⁸ Moreover, while landing on a moving target, schemes either optimizing the marker detection rate¹⁹ or exploiting the moving target's dynamic model were developed accordingly.²⁰ However, the performance of the aforementioned schemes mainly relies on the specific target pattern and is unlikely to be applied in an unvisited environment where there is no chance to set a well-defined landing guide in advance.

Our team aims to develop a fully vision-based system for UAV landings in a previously unvisited environment. Resuming our previous work on vision-based landing motion estimation,²¹ we further integrated the vision system with the learning algorithm. The major function of the proposed system is to classify the obstacle appearance on the ground and provide an accurate measure of motion during the landing. To achieve these aims, we introduced the SSL to model the relationship between visual appearance and surface roughness and developed a classifier to determine if the land is safe for landing by recognizing the predicted roughness (yes/no question). Moreover, the hybrid optical flow scheme was also employed to ensure the motion estimation throughout the entire landing process without prior knowledge of guiding markers. The remainder is organized

*Address all correspondence to Chung-Hao Tien, E-mail: chtien@nctu.edu.tw

as follows. In Sec. 2, we explain the concept of roughness estimation as well as the methodology for landing site identification. In Sec. 3, we introduce the hybrid framework for visual motion detection, including the multiscale strategy for positioning to tackle the field-of-view problem during descent. Afterward, experimental verification is given in Sec. 4. Finally, conclusion and future work are drawn in Sec. 5.

2 Learning of Obstacle Appearance

SSL is a classic approach that uses input signals as the sources for supervision. Instead of human intervention, the training labels were determined by the collected data. Therefore, to learn the obstacle appearance in view, we must gather the visual cues as the input objects and the surface planarity as the corresponding supervised output values.

Figure 1 shows the process of the learning algorithm for the obstacle appearance. Phase I: since the area is previously unvisited, we first navigated the UAV to capture the images as the clustering dataset. The texton dictionary^{22,23} was then built as an attribute of the surface texture features. Phase II: we collected the training data, including the surface roughness measured from the optical flow field, and the texton distribution formed by matching the randomly selected patches and the labeled textons. We used the regression to model the relationship amid the surface roughness and the texton distribution. Phase III: after completing the learning step, the UAV would have the capability to identify obstacles in a still image through the predicted roughness, thereby ensuring a safe landing on the unvisited area. In order to save computational effort, the imaging algorithms were merely effective in the region-of-interest (ROI) of the input image stream. Details will be explained in the following content.

2.1 Patch Operation for Visual Appearance

In this study, we used the texton method²⁴ to attribute the visual appearance of an unvisited airfield. By clustering the characteristic values from multiple image patches, we can generate a texton dictionary that represents the surface texture features. In our implementation, each 3×3 image patch was rearranged into a 1×9 vector \mathbf{x} with its grayscale values. Then these vectors were partitioned into k sets by a K -means clustering algorithm,²⁵ where each vector belongs to the cluster with the nearest mean. The objective function can be defined as

$$\arg \min_S \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2, \tag{1}$$

where $\boldsymbol{\mu}_i$ is the mean of observations in cluster S_i . With this method, the cluster centroids formed a visual dictionary for the unvisited area.

After creating the texton dictionary, the rendering features from images can be characterized by a texton probability distribution, as shown in Fig. 2. For each randomly extracted patch, we searched for the closest match in the dictionary based on the Euclidean distance and added it to the corresponding bin in a histogram. Finally, we obtain the texton distribution \mathbf{q} by normalizing with the number of extracted patches m as follows:

$$\mathbf{q} = \frac{1}{m} (q_1 \ q_2 \ \dots \ q_k), \tag{2}$$

where

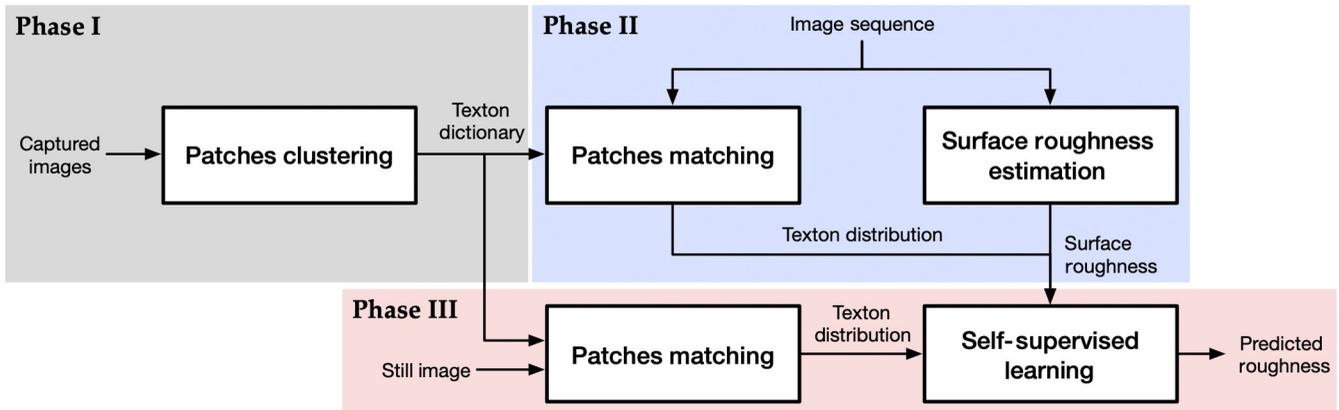


Fig. 1 The overview of learning the obstacle appearance. Phase I: creating a texton dictionary for the unvisited area based on the captured images during navigation. Phase II: collecting the training data, texton distribution, and surface roughness through the image sequence in the second navigation. Phase III: hovering above the test area (red star) and predicting the surface roughness using the SSL method.

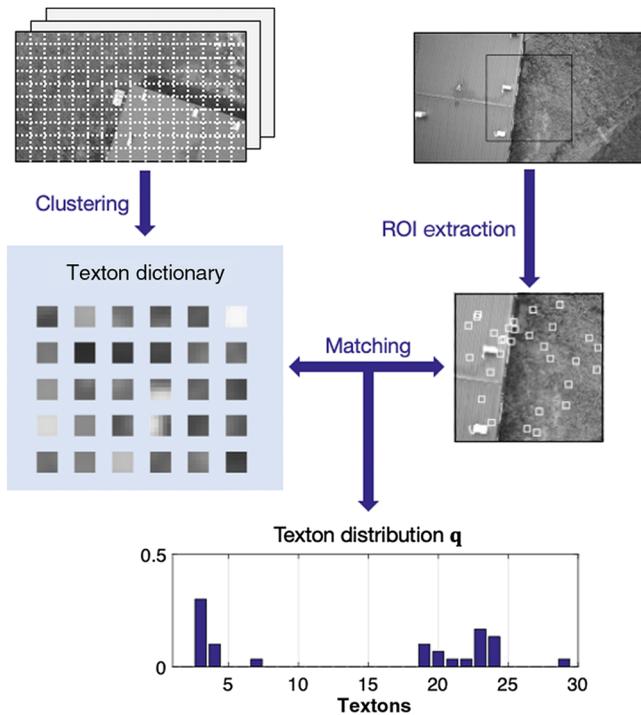


Fig. 2 After the images were split into small patches, the texton dictionary was created by K -means clustering algorithm ($k = 30$). Then the texton distribution was obtained through a series of patch matching.

$$q_i = \sum b_i, \quad i = 1, \dots, k, \quad (3)$$

$$b_i = \begin{cases} 1, & i = \arg \min_j \| \mathbf{x} - \boldsymbol{\mu}_j \| \\ 0, & \text{otherwise} \end{cases}. \quad (4)$$

2.2 Surface Roughness using Optical Flow Algorithm

In order to determine if the selected spot is suitable for landing, we estimated the surface roughness as the merit of the

safe landing. The concept of roughness estimation is to regard the optical flow components as a set of points for a plane fitting problem, where the fitting error was adopted as the measure of roughness. Based on previous research in Ref. 10, the camera model based on the optical flow algorithm shall satisfy the following conditions: (1) downward-looking camera, (2) planar surface in sight, and (3) known angular rates of the camera. Under these assumptions, the optical flow vectors can be generalized as follows:

$$\begin{cases} u = -u_0 + (au_0 + w_0)x + \beta u_0y - \beta w_0xy - aw_0x^2 \\ v = -v_0 + \alpha v_0x + (\beta v_0 + w_0)y - aw_0xy - \beta w_0y^2 \end{cases}, \quad (5)$$

where u and v are the optical flow vectors in the x and y directions of the image coordinates system, respectively. u_0 , v_0 , and w_0 are the corresponding velocities in the x , y , and z directions scaled with respect to the altitude. α and β are the tangents of the slope angles of the surface.

According to Eq. (5), the magnitude of the optical flow would be inversely proportional to the flight height above the surface. Therefore, we can estimate the surface roughness by fitting the optical flow field. Since the UAV moved nearly laterally, Eq. (5) can be simplified to Eq. (6). The parameter vectors \mathbf{p}_u and \mathbf{p}_v can be calculated separately by solving a linear fitting problem within a random sample consensus (RANSAC) procedure.

$$\begin{cases} u = \mathbf{p}_u(1 \ x \ y)^T \\ v = \mathbf{p}_v(1 \ x \ y)^T \end{cases}. \quad (6)$$

RANSAC iterations gave us the estimated surface plane and the corresponding fitting error, serving as the measure of surface roughness (Fig. 3). If there exists any obstacle on the surface, the procedure would lead to a higher fitting error in u , v , or both directions. Consequently, we combined the results in both directions as the overall surface roughness.

$$\varepsilon_{\text{opt}} = \varepsilon_u + \varepsilon_v. \quad (7)$$

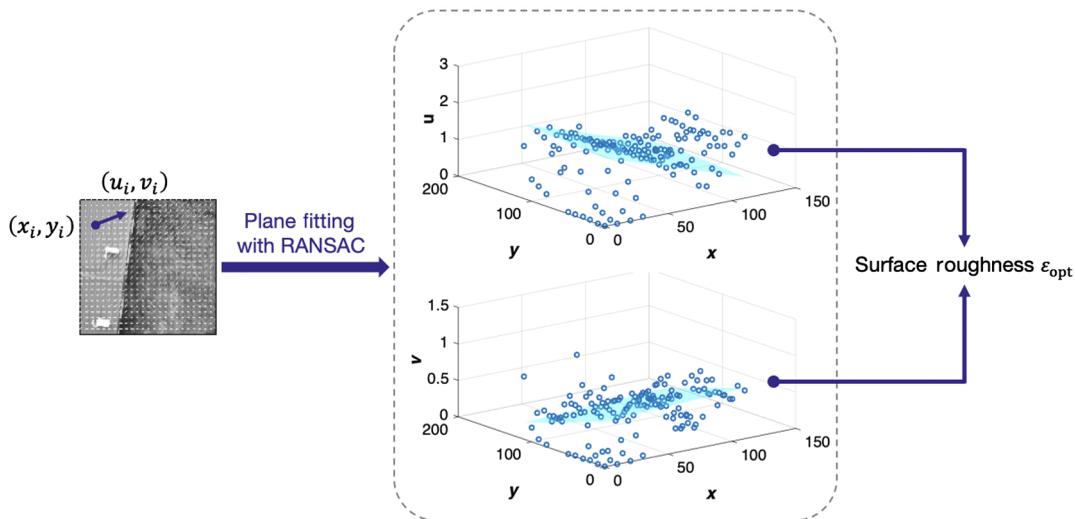


Fig. 3 An example of roughness estimation: in each RANSAC iteration, the parameter vectors were calculated by fitting a plane to a randomly selected subset of data points. The procedure returned the optimal model parameters as well as the fitting error, which can be interpreted as the measure of surface roughness.

2.3 Self-Supervised Learning

In the previous section, we introduced the concept of roughness estimation using the optical flow technique. However, since the fitting dataset consists of velocity vectors, the roughness estimation requires significant movement to guarantee a moderate result, which is not viable in the hovering mode. In order to ensure the accuracy of obstacle prediction, we proposed an SSL scheme to map visual appearance features \mathbf{q} to roughness values ε_{opt} .

In this study, K -nearest neighbor regression²⁶ was used as the learning method due to its simplicity and flexibility. The algorithm is a nonparametric method that keeps all available data and predicts the numerical response based on the proximity measure. A dataset of training samples is given as follows:

$$\{\boldsymbol{\varepsilon}, \mathbf{Q}\} = \begin{pmatrix} \varepsilon_{\text{opt}}^1, \mathbf{q}^1 \\ \vdots \\ \varepsilon_{\text{opt}}^n, \mathbf{q}^n \end{pmatrix}. \quad (8)$$

The algorithm performed predictions by calculating the similarity between the input sample \mathbf{q}' and each instance of the training data. Finally, the predicted roughness came out with the mean of K neighbors' responses.

$$\varepsilon_{\text{SSL}} = \frac{1}{K} \sum_{\mathbf{q} \in N_K} \varepsilon_{\text{opt}}, \quad (9)$$

where N_K denotes the K objects in the database that are closest to the input \mathbf{q}' . After completing the learning process, we can estimate the surface roughness corresponding to any input distribution through the regression model.

3 Vision Motion Estimation for Landing

Once the landing site was selected, the UAV was commanded to descend steadily until touchdown. For vision-based landings, in the case of an unvisited environment, there exist some possibilities that the marker-based vision system might fail to recognize the marker feature. Therefore, we introduced a hybrid framework to select the processing image frames and algorithms based on the required estimations, thus meeting the need to land in an unvisited environment with no target recognition required. In addition, an augmented phase correlation (PC) method with a multiscale strategy was introduced to tackle the problem of scale variation resulting from the field-of-view change during descent.

3.1 Hybrid Optical Flow Technology

The hybrid optical flow technique, correlating multiple image frames, was proposed to measure two dynamic motions for landing controls: velocity and position. As shown in Fig. 4, the velocity estimation is determined by comparing two consecutive frames ($I_N - I_{N-1}$). On the other hand, the position information is computed by the deviation between the reference frame and the N 'th frame ($I_N - I_0$) to avoid integral error accumulations.

The image processing in the part of motion estimation was also shown in Fig. 4. The vision system generated the guidance information with the flight data from the control system. Following our previous work,²¹ we dynamically

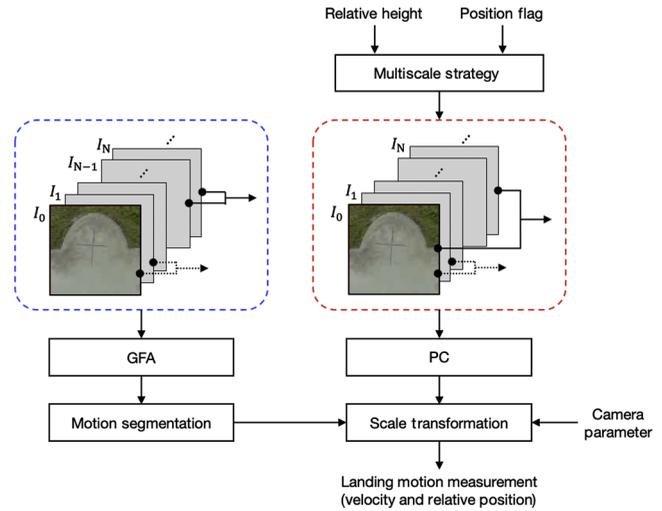


Fig. 4 The conceptual framework of visual motion estimation for landing: the velocity is obtained by comparing image patterns on two consecutive frames based on the Grunnar–Farneback algorithm, and the position is computed by the proposed multiscale PC method with respect to a reference image.

adopted the Gunnar–Farneback algorithm²⁷ and PC method^{28–30} to obtain the velocity and position, respectively. These two algorithms worked together to combine the advantages of dense and sparse optical flow in terms of accuracy and robustness. Both the velocity and position measurements were employed as the feedback signals to the flight control system.

3.2 Multiscale Strategy

The typical PC approach could only tolerate a small range of scale difference in between two image frames. During the landing process, however, the vision-based motion estimation would experience a large-scale difference due to the decrease in height. Accordingly, in the previous work, we introduced a multiscale strategy to autonomously adjust the sensed ROI and update the reference ROI for relative position estimation.

Figure 5 shows the concept of the multiscale PC method. First, the vision system set the reference ROI (I_0) at the instant that the UAV was commanded to descend. As the UAV was descending, the size of the ROI extraction was enlarged by a scale factor λ . Then the sensed ROI (I_N) was carried out by resizing with the same parameter to maintain the same scale as the reference ROI. Finally, the sensed ROI and the reference ROI were applied to the PC function. The factor λ can be computed by the height of the UAV.

$$\lambda = \frac{H_{\text{ref}}}{H}, \quad (10)$$

where H_{ref} and H are the flight heights of the reference image and the sensed image, respectively. In our experiments, the flight height was obtained by a laser altimeter. In addition, while the UAV was descending, an instantly updating reference ROI image was necessary to ensure that it contained a sufficient overlap region for ROI extraction in subsequently captured images. The reference ROI was autonomously updated when λ reached a preset threshold value. It is noted that the position estimate was carried out

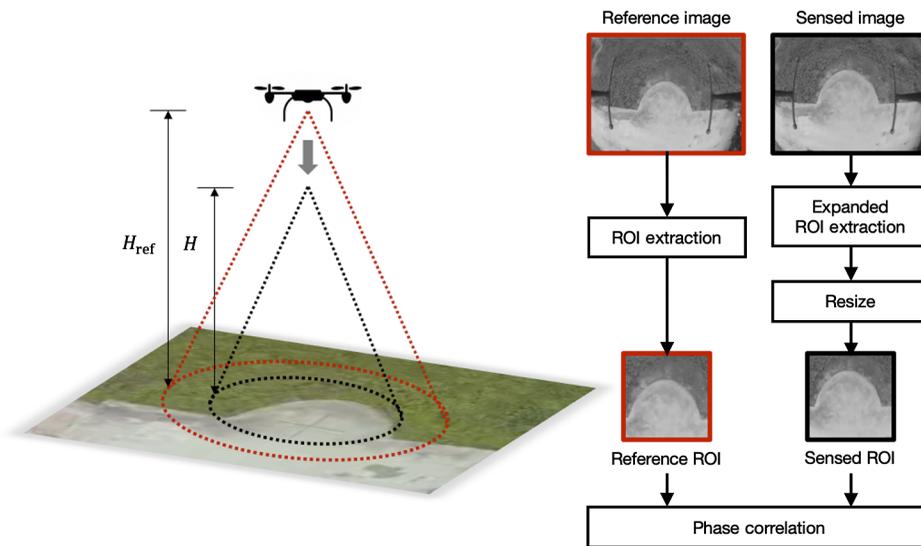


Fig. 5 Illustration of the multiscale PC method (adapted from Ref. 21, Fig. 5, with permission). The size of ROI extraction was expanded according to the flight altitude and the sensed ROI was applied to the PC function after resizing back to the default size. Furthermore, a threshold value was set to trigger the update of the reference ROI.

with the prior measurement as an initial condition when passing through the next epoch of the reference ROI. Augmenting the PC method with the proposed multiscale strategy, we effectively minimize the sensing error, which was proved by our experimental results.

4 Experimental Results

In this section, we reveal the experiments of vision-aided landing in a previously unsurveyed area. First, we compared the SSL method with the optical flow method in the roughness estimation. Then we conducted the UAV landing with a hybrid optical flow scheme. It is noted that the learning process was carried out before these experiments.

The testing UAV was a commercial quadrotor Stellar 1000× from InnoFlight™ as shown in Fig. 6. It was pre-equipped with a flight control system (InnoFlight™ Jupiter JM-1 Autopilot), an inertial measurement unit, a laser altimeter, and a GPS module, respectively. In addition, for visual equipment, the image processing system included a NVIDIA™ Jetson TK1 module and a GoPro™ HERO 3+

camera. The embedded program in the vision computer executed the image processing algorithms and carried out communications with the flight control computer, which ran the proportional–integral–derivative control scheme.

4.1 Obstacle Detection using Self-Supervised Learning

In this experiment, we hovered the UAV above several spots (marked by the red star in Fig. 1), 10 s per spot, and examined the capability of the vision system for obstacle detection. Area 1 and area 2 contained obstacles for buildings and cars, respectively, whereas the other two spots (area 3 and area 4) had no obstacles. We computed the mean and the standard deviation of estimated roughness from the optical flow algorithm (ϵ_{opt}) and the SSL method (ϵ_{SSL}) and inferred the classification rule for a safe landing site based on these results.

Figure 7 shows the estimated results of surface roughness while the UAV was hovering. As we predicted in Sec. 2.3, the accuracy of surface roughness estimation via optical flow

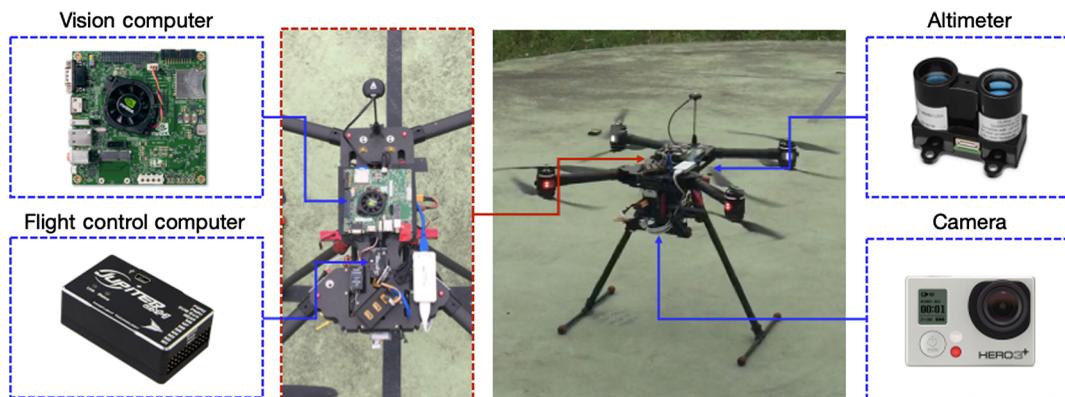


Fig. 6 The testbed used in the experiments (reprinted from Ref. 21, Fig. 7, with permission). In addition to the preinstalled flight control system, the quadrotor is equipped with a vision computer and a HERO 3 external camera.

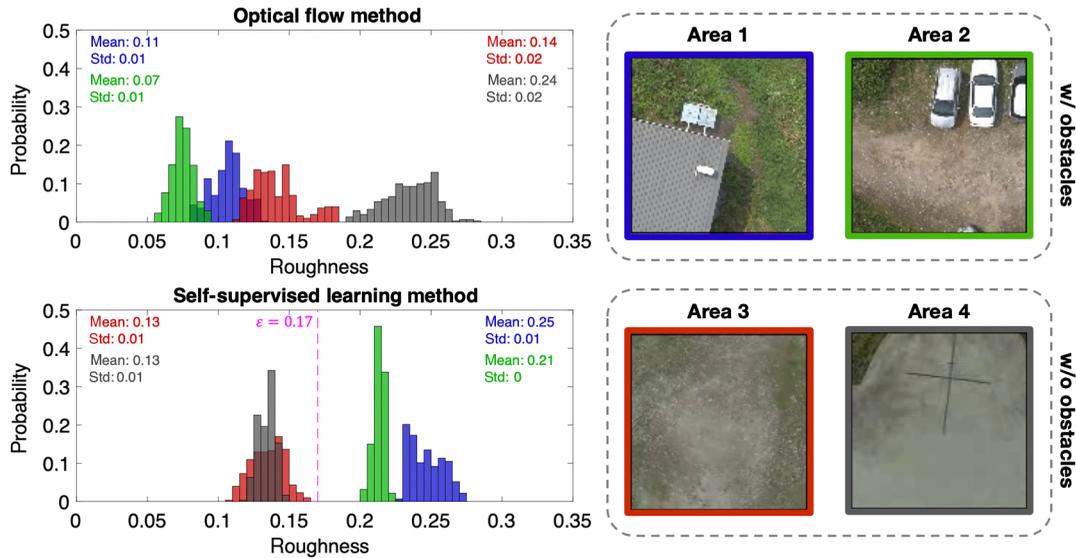


Fig. 7 Results of surface roughness while hovering. The SSL method managed to detect obstacles by giving a higher value of predicted roughness, whereas the optical flow method failed to correlate the estimates with surface flatness.

highly depends on the UAV movement extent. Due to the absence of lateral movement, the optical flow vectors would fail to reveal the magnitude difference that is subject to surface fluctuation. As can be seen from the results of the optical flow method (top-left figure), the roughness estimation failed to cluster, and thus had poor determination in terms of classification. In contrast, the roughness predicted by the SSL method provided a better evaluation (bottom-left figure). The roughness distribution can be entirely classified into two groups, corresponding to regions with and without obstacles. In addition to the clear threshold ($\epsilon_{SSL} < 0.17$) that the proposed vision system can identify a safe landing area (with low roughness value), the distribution exhibits that the larger the area occupied by the obstacles, the higher the roughness value.

4.2 Landing Controls with Hybrid Motion Estimation

In preparation for landing (hovering phase), the UAV was commanded to hold its position above the likely landing spot. After confirming no obstacles in view, the UAV then started to steadily descend with visual feedback using the hybrid optical flow technique. Although the visual motion estimation was in effect, the system also collected the GPS data simultaneously as the benchmark. In this work, the position accuracy was verified using the template matching method^{31,32} under multiple flight trials.

A video demo can be found in Ref. 33, whereas Figs. 8(a) and 8(b) show the flight data and the in-plane route during the landing process. In terms of velocity estimation, the precision of the vision-based method was comparable with the state-of-the-art GPS, with only a 0.1 m/s difference in both x and y directions. The vision-based landing was activated at P_0 (black cross), i.e., the target spot set at the coordinate origin. At the end of the landing, the UAV was located at P_{vision} (blue dot) and P_{gps} (green dot) according to the sensing value of the vision system and GPS, respectively. The corresponding images of camera view were also presented in Fig. 8(c). For the position part, we used the template matching method

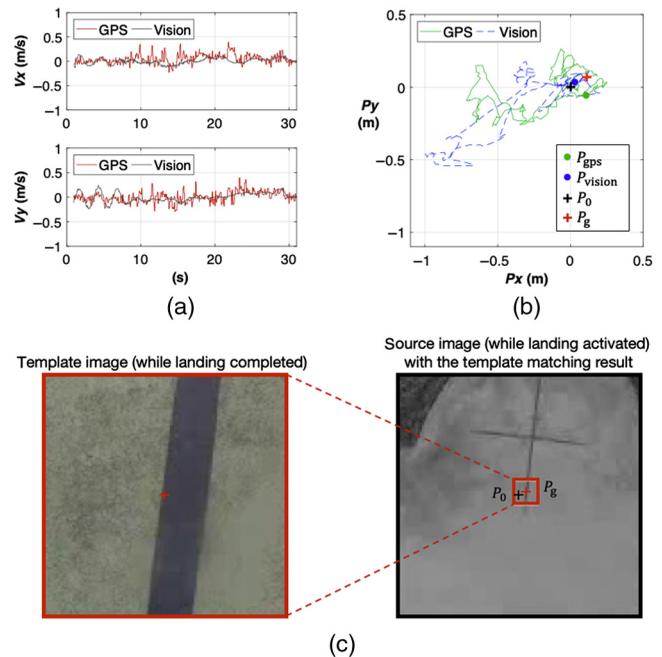


Fig. 8 Vision-based landing (adapted from Ref. 21, Figs. 10 and 11, with permission). (a) The resulting velocity estimates of the vision system (black line) followed a similar trend with GPS (red line), with a standard deviation of error about 0.1 m/s in both directions. (b) The in-plane route is obtained from the GPS (green line) and the vision system (blue line). The landing process started from the origin (black cross) and ended at the point marked with a dot. (c) The result of template matching shows that the UAV landed ~ 0.1 m away from the designated spot.

to authenticate the landing accuracy. The detected location P_g (red cross) by template matching was considered as the ground truth of positioning accuracy, and its coordinate was also indicated in Fig. 8(b). In terms of overview, the vision-based landing resulted in ~ 0.1 m of in-plane positioning error. These results suggested that the hybrid vision-based

scheme is able to guide the UAV landing precisely without prior information of a particular marker.

5 Conclusions

In this paper, we proposed a vision-aided system to aid the UAV landing in an unsurveyed environment. The overall procedure involved identifying the safety of the landing spot and completing the landing at that location. To assess the landing risk, the system used an SSL algorithm to construct the regression model for roughness estimation. The texton distribution formed by patch matching was used to represent the visual features in view, and the concept of roughness was used to determine whether the ground underneath was a safe landing site or not. For a newly acquired distribution, the proposed system obtained the roughness information through the regression model and further classified the presence of obstacles. Compared with the pure optical flow method, the SSL method allowed the UAV to estimate the roughness in the hovering phase, which is more practical in UAV landing operations.

Then we applied the visual motion estimation framework for landing as proposed in our previous work, including a multiscale strategy that could tackle the problem of scale variation during descending. With this method, we can land a UAV with no well-defined target required on the ground. In addition, the experimental results indicated that we successfully carried out the vision-based autonomous landing with a positioning error of ~ 0.1 m. The detailed discussion of landing performance can be found in our previous work.²¹

To enable the vision system more completely in UAV applications, more effort shall be paid to tackle the problem of visually retrieving depth information. Moreover, online learning would make the UAV more versatile by autonomously selecting a safe landing spot. In this way, a fully vision-based system can be employed to implement UAV autoland in an unsurveyed environment.

Acknowledgments

This work was financially supported by the Ministry of Science and Technology (MOST) of Taiwan government under Grant Nos. 107-2221-E-009-084, 107-2221-E-009-115-MY3, and 108-2221-E-009-108.

References

1. Y. Diskin and V. Asari, "Dense point-cloud representation of a scene using monocular vision," *J. Electron. Imaging* **24**(2), 023003 (2015).
2. S. Ullman, "The interpretation of structure from motion," *Proc. R. Soc. London Ser. B* **203**(1153), 405–426 (1979).
3. T. Templeton et al., "Autonomous vision-based landing and terrain mapping using an MPC-controlled unmanned rotorcraft," in *Proc. IEEE Int. Conf. Rob. and Autom.*, pp. 1349–1356 (2007).
4. C. Forster et al., "Continuous on-board monocular-vision-based elevation mapping applied to autonomous landing of micro aerial vehicles," in *Proc. IEEE Int. Conf. Rob. and Autom.*, pp. 111–118 (2015).
5. T. Hinzmann et al., "Free LSD: prior-free visual landing site detection for autonomous planes," *IEEE Rob. Autom. Lett.* **3**, 2545–2552 (2018).
6. A. Johnson, J. Montgomery, and L. Matthies, "Vision guided landing of an autonomous helicopter in hazardous terrain," in *Proc. IEEE Int. Conf. Rob. and Autom.*, pp. 3966–3971 (2005).
7. J. Mackay, G. Ellingson, and T. W. McLain, "Landing zone determination for autonomous rotorcraft in surveillance applications," in *AIAA Guidance, Navigation, and Control Conf.*, American Institute of Aeronautics and Astronautics (2016).
8. S. Bosch, S. Lacroix, and F. Caballero, "Autonomous detection of safe landing areas for a UAV from monocular images," in *IEEE/RSJ Int. Conf. Intell. Rob. and Syst.*, pp. 5522–5527 (2006).
9. M. Rebert et al., "Parallax beam: a vision-based motion estimation method robust to nearly planar scenes," *J. Electron. Imaging* **28**(2), 023030 (2019).
10. G. de Croon et al., "Optic-flow based slope estimation for autonomous landing," *Int. J. Micro Air Veh.* **5**(4), 287–297 (2013).
11. A. Cesetti et al., "Autonomous safe landing of a vision guided helicopter," in *Proc. IEEE/ASME Int. Conf. Mechatron. and Embedded Syst. and Appl.*, pp. 125–130 (2010).
12. A. Loquercio et al., "DroNet: learning to fly by driving," *IEEE Rob. Autom. Lett.* **3**(2), 1088–1095 (2018).
13. R. Polvara et al., "Toward end-to-end control for UAV autonomous landing via deep reinforcement learning," in *Int. Conf. Unmanned Aircraft Syst.*, pp. 115–123 (2018).
14. H. W. Ho et al., "Optical-flow based self-supervised learning of obstacle appearance applied to MAV landing," *Rob. Autom. Syst.* **100**, 78–94 (2018).
15. S. Lange, N. Sunderhauf, and P. Protzel, "A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments," in *Int. Conf. Adv. Rob.*, Munich, Germany, pp. 1–6 (2009).
16. H. Lee, S. Jung, and D. H. Shim, "Vision-based UAV landing on the moving vehicle," in *Int. Conf. Unmanned Aircraft Syst.*, pp. 1–7 (2016).
17. X. Chen, S. K. Phang, and B. Chen, "System integration of a vision-guided UAV for autonomous tracking on moving platform in low illumination condition," in *ION Pacific PNT Conf.* (2017).
18. S. Lin, M. A. Garratt, and A. J. Lambert, "Monocular vision-based real-time target recognition and tracking for autonomously landing an UAV in a cluttered shipboard environment," *Auton. Rob.* **41**(4), 881–901 (2017).
19. S. Kyriatsis et al., "Towards autonomous modular UAV missions: the detection, geo-location and landing paradigm," *Sensors* **16**(11), 1844 (2016).
20. D. Falanga et al., "Vision-based autonomous quadrotor landing on a moving platform," in *IEEE Int. Symp. Safety, Secur. and Rescue Rob.*, Shanghai, China, pp. 200–207 (2017).
21. H.-W. Cheng, T.-L. Chen, and C.-H. Tien, "Motion estimation by hybrid optical flow technology for UAV landing in an unvisited area," *Sensors* **19**(6), 1380 (2019).
22. B. Julesz, "Textons, the elements of texture perception, and their interactions," *Nature* **290**(5802), 91–97 (1981).
23. T. Leung and J. Malik, "Recognizing surfaces using three-dimensional textons," in *Proc. Seventh IEEE Int. Conf. Comput. Vision*, Vol. 1012, pp. 1010–1017 (1999).
24. M. Varma and A. Zisserman, "Texture classification: are filter banks necessary?" in *IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognit., Proc.*, p. II-691 (2003).
25. D. Arthur and S. Vassilvitskii, "K-means++: the advantages of careful seeding," in *Proc. Eighteenth Annu. ACM-SIAM Symp. Discrete Algorithms* (2007).
26. T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory* **13**(1), 21–27 (1967).
27. G. Farnèbäck, "Two-frame motion estimation based on polynomial expansion," *Lect. Notes Comput. Sci.* **2749**, 363–370 (2003).
28. C. D. Kuglin and D. C. Hines, "The phase correlation image alignment method," in *IEEE Int. Conf. Syst., Man, and Cybern.*, New York, pp. 163–165 (1975).
29. V. Argyriou and T. Vlachos, "Extrapolation-free arbitrary-shape motion estimation using phase correlation," *J. Electron. Imaging* **15**(1), 010501 (2006).
30. N. Ma et al., "A subpixel matching method for stereovision of narrow baseline remotely sensed imagery," *Math. Probl. Eng.* **2017**, 1–14 (2017).
31. J. P. Lewis, "Fast template matching," *Vision Interface* **95** (1994).
32. A. Sibiryakov, "Fast and high-performance template matching method," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 1417–1424 (2011).
33. H.-W. Cheng, T.-L. Chen, and C.-H. Tien, "exp.mp4," <http://tinyurl.com/y656uepr> (5 March 2019).

Hsiu-Wen Cheng received her BS and MS degrees in aeronautics and astronautics engineering from National Cheng Kung University, Tainan, Taiwan, in 2007 and 2009, respectively. She is a PhD candidate in mechanical engineering at National Chiao Tung University (NCTU), Hsinchu, Taiwan. Her research interests are in image processing with a focus on unmanned aerial vehicle applications.

Tsung-Lin Chen received his BS and MS degrees in power mechanical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 1990 and 1992, respectively, and his PhD in mechanical engineering from the University of California, Berkeley, in 2001. From 2001 to 2002, he joined Analog Devices Inc. as a MEMS design engineer. Since 2003, he has been in the Department of Mechanical Engineering, NCTU, Hsinchu, Taiwan. Currently, he is a full professor.

His research interests include microelectromechanical systems and controls.

Chung-Hao Tien received his BS degree in electrical and computer engineering and his PhD in electro-optical engineering from NCTU, Hsinchu, Taiwan, in 1997 and 2003, respectively. He was a research assistant at the University of Arizona, Tucson, USA, in

2001. After being a postdoctoral fellow at Carnegie Mellon University, Pittsburgh, USA, from 2003 to 2004, he joined NCTU as a faculty and now is a full professor in the Department of Photonics. His current research interests include computational imaging and information optics. He is currently a member of the Phi Tau Phi Honor Society, the Optical Society of America, and SPIE.