Proceedings of The 2017 IEEE International
Conference on Real-time Computing and Robotics
July 14-18, 2017, Okinawa, Japan

A Monocular Vision Localization Algorithm Based on Maximum Likelihood Estimation*

Shiyu Chen, Yanjie Li, and Haoyao Chen

Abstract— **In this paper, we present a method that uses the theory of maximum likelihood estimation to improve the precision of the unmanned aerial vehicle (UAV) localization algorithm based on monocular vision. The main goal of this work is to obtain the accurate position information of UAV and achieve the autonomous navigation in complex indoor and outdoor environments. An embedded camera mounted on the UAV platform is used to provide real-time video streams to the vision-based localization algorithm. All the algorithms run in the onboard computer to ensure the real-time property of the system. Simulation of the UAV platform with monocular camera is performed to verify the feasibility of the improved localization algorithm firstly. After the simulation verification, a series of real-time experiments are implemented to demonstrate the precision of the algorithm.**

I. INTRODUCTION

With the development of science and technology, the UAV systems have been widely used in various fields, such as traffic monitoring [1], military surveillance [2] and disaster relief [3] in recent years. In order to ensure that the tasks can be completed satisfactorily, the position information of UAV plays an important role during the process of flight control.

In general, there are four kinds of localization methods that can get the position of the UAV, including global positioning system (GPS), motion capture system, laser-based and vision-based positioning technology. GPS is usually used to locate the UAV in outdoor environment, but it will out of action when UAV flies indoor or at the place with many buildings. The motion capture system can get the position of UAV with very high accuracy, however, the application range is limited. Laser-based and vision-based positioning methods can be adapted in more general environments. Because of the high cost of laser sensor, the vision-based localization algorithms have been rapidly developed and applied into low-cost UAV systems in recent years.

In 2009, Soloviev [4] combined the optical flow method with inertial measurement unit (IMU) to determine the position of UAV and realized the autonomous flight control of UAV. Klein [5] adapted simultaneous localization and mapping (SLAM) method for the localization of UAV and presented a localization algorithm named parallel tracking and mapping (PTAM). The algorithm can perform well just based on a monocular camera in a planar scene, but the accuracy will be declined in general scene. Then, Forster [6] developed the algorithm and proposed a kind of monocular visual odometry method named semi-direct monocular visual odometry (SVO). Because the method used pixel gray value as the feature points of images, the feature was easy to be lost when the camera moved at a high speed and caused the failure of localization. In order to get a more robust localization algorithm of UAV, Mur-Artal used oriented FAST and rotated BRIEF (ORB) feature [7] to characterize the images and presented a localization method named ORB-SLAM [8]. The method just used a monocular camera to build the map of current scene and obtain the position of the camera real-timely. But this algorithm has a general defect that the monocular camera can't get the depth information of the scene. Thus, there is a certain scale error between the position obtained from the method and real environment. Weiss [9], [10] used the theory of Extended Kalman Filter (EKF) to fuse the information created by the localization method with IMU data and got the real position information with little error.

In this paper, we use ORB-SLAM method to obtain the position of the UAV platform which has a scale problem. And we present a method that uses the theory of maximum likelihood estimation to solve the scale problem. Finally, simulations and experiments are implemented to verify the effectiveness of the method.

The paper is organized as follows. Section II describes the system overview. The ORB feature extraction is presented in Section III. In Section IV, the process of ORB-SLAM is given. Section V describes the method of maximum likelihood estimation for the scale problem. Then, simulations and experiments are described in Section VI. Finally, Section VII concludes this paper and proposes some future research works.

II. SYSTEM OVERVIEW

As shown in Figure 1, the framework of the system consists of three parts, including ORB-SLAM algorithm, the altitude information of UAV obtained by PX4-Flow module and the method based on maximum likelihood estimation which is used to improve the accuracy of the position information collected from the ORB-SLAM algorithm. Then, the UAV platform can localize the position of itself and achieve the autonomous navigation flight. In the following sections, we will expound each part of the system and implement a series of simulation and experiments to demonstrate the accuracy of the method which we presented.

*This work is supported by Shenzhen basic research program JCYJ20150731105106111.

Shiyu Chen, Yanjie Li and Haoyao Chen are with Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, 518055, China. (Corresponding author is Yanjie Li, e-mail: autolyj@hit.edu.cn).

978-1-5386-2035-9/17/$31.00 © 2017 IEEE

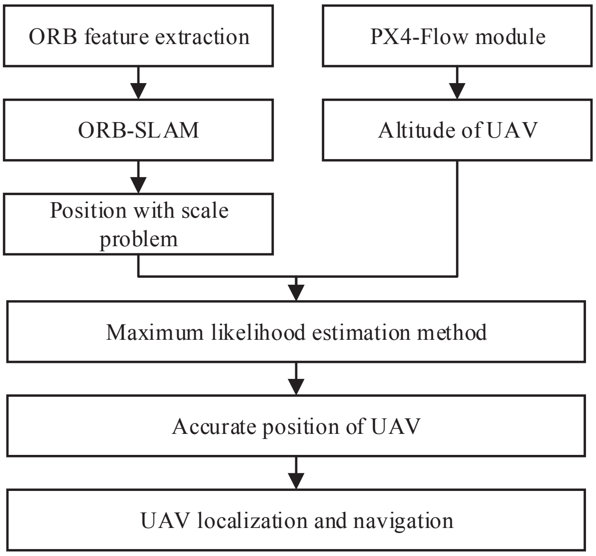

Figure. 1 System framework.

III. ORB Feature Extraction

The ORB feature is an integrated description method of images and it consists of two parts including feature detection and feature description. In the part of feature detection, FAST corner feature [11] is used as the characteristic points of interest. Because of the non-direction of the features, we add the direction information to it. In the part of feature description, we adapt the binary feature descriptor BRIEF [12] to describe the feature points. The detailed process of the ORB feature extraction is explained as follows.

A. Feature Detection

The theory of FAST feature detection is that comparing the pixel value of a feature point with the points around it. If there are enough points which have larger or smaller pixel values, we can determine this feature point as a FAST feature. As shown in Figure 2, take the pixel point p as the center of a circle with a radius of three pixel, and let I_k denotes the pixel value of the points on the circle where $k=1,2,\cdots,n$. Thus, we can judge whether the point p is a feature point through the following formula:

$$P=\begin{cases}1 & |I_p-I_k| > t \\ 0 & \text{else}\end{cases}, \tag{1}$$

where I_p is the pixel value of point p and t is a given threshold. If the number of $P=1$ is larger than 12, the point p will be considered as a FAST feature of the image.

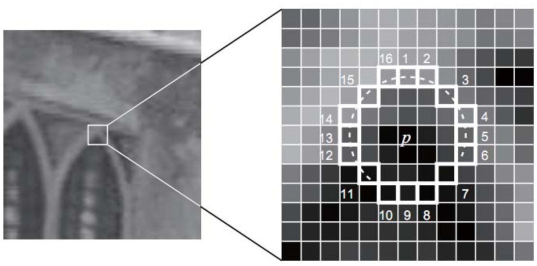

Figure. 2 FAST feature detection.

In order to solve the problem of FAST feature which is non-directional, we can obtain the direction of the features by calculating the $p+q$ order moment of the point (x,y) as follows:

$$M_{pq}=\sum_{x,y}x^p y^q I(x,y), \tag{2}$$

where $I(x,y)$ is the pixel value of point (x,y). The center coordinate C and declination direction α of the feature point can be obtained:

$$C=(C_x,C_y)=\left(\frac{M_{10}}{M_{00}},\frac{M_{01}}{M_{00}}\right), \tag{3}$$

$$\alpha=\arctan\left(\frac{C_y}{C_x}\right). \tag{4}$$

B. Feature Description

The BRIEF feature descriptor is mainly used to get a set of binary characters as the feature descriptor by comparing the pixel gray value of pixels which are selected randomly in the neighborhood of FAST features. Suppose a pixel pair is η, the pixel values of pixels γ and β are $\eta(\gamma)$ and $\eta(\beta)$, respectively. The rule of binary comparison is shown as follows:

$$\tau(\eta;\gamma,\beta)=\begin{cases}1 & \eta(\gamma)<\eta(\beta) \\ 0 & \text{else}\end{cases}. \tag{5}$$

If there are n group pixel pairs $(\gamma_i,\beta_i),\ i=1,2,\cdots,n$, the binary character can be calculated by the following formula:

$$f_n(\eta)=\sum_{1\le i\le n}2^{i-1}\tau(\eta;\gamma_i,\beta_i). \tag{6}$$

Because the BRIEF feature descriptor is obtained by comparing the gray value of the pixels, it is robust to the change of light intensity. However, the descriptor is sensitive to the rotation of the feature points. Therefore, we fuse the declination direction α of the feature points with the descriptor and then get the rotation matrix $\boldsymbol{R}_\alpha$. The matrix of the pixel pairs $(\gamma_i,\beta_i),\ i=1,2,\cdots,n$ is defined as follows:

$$\boldsymbol{S}=\begin{pmatrix}\gamma_1 & \cdots & \gamma_n \\ \beta_1 & \cdots & \beta_n\end{pmatrix}. \tag{7}$$

Thus, the new matrix of pixel pairs after rotating can be obtained by:

$$\boldsymbol{S}_\alpha=\boldsymbol{R}_\alpha\boldsymbol{S}. \tag{8}$$

The feature descriptor which is rotation-invariant can also be calculated as follows:

$$g_n(\eta,\alpha)=\sum_{1\le i\le n}2^{i-1}\tau(\eta;\gamma_i^{\alpha},\beta_i^{\alpha}), \tag{9}$$

where $(\gamma_i^{\alpha},\beta_i^{\alpha})$ denotes the pixel pairs in matrix $\boldsymbol{S}_\alpha$.

IV. The Process of ORB-SLAM

ORB-SLAM is a kind of monocular SLAM algorithm based on ORB feature. There are three parallel modules in this algorithm including feature tracking, local mapping and loop closing. All the modules use ORB feature as the basis of calculation. The process of ORB-SLAM algorithm is shown in

Figure 3, and the more detailed explanation of this algorithm can be found in the reference [8].

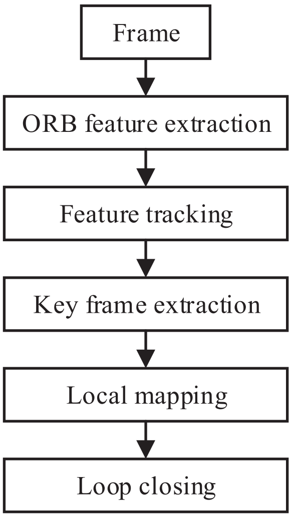

Figure. 3 The process of ORB-SLAM.

A. Feature Tracking

After extracting the ORB features of the image frame, the features should be tracked real-timely. We can obtain the current position information of the camera by comparing the corresponding relationship between the consecutive frames. If the feature is lost, the current frame image is compared with the image database for global relocation.

B. Local Mapping

In order to reduce the complexity of the map, the algorithm is mainly based on the construction of local map to complete the global map building and optimization. According to the effect of feature tracking, the information of key frame image is inserted and the new map is obtained according to the correlation between the key frames. In the process of building the local map, the redundant key frames are removed constantly according to the number of key frames and the repetition rate of the features. Thus, the computational complexity of the algorithm is reduced and the real-time property of the method is guaranteed.

C. Loop Closing

The closed loop detection of SLAM methods is a process of continuous optimization of the map obtained by the local mapping. The error can be corrected by calculating the cumulative error between the current key frame and the loop key frame. In the process of map correction, the point cloud in the map is also adjusted according to the similar transformation of the map and get more accurate map information finally.

V. Maximum Likelihood Estimation

The monocular vision SLAM algorithms are widely used as a tool for real-time positioning of UAV in the air because of the simple hardware structure. But the depth information of the scene can't be calculated precisely only depending on a monocular camera. Thus, there is a position scale problem among the monocular vision-based SLAM methods. In this paper, we adapt the theory of maximum likelihood estimation to improve the ORB-SLAM algorithm and add the depth information to the scene. The data sequence $(h_i^s, h_i^u)\ i = 1, 2, \cdots, n$ is defined to denote the altitude information of the UAV, where h_i^s denotes the altitude of the UAV measured by ORB-SLAM algorithm and h_i^u denotes the altitude measured by the ultrasonic range finder. The scale factor of the ORB-SLAM method is defined as λ, and the relationship of the measured data is shown as follows:

$$h_i^s \approx \lambda h_i^u \, . \tag{10}$$

Suppose that the measurement noise of the altitude information is in accordance with the Gauss noise, we can obtain that:

$$h_i^s \sim \mathrm{N}(\lambda\mu_i, \sigma_s^2) \, , \tag{11}$$

$$h_i^u \sim \mathrm{N}(\mu_i, \sigma_u^2) \, , \tag{12}$$

where μ_i denotes the altitude value of UAV and σ_s^2, σ_u^2 represent the measurement variance of ORB-SLAM algorithm and the ultrasonic range finder, respectively. In order to estimate the value of scale factor, we adapt the method based on maximum likelihood estimation and get the optimal value of λ^* and μ_i^* via minimizing the following likelihood function:

$$\ell(\mu_1, \cdots, \mu_n, \lambda) \propto \frac{1}{2}\sum_{i=1}^{n}\left(\frac{\| h_i^s - \lambda\mu_i \|^2}{\sigma_s^2} + \frac{\| h_i^u - \mu_i \|^2}{\sigma_u^2}\right) . \tag{13}$$

Then, the altitude value and the scale factor can be obtained by computing the minimum value of the above function, which are shown in the following formulas:

$$\mu_i^* = \frac{\lambda^* \sigma_u^2 h_i^s + \sigma_s^2 h_i^u}{\lambda^{*2}\sigma_u^2 + \sigma_s^2} \, , \tag{14}$$

$$\lambda^* = \frac{Z_{ss} - Z_{uu} + sign(Z_{su})\sqrt{(Z_{ss} - Z_{uu})^2 + 4Z_{su}^2}}{2\sigma_s^{-1}\sigma_u Z_{su}} \, , \tag{15}$$

$$Z_{ss} = \sigma_u^2 \sum_{i=1}^{n} (h_i^s)^2 \, , \tag{16}$$

$$Z_{uu} = \sigma_s^2 \sum_{i=1}^{n} (h_i^u)^2 \, , \tag{17}$$

$$Z_{su} = \sigma_u \sigma_s \sum_{i=1}^{n} (h_i^s h_i^u) \, . \tag{18}$$

VI. Simulation and Experiments

To verify the proposed improved UAV localization algorithm, we develop a UAV simulation platform and all the algorithms can be operated on it. After good performance is obtained in simulations, a series of real-time experiments have been implemented in real environment to demonstrate the robustness and accuracy of the algorithm.

A. Simulation

The simulation platform [13] is established based on Gazebo. We use the Firefly UAV model as the platform which is the product of AscTec company. A camera has been attached to the platform and the video sequence is published at 30 fps with a resolution 320*240, as shown in Figure 4. All the algorithms use robot operating system (ROS) as the interfacing robotics middleware.

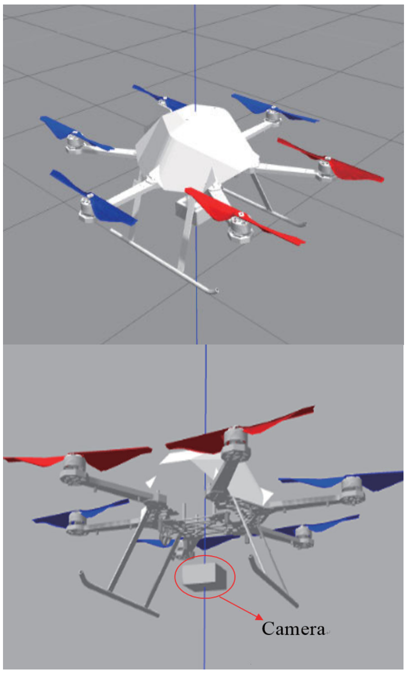

Figure. 4 The UAV simulation platform.

In the simulation environment, we make the UAV model fly to the position of $(0,0,1)$ at the coordinate system, and only use the ORB-SLAM algorithm to obtain the real-time position of the UAV platform. The result of the simulation is shown in Figure 5. From this picture we can see that the altitude of UAV platform is not correct and the position at y axis has great fluctuation. Then, we fuse the maximum likelihood estimation method with the ORB-SLAM algorithm and obtain the position of UAV which is shown in Figure 6. Compare the two pictures, we can obtain that the position of UAV platform has been improved with this method. So, the result of the simulation demonstrates that the method added to the ORB-SLAM algorithm is feasible and effective.

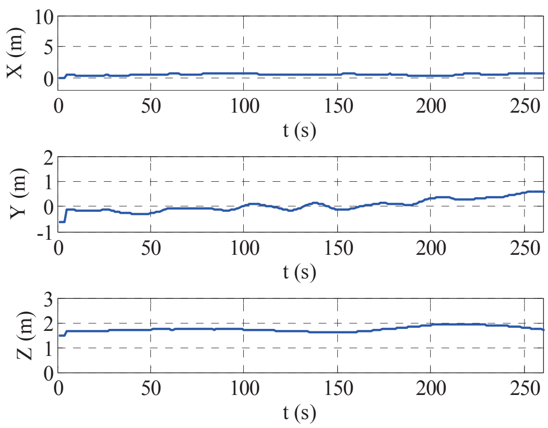

Figure. 5 The position of UAV use only ORB-SLAM.

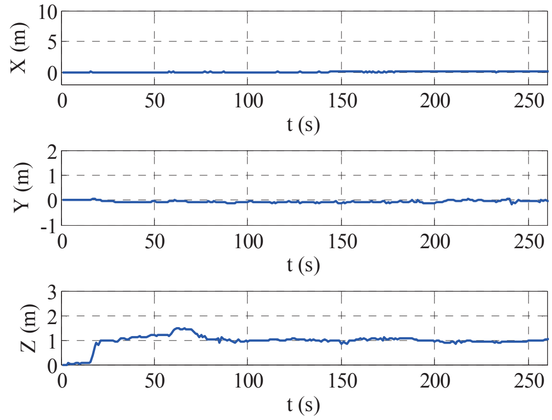

Figure. 6 The position of UAV after improved.

B. Experiments

The experimental platform is based on a quadrotor equipped with the Pixhawk autopilot consisting of an IMU and a programmable ARM-Cortex-M4 microcontroller, as shown in Figure 7. The main computation unit onboard is an Intel NUC with a 3.1 GHz Core i7 processor with 16 GB of RAM and a 256 GB SSD. A high frame rate camera PS3-Eye is used to capture real-time video sequence for the ORB-SLAM algorithm. And a PX4-Flow module is used as the ultrasonic range finder mounts on the UAV platform. ROS tool is the interface middleware for all the running algorithms. In order to avoid the time delay of the information interaction between quadrotor and images, all apps are operated on the onboard computer.

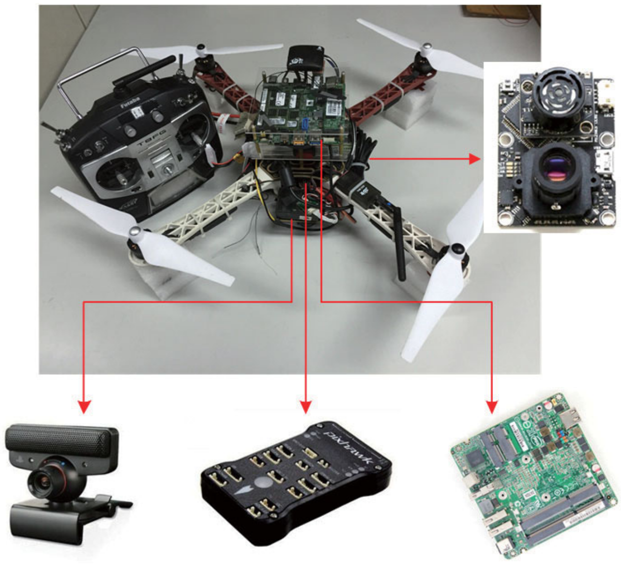

Figure. 7 The UAV platform in real environment.

After the establishment of the experimental platform, the ORB-SLAM algorithm was implemented in a corridor environment firstly. The result of the ORB feature extraction is shown in Figure 8, and the map of the corridor is shown in Figure 9. The rectangular boxes represent the position of the camera and the points cloud denote the features of the scene.

All the algorithms were working well and the improved UAV localization algorithm was verified by the hovering experiment of the UAV platform.

Figure. 8 ORB feature extraction.

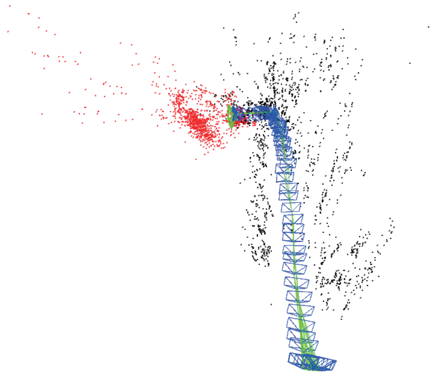

Figure. 9 The map of the environment generated by ORB-SLAM.

As shown in Figure 10, the UAV platform was hovering in the outdoor environment and the real-time position information was collected and saved in the onboard computer. The position of UAV was generated by fusing the ORB-SLAM algorithm and the altitude of the UAV obtained from PX4-Flow module. Then, we drew the position of UAV in Figure 11. Compare the values at each axis with $(0,0,1)$, we can obtain that the UAV platform was hovering steadily with a little error.

Figure. 10 The hovering experiment of UAV platform.

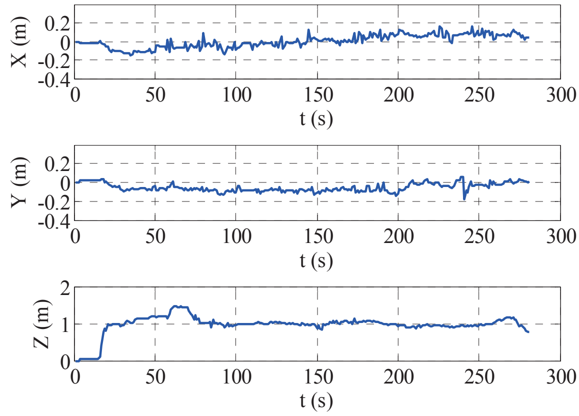

Figure. 11 The position of UAV in hovering experiment.

The hovering experiment of UAV platform indicated that the improved UAV localization algorithm can provide accurate position information for the UAV platform. The method based on maximum likelihood estimation theory was effective to improve the localization algorithm.

VII. Conclusion and Future Work

In this paper, we have proposed a method that based on the theory of maximum likelihood estimation to improve the ORB-SLAM localization algorithm. The PX4-Flow module mounted on the UAV was used to obtain the altitude information of UAV. Then, the altitude of UAV was combined with the position obtained by ORB-SLAM algorithm with a scale factor to solve the scale problem. A series of simulation and experiments were implemented to verify the robustness and accuracy of the method. Future work will focus on the control algorithms of UAV to increase its robustness and accuracy when executes a mission in complex environment. Also, the localization algorithm will be studied by combining the computer vision with machine learning.

References

[1] S. Gil, R. Milanese, and T. Pun, "Comparing features for target tracking in traffic scenes," *Pattern Recognition*, vol. 29, no. 8, pp. 1285-1296, 1996.

[2] A. Dukeman, and M. Anderson, "A CSP solution to multi-camera surveillance and target tracking," *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 97-102, 2011.

[3] D. Q. Abell, *Mobile Tracking and Location Awareness in Disaster Relief and Humanitarian Assistance Situations*, Thesis Collection, 2012.

[4] A. Soloviev, "Fusion of inertial, optical flow, and airspeed measurements for UAV navigation in GPS-denied environments," *The International Society for Optical Engineering*, pp. 733202-733212, 2009.

[5] G. Klein, and D. Murray, "Parallel tracking and mapping on a camera phone," *IEEE International Symposium on Mixed and Augmented Reality*, pp. 83-86, 2009.

[6] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," *IEEE International Conference on Robotics and Automation*, pp. 15-22, 2014.

[7] E. Rublee, V. Rabaud, K. Konolige, et al, "ORB: An efficient alternative to SIFT or SURF," *IEEE International Conference on Computer Vision*, vol. 58, no. 11, pp. 2564-2571, 2011.

[8] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “ORB-SLAM: A versatile and accurate monocular SLAM system,” *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147-1163, 2015.
[9] S. Lynen, M. W. Achtelik, S. Weiss, et al, “A robust and modular multi-sensor fusion approach applied to MAV navigation,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3923-3929, 2013.
[10] S. M. Weiss, “Vision based navigation for micro helicopters,” *Commonwealth of Pennsylvania: Doctoral Dissertation, University of Pennsylvania*, pp. 61-119, 2012.
[11] E. Rosten, T. Drummond, “Machine learning for high-speed corner detection,” *European Conference on Computer Vision*, pp. 430-443, 2006.
[12] M. Calonder, V. Lepetit, C. Strecha, et al, “Brief: Binary robust independent elementary features,” *European Conference on Computer Vision*, pp. 778-792, 2010.
[13] F. Furrer, M. Burri, M. Achtelik, et al, *RotorS—A Modular Gazebo MAV Simulator Framework. Robot Operating System (ROS)*, Springer International Publishing, 2016.