A Flying Robot Localization Method Based on Multi-sensor Fusion

Regular Paper

Changan Liu1, Sheng Zhang1, Hua Wu1 and Ruifang Dong1,*

1 School of Control and Computer Engineering, North China Electric Power University, Beijing, China
* Corresponding author E-mail: drf345@126.com

Received 27 Apr 2014; Accepted 25 Jul 2014

DOI: 10.5772/58927

© 2014 The Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract This paper proposes a novel localization method for a power-tower-inspection flying robot based on fusion of vision, IMU and GPS. First, the research background is introduced in relation to a visual localization algorithm derived from 3D-model-based tracking and a coordinate transformation model for related coordinate frames. Then, a multi-sensor fusion-based localization method is presented, in which two collaborative Kalman filters are designed to fuse IMU/GPS and visual information. Finally, experimental results are presented to show the robustness and precision of the proposed method.

Keywords Flying Robot, Power-tower Inspection, Multi-sensor Fusion-based Localization

1. Introduction

Traditional artificial power-tower inspection suffers from both inefficiency and safety problems in harsh geographical environments. Despite the fact that the use of helicopters may now reduce inspection risks, pilot safety still cannot be guaranteed. Thus, autonomous flying robot inspection, with the advantage of both greater accuracy and safety, has become a major trend in power-tower inspection [1, 2].

A critical issue in autonomous-flying-robot inspection is pose estimation. Commercial GPS-based localization only provides accuracy at between 2 and 10 m, which not only constitutes a great threat to safety, but also means it is difficult to maintain interacted spatial relationship between tower and flying robot. Visual localization turns out to be an effective alternative. However, camera shaking, edge ambiguity and background complexity may degrade robustness and flexibility of visual pose estimation.

In order to promote the effectiveness of localization, this paper proposes a new method that can improve both precision and robustness. Section 2 gives a quick review of recent related research. Section 3 presents a short description of a 3D-model-based tracking system and a related coordinate transformation model involving IMU, GPS and corresponding frames. Section 4 explains a new localization approach based on multi-sensor fusion. Experimental results are demonstrated in Section 5, followed by our conclusions in the last section.

2. Related Research

Vision-based localization derives from multiple geometry. Hartley [3] presented classic ways of recovering camera pose using a set of correspondent
features extracted from several image sequences; this is known as structure from motion. PTAM (Parallel Tracking and Mapping) is an excellent implementation of the algorithm by Klein [4]. Nützi [5] applied this method on a real flying robot, using an IMU sensor to acquire absolute scale estimation. Shen [6] built a system based on a fisheye camera and another ordinary camera, along with IMU information to yield state estimates. The author proposed a vision-based estimation approach that combines the advantages of monocular vision and stereo vision in a complex environment. However, simplicity and repeatability of background texture makes it hard to extract and match features. In 1990, Harris [7] proposed a 3D wireframe model-based tracking algorithm, which is theoretically suitable for power-tower inspection as it best fits the rigid structure of a tower. Lepetit [8] surveyed all model-based tracking methods and offered some advice for future work. Later, Addrew [9] innovatively combined robust control law and model-based tracking, developing a more generalized toolbox for real-time vision servo application. However, edge ambiguity and motion blur, which often occur during tower inspection due to instability of the flying robot, may directly lead to a tracking failure.

For these two problems, researchers have developed many solutions. For example, the visual algorithm by Choi [10], which involves a particle filter, and the multiple-edge hypothesis verification by Teulière [11]. These authors share the idea of constructing a hypothesis-searching space for all the possible edge combinations to calculate the best pose to handle edge mismatches. Kempter [12] created a system that integrates PTAM and a model-based tracking system, and which automatically switches between two systems according to tracking situation and condition. However, the most suitable solution for power-tower inspection is multi-sensor fusion. With the help of Kalman filters, one can fuse the sensor data from GPS and IMU to correct visual pose and gain optimal pose estimation. Klein [13] used a gyroscope to acquire angular velocity to provide an initial pose for the next frame tracking. Klein also proposed a parametric-blur edge detector with angular rate and camera exposure time as parameters. However, only camera rotations were considered and translation was still estimated from Kalman filters. You [14] chose a flat and distinguished landmark for tracking, which does not work for power-tower inspection.

For power-tower inspection, tracking failure and wrong pose estimation may happen because of variation of viewpoint and camera shaking. Translation and rotation may happen at the same time. Hence, more sensors should be deployed, rather than only using a simple visual sensor. However, GPS/IMU-based localization cannot reach high precision. Therefore, we propose a new localization approach that fuses IMU/GPS and vision information, and which has the following advantages: (1) System error can be reduced and error accumulation due to edge ambiguity can be avoided by providing an initial pose guess for 3D-model-based tracking in the current frame. (2) The random error can be reduced by using a framework of two collaborative Kalman filters. This framework has the ability to automatically switch between IMU/GPS/aided localization and IMU/GPS/vision combined localization in accordance with the tracking condition. It is also capable of maintaining localization function and visual tracking recovery, whilst and after the visual tracking is unavailable, respectively.

3. Backgrounds

3.1 3D Model-based Visual Localization

An existing model-based tracking [9] method is brought into our method to perform pose estimation for the flying robot. On the one hand, a model of the power tower can be obtained off-line, for example a CAD model or LIDAR scan. The rigid structure of the power tower also fits model-based tracking. On the other hand, model-based visual localization is more accurate than pure GPS localization. The measurement unit is metres, which can be easily employed for inspection path planning.

The algorithm works as follows: Consider a point belonging to a 3D object with coordinate \( P_r \) in object frame and coordinate \( P_i \) in the projected image frame. Knowing the camera’s intrinsic parameter \( \xi \), pose \( r \) can be calculated as in Eq. (1).

\[
\text{obj. min} \left( \text{proj}(r, \xi, P_r) - P_i \right)
\]

(1)

Through tracking the image in a relative pose between the camera and the tower \( E = H_{\text{Tower} \rightarrow \text{Camera}} r \) can be acquired, which satisfies Eq. (2).

\[
\begin{align*}
P_{\text{Camera}} & = H_{\text{Tower} \rightarrow \text{Camera}} P_{\text{Tower}} \\ H_{\text{Tower} \rightarrow \text{Camera}} & = \begin{bmatrix} R_{03} & t_{03} \\ \theta_{03} & 1_{03} \end{bmatrix}
\end{align*}
\]

(2)

\( P_{\text{Tower}} \) stands for a point coordinate in the tower frame; \( P_{\text{Camera}} \) indicates the same point in the camera frame. \( H_{\text{Tower} \rightarrow \text{Camera}} \) is an extrinsic matrix, describing transformations between the above two frames. \( R \) is a rotation matrix, and \( t \) is a translation matrix. Here, we can get the coordinate of the camera centre in the tower frame in formula (3).

\[
P_{\text{CameraCentre}}^{\text{Tower}} = -R^t t
\]

(3)

This coordinate will be treated as one component of system state variables in subsequent computations.
3.2 Coordinate transformation

The approach proposed in this paper requires fusing of the data from IMU, GPS and visual pose estimation. Thus, a coordinate transformation is necessary. According to the chain rule of transformation, the overall transformation is divided into five successive frames: (1) world frame; (2) power-tower frame; (3) inertial frame; (4) robot body frame; (5) camera frame.

The transformation matrix between power-tower frame and world frame, which can be pre-calibrated manually as \( H_{\text{Tower} \rightarrow \text{World}} \) when tracking starts, stays constant as in Eq. (4):

\[
P_{\text{World}} = H_{\text{Tower} \rightarrow \text{World}} P_{\text{Tower}}
\]

\( P_{\text{World}} \) and \( P_{\text{Tower}} \) present the same point in the world frame and power-tower frame, respectively.

During the inspection, the flying robot is always changing its pose relative to the tower. The transformations between camera frame and power-tower frame are therefore dynamic, and can be described as in Eq. (5):

\[
P_{\text{Camera}} = H_{\text{Tower} \rightarrow \text{Camera}} P_{\text{Tower}}
\]

where

\[
H_{\text{Tower} \rightarrow \text{Camera}} = \begin{bmatrix} R_{\text{body} \rightarrow \text{camera}} & t_{\text{body} \rightarrow \text{camera}} & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} = 0 \begin{bmatrix} 0 & 1 \end{bmatrix} R_{\text{tower} \rightarrow \text{camera}} t_{\text{tower} \rightarrow \text{camera}}
\]

In formula (6), \( t_{\text{tower} \rightarrow \text{camera}} \) and \( t_{\text{camera} \rightarrow \text{body}} \) have to be calculated in real time, while the other variables can be pre-calibrated. \(-R_{\text{tower} \rightarrow \text{camera}} R_{\text{camera} \rightarrow \text{body}}\) is equal to the coordinate of the origin of the inertial frame in terms of the power-tower frame, for which \( t_{\text{tower} \rightarrow \text{camera}} \) can be obtained with data from GPS using Eq. (4), assuming that the origin of inertial frame coincide with GPS-positioned point. \( R_{\text{camera} \rightarrow \text{body}} \) can be calculated using roll, pitch and yaw from IMU through Eq. (7).

\[
R_{\text{camera} \rightarrow \text{body}} = R(\alpha, \beta, \gamma) = R(\gamma) R(\beta) R(\alpha) = R_{\gamma} R_{\beta} R_{\alpha}
\]

where \( \gamma \) is roll, \( \beta \) is pitch, \( \alpha \) is yaw. \( R_{\gamma}, R_{\beta}, R_{\alpha} \) are basic rotation matrices about the x, y and z axes, respectively.

\( H_{\text{Tower} \rightarrow \text{Camera}} \) is obtained as in Eq. (2). Using Eq. (6), \( R_{\text{camera} \rightarrow \text{body}} \) can be easily obtained. Euler angles from visual localization algorithm are finally estimated as in Eq. (8):

\[
\begin{align*}
\psi &= -\tan^{-1}\left(\frac{-R_{31}}{R_{30}}\right) \\
\theta &= -\tan^{-1}\left(\frac{R_{32}}{\sqrt{R_{32}^2 + R_{33}^2}}\right) \quad \text{where} \quad R = \begin{bmatrix} R_{11} & R_{21} & R_{31} \\
R_{12} & R_{22} & R_{32} \\
R_{13} & R_{23} & R_{33} \end{bmatrix} \\
\phi &= -\tan^{-1}\left(\frac{-R_{20}}{R_{22}}\right)
\end{align*}
\]

where \( \psi \) is roll, \( \theta \) is pitch, and \( \phi \) is yaw. These three variables partially constitute the flying robot’s system state.

There may exist different solutions to Eq. (8) when it comes to the implementation of arctangent. In general, raw output of roll pitch and yaw stay with \((-\pi, \pi), (-\frac{\pi}{2}, \frac{\pi}{2}), (0, 2\pi)\). Specifically during the inspection, however, range of pitch and roll is \((-\frac{\pi}{2}, \frac{\pi}{2})\). Accounting for specific ranges of Euler angles, by using built-in atan and atan2 function in programming languages like C++, roll, pitch and yaw can be recovered as in Eq. (9). In the third formula of Eq. (9), the atan2 results should be mapped to \((0, 2\pi)\) by adding \(2\pi\) and taking a modulo operation of \(2\pi\), before timing negative one.

\[
\begin{align*}
\psi &= -\text{atan}(\frac{R_{31}}{R_{30}}) \\
\theta &= -\text{atan}(\frac{R_{32}}{\sqrt{R_{32}^2 + R_{33}^2}}) \\
\phi &= -2\text{atan}(\frac{R_{20}}{R_{22}})
\end{align*}
\]

4. Multi-Sensor Fusion-based Localization

Purely model-based tracking for pose estimation is executed for successive frames, which also means that the pose estimated in the current frame depends on the one in the last frame. However, due to the complexity of the tower model, when inspecting the tower from a different view, separated edges project closely together in the image. When the tracking algorithm searches the edges, multiple matching points will be found, as shown in Figure 1. If that happens only for few frames, the error stays within an acceptable range. If no pose correction is carried out, however, error will be accumulated, resulting in tracking failure.

In order to solve the problem, fusion of GPS and IMU based on Kalman filters is used to correct the pose for visual tracking.

The Kalman filter method is designed as follows. Say the system state variable is \( X_{k} \), as shown in formula (10):

\[
X_{k} = (x_{k}, y_{k}, z_{k}, v_{xk}, v_{yk}, v_{zk}, \psi_{k}, \theta_{k}, \phi_{k}, \alpha_{k}, \beta_{k}, \gamma_{k})^{T}
\]

\[
X_{k} = (x_{k}, y_{k}, z_{k}, v_{xk}, v_{yk}, v_{zk}, \psi_{k}, \theta_{k}, \phi_{k}, \alpha_{k}, \beta_{k}, \gamma_{k})^{T}
\]

Figure 1. Edge ambiguity

Image edges

Model projected edge

Image edges

Search direction

Changan Liu, Sheng Zhang, Hua Wu and Ruifang Dong: A Flying Robot Localization Method Based on Multi-sensor Fusion
is flying speed, and $(\psi_{x,i}, \psi_{y,i}, \psi_{z,i}, \omega_{x,i}, \omega_{y,i}, \omega_{z,i})^T$ is the Euler angles and angular rate of the flying robot.

For a high sampling frequency, it is reasonable to consider that during interval time $\Delta T$, the flying robot is moving at a constant speed. Predictions and updates are given in Eq. (11):

$$
\begin{align*}
X_{t+1}^i &= F_t X_t^i \\
P_{t+1}^i &= F_t P_t^i F_t^T + Q_t \\
y_t &= z_t - H_t X_{t+1}^i \\
S_t &= H_t P_t^i H_t^T + R_t \\
K_t &= P_t^i H_t^T S_t^{-1} \\
X_{t+1}^i &= X_{t+1}^i + K_t y_t \\
P_{t+1}^i &= (I - K_t H_t) P_t^i
\end{align*}
$$

where $F_t$ is state transformation model, $P_t$ is state covariance matrix, $Q_t$ is process noise covariance matrix, $R_t$ is measurement covariance matrix, and $H_t$ is mapping matrix of measurement, and $z_t = (x_t, y_t, z_t, \psi_{x,i}, \psi_{y,i}, \psi_{z,i})^T$ is observation matrix. In the following section, the pose of the flying robot is noted as $x_t = (x_t, y_t, z_t, \psi_{x,i}, \psi_{y,i}, \psi_{z,i})^T$.

The whole procedure of the proposed method is depicted in Figure 2. The details are as follows:

1. Recursively estimate a posteriori pose $x_{t+1}^*$ from last frame. Priori Pose $x_t$ is predicted in current frame.
2. Gather data from IMU and GPS as observations, marked as $z_t$.
3. Update pose with observations in (2) and get a posteriori pose $x_{t+1}^*$. 
4. Set pose in (3) as an initial pose for visual tracking. Kalman filter for IMU/GPS ends here. 
5. Track the model one more time to get a pose for the current frame. If it is untrackable, the pose from (4) is directly used as the new pose.
6. The pose from (5) is regarded as the observation for the vision Kalman filter. After one time update, final pose for current frame is estimated.

Two collaborative Kalman filters are used: (1) Kalman filters can reduce the side effect of random noise; (2) pose correction from IMU/GPS helps to avoid error accumulation for visual tracking; (3) visual-tracking-based pose estimation has higher precision than pure IMU/GPS.

Model-based tracking cannot recover itself once it fails. Thus, when facing with a motion blur, which makes it hard to identify enough feature points, the whole system has to be switched into IMU/GPS-aided localization. As long as the error stays within a specific range, visual tracking can automatically recover when the images become clear. It is worth noting that IMU/GPS-aided localization will increase error to some degree. But when visual tracking is available, error will be narrowed again.

![Figure 2. Fusion-based method](image)

Figure 3 depicts one frame of the tracking procedure. In the IMU/GPS Kalman filter, we get a re-projected edge using predicted pose. Projection with observations means projected edge using purely IMU/GPS data. Through one filter update, a filter-updated pose is calculated. Then, one more tracking is performed to get the visual tracking pose, indicated as a projection with the tracking update. The final edge is located closer to the real-image edge, indicating a higher precision.

![Figure 3. Snapshot for stable visual tracking](image)

5. Experiments and analysis

In order to prove the accuracy and robustness of the proposed method, a comparison between visual localization and fusion-based localization is made.

The whole experiment is carried out as in Figure 2. The experimental device is shown in Figure 4. A flying robot
equipped with IMU and a camera with a 640*360 resolution at 30 FPS is used. The tracking target is a mini power-tower model. The operating system is Ubuntu Linux 12.04. The programs run on a computer with Intel Core i5 2.5GHz and 4G RAM. An IGPS (Indoor Global Positioning System) is employed, simulating a real GPS.

As mentioned before, purely visual tracking may fail due to motion blur or edge ambiguity. In order to prove the effectiveness of the proposed method, the experiment focuses on these two specific occasions.

5.1 Switching Condition

It is key to measure tracking quality so as to determine whether or not switch to IMU/GPS-aided tracking. For 3D-model-based visual-tracking pose estimation, corresponding points sampled along the edges for two successive frames are used to calculate the pose. Therefore, enough feature points must be used for pose estimation. Because of motion blur or camera shaking, this number may fall rapidly, which disables visual tracking. Here, points available for pose estimation are defined as Trackable Points (TP), and those not available are defined as UnTrackable Points (UTP). In order to ensure visual-tracking accuracy, a threshold needs to be configured.

\[
\text{Threshold} < \frac{\text{TP}}{\text{TP} + \text{UTP}}
\]  

When Eq. (12) is satisfied, tracking is considered to be good; otherwise, the system will switch to IMU/GPS-aided localization. In the experiment, the threshold is set to 0.7, which performs well under this circumstance.

5.2 Edge Ambiguity

Purely model-based tracking suffers from error accumulation out of edge ambiguity, as shown in Figure 5, in which the white frame shows the visual localization result, showing that the error has accumulated hugely, deviating from the real tower target. The red frame stands for fusion-based localization, in which visual pose is corrected by IMU/GPS to avoid error accumulation, demonstrating higher accuracy and robustness.

As described above, for evaluation of tracking quality, measurement of TP is involved. The amount of TP is somewhat proportional to tracking quality and localization accuracy. Figure 6 depicts the process from stable tracking to error accumulation due to edge ambiguity. TP of fusion-based localization is apparently higher than that of pure visual localization.

5.3 Camera Shaking

Flying robots may inevitably suffer from sudden camera shaking. The experimental result for camera shaking is shown in Figure 7: the white frame stands for pure visual localization and the red frame for fusion-based localization. As shown, vertical shaking causes edge blurring, and pose difference between two frames is too big, making it unable to find point matches. During shaking, TP is shown in Figure 8. TP for fusion-based localization stays high, whereas for pure visual localization, it drops enormously. The proposed method therefore has better performance.

Figure 9 indicates the tracking recovery result after tracking failure due to severe motion blur. The red frame shows the recovery consequence as aided by IMU/GPS; pure visual localization, marked with the white frame, is clearly unable to recover the right pose.
6. Conclusion and future work

In this paper, we propose a new multi-fusion-based localization method for a power-tower-inspection flying robot, in which two collaborative Kalman filters are designed to fuse IMU/GPS and visual information for higher accuracy and robustness. The experimental results show the advantage of being immune to camera shaking and edge ambiguity, meeting the requirements for power-tower inspection.

It needs to be stressed that when the error of IMU/GPS stays within a particular range, the proposed method has relatively high accuracy and stability. However, if the system error for IMU/GPS gets too large, not following a Gaussian distribution, or fluctuates randomly with great magnitude, the initial pose provided by IMU/GPS will not be acceptable, which means visual tracking cannot depend on IMU/GPS anymore. Also, switching to IMU/GPS-aided localization will lose localization precision.

Our future work will have two aims. One is to use an accelerometer in the IMU to partially replace GPS for greater reliability. The other is to fuse the particle filters. With the help of sensors like the IMU, the search space for particle filters will be able to be constrained for better performance.

7. Acknowledgement

This work is supported by the Programme for New-Century Excellent Talents in University (Grant No. NCET-11-0634), the National Natural Science Foundation of China (Grant No. 61105083) and the Fundamental Research Funds for the Central Universities (Programme No. 12ZX16).

8. References

[1] Liu, C.A., Wang, L., Liu, C.Y. (2009) Mission planning of the flying robot for power line inspection. Progress in Natural Science, 19: 1357-1363.
[2] Hua Wu, Min Lv, Chang-An Liu, Chun-Yang Liu. (2012) Planning efficient and robust behaviors for model-based power tower inspection. IEEE International Conference on Applied Robotics for the Power Industry 2012, Sep 11 -13, Zurich, Switzerland. pp. 163-166.
[3] Hartley, R., Zisserman, A. (2004) Multiple View Geometry in Computer Vision, Second Edition. Cambridge University Press, March 2004, Cambridge, UK, pp. 262-278.
[4] Klein, G. and Murray, D. (2007) Parallel Tracking and Mapping for Small AR Workspaces. 6th IEEE and ACM International Symposium on Mixed and Augmented Reality 2007, Nov 13-16, Nara, Japan. pp. 225-234.
[5] Nützi, G., Weiss, S., Scaramuzza, D., Siegwart, R. (2012) Fusion of IMU and Vision for Absolute Scale Estimation in Monocular SLAM. Journal of Intelligent & Robotic Systems, 61: 287-299.
[6] Shen, S., Mulgaochar, Y., Michael, N., Kumar, V. (2013) Vision-Based State Estimation for Autonomous Rotorcraft MAVs in Complex Environments. International Conference on Robotics and Automation 2013, May 6-10, Karlsruhe, Germany. pp. 1758-1764.
[7] Harris, C., Stennett, C. (1990) RAPIR – A Video Rate Object Tracker. Proceedings of British Machine Vision Conference 1990, Oxford, England. pp. 73-78.
[8] Lepetit, V., Fua, P. (2005) Monocular Model-Based 3D Tracking of Rigid Objects: A Survey. Foundations and Trends in Computer Graphics and Vision, 1: 1-89.
[9] Andrew I. Comport, Marchand, E., Chaumette, F. (2004) Robust model-based tracking for robot vision. IEEE International Conference on Intelligent Robots and Systems 2004, Sep 28-Oct 2, Sendai, Japan. pp. 692-697.
[10] Choi, C., and Christensen, H.I. (2011) Robust 3D Visual Tracking Using Particle Filtering on the SE(3) Group. International Conference on Robotics and Automation 2011, Sep 25-30, San Francisco, CA, USA. pp. 4384-4390.
[11] Teulière, C., Eck, L., Marchand, E., Guénard, N. (2010) 3D model-based tracking for UAV position control. International Conference on Intelligent Robots and Systems 2010, Oct 18-22, Taipei, Taiwan. pp. 1084-1089.
[12] Kempter, T., Wendel, A., Bischof, H. (2012) Online Model-Based Multi-Scale Pose Estimation. Proceedings of 17th Computer Vision Winter Workshop 2012, Jan 1-3, Mala Nedelja, Slovenija. pp. 1-8.
[13] Klein, G.S.W., Drummond, T.W. (2004) Tightly integrated sensor fusion for robust visual tracking. Journal of Image and Vision Computing, 22: 769-776.
[14] You, S., Neumann, U. (2001) Fusion of vision and gyro tracking for robust augmented reality registration. Proceedings of the IEEE Conference on Virtual Reality 2001, Mar 17-17, Yokohama, Japan. pp. 71-78.