IMAV2016-12
http://www.imavs.org/pdf/imav.2016.12

Autonomous Positioning and Navigation for UAV in GPS-Denied Environments

Xiaolong Zhang*, Zhihong Peng, and Hongyang Liu
School of Automation, Beijing Institute of Technology, Beijing, 100081, China
State Key Laboratory of Intelligent Control and Decision of Complex System, Beijing, 100081, China
School of Automation, Beijing Institute of Technology, Beijing, 100081, China

Abstract

A platform integrating visual camera into Unmanned Aerial Vehicle (UAV) system to realize the function of autonomous positioning and navigation in GPS-denied environments is presented. Through motion tracking and depth perception, the visual camera can provide estimated information on the surrounding environments for UAV to navigate. A control method of pose estimation and navigation strategy is proposed to make this system more stable and efficient, and to enhance its robustness. Several experimental tests have been carried out, illustrating the validity of the proposed approach.

1 Introduction

Recent years have witnessed a wide range of autonomous applications of UAVs. UAVs have obtained lots of attention as low-cost, flexible, high-efficiency platforms covering vast regions and providing widespread field of view. UAVs always take field operations, including environment exploration, intelligence gathering, searching and rescuing victims and etc. All the jobs UAVs can do are mostly based on autonomous positioning and navigation technologies which rely on GPS and have reached its maturity. However, there are plenty of possibilities that UAVs are not able to access GPS in complex environments. In such cases, GPS signal is not available because of the poor weather condition or the block of obstacles.

Several studies focus on how to locate an UAV and guide it to desired waypoints in GPS-denied environments. Navigation strategy with RGB-D camera is applied in [1, 2]. [3] proposes a navigation research in GPS-denied environment based on monocular-SLAM technology. [4] proposes a visual navigation system for an UAV using optical flow in a GPS-denied environment. A framework for Google Map aided UAV navigation in GPS-denied environment is used by [5]. It applies geo-referenced navigation to provide drift-free localization. [6] presents a vision-aided inertial navigation system for small UAV and develops a delay-based approach for visual measurement in GPS-denied environments.

Consumer-grade technology seen in cameras and tablet has led to price-performance ratio falling dramatically over the last decade. Google, for example, is developing a customized tablet named Project Tango equipped with conventional sensors, like inertial measurement units (IMU), as well as depth cameras and it has significant computational capability. This paper constructs a programmable platform with this tablet as a visual sensor and chooses Pelican quad-rotors produced by Ascending Technologies as the experimental carrying body. Additionally, a personal computer (PC) plays the role of a ground station to make the system more efficient and flexible. Afterwards, this paper puts forward a method of its pose estimation and autonomous positioning and navigation in GPS-denied situation based on the visual camera.

The rest of this paper is organized as follows. Section 2 introduces a dynamic model for quad-rotors UAV. Section 3 shows how to construct the programmable platform in terms of hardware and software respectively. Section 4 proposes a control method of pose estimation and autonomous positioning and navigation strategy of UAV. Section 5 concludes the work of this paper.

2 Modeling

A quad-rotors is a system consists of four identical rotors and propellers located at the vertices of a square (see Figure 1).

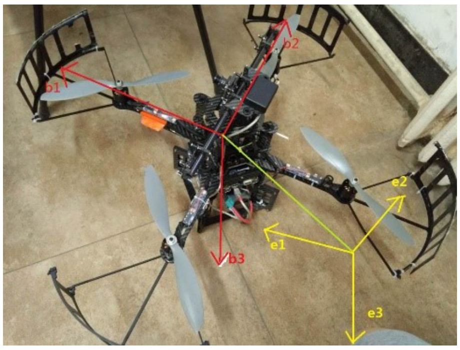

Figure 1: A quad-rotors.

Considering an inertial reference frame donated by $\{e_1, e_2, e_3\}$ and a body reference frame centered in the center of mass (COM) of the quad-rotors donated by $\{b_1, b_2, b_3\}$, a dynamic model can be expresses as

*Email address(es): zhangxiaolong23@sina.cn

IMAV 2016, Beijing, PR of China, 17-21 October 2016
International Micro Air Vehicle Competition and Conference 2016

$$\dot{x} = v$$
$$m\dot{v} = -R\tau e_3 + mge_3$$
$$\dot{R} = R\widehat{\Omega}$$
$$J\dot{\Omega} + \Omega \times J\Omega = M \tag{1}$$

where $x \in R^3$ is the Cartesian position of the quad-rotors related to inertial frame, $v \in R^3$ is the velocity of the quad-rotors in the inertial frame, $m \in R$ is the mass, $\Omega \in R^3$ is the angular velocity of the quad-rotors in the body frame, $J \in R^3$ is the inertia matrix with respect to the body frame, and $R \in SO(3)$ is the rotation matrix with respect to the body frame. $M \in R^3$ being the total moment along all axes of the body frame and the thrust $F \in R$ are control inputs of the plant. g is the standard gravitational acceleration with $e_3 = [0 \;\; 0 \;\; 1]^T$. The hat symbol $\widehat{*}$ means the skew-symmetry operator as $x\widehat{y} = x \times y, \forall x, y \in R^3$. Assuming that the force of each propeller is directly controlled [7] , the relationship among F, M and single-motor force $f_i, i = 1, 2, 3, 4$ can be expressed as

$$\tau = f_1 + f_2 + f_3 + f_4$$
$$M_1 = df_2 - df_4$$
$$M_2 = df_1 - df_3$$
$$M_3 = -cf_1 + cf_2 - cf_3 + cf_4 \tag{2}$$

where d is the distance from the center of each rotor to COM and c is a constant value.

3 System structure

3.1 Hardware structure

As previously mentioned, this paper chooses Project Tango tablet as a visual sensor, Pelican quad-rotors as a carrying body and PC as a ground station to construct a platform. Its structure is seen in Figure 2. These three parts are connected to each other through a wireless connection, which can make the system flexible, simple and easily debugged.

The Project Tango has such three core technologies as motion tracking, depth perception and area learning which we can take advantage of to allow the platform to know where it is and what its surroundings are, then to predict a way to move on.

The Pelican quad-rotors is equipped with an Autopilot board which consists of IMUs and two ARM7 processors named Low Level Processor (LLP) and High Level Processor (HLP) respectively. The LLP handles data fusion and other underlying operations. The HLP is programmable for our algorithm and it has serial communication interfaces to connect the ground station.

The Project Tango is attached to the Pelican physically.

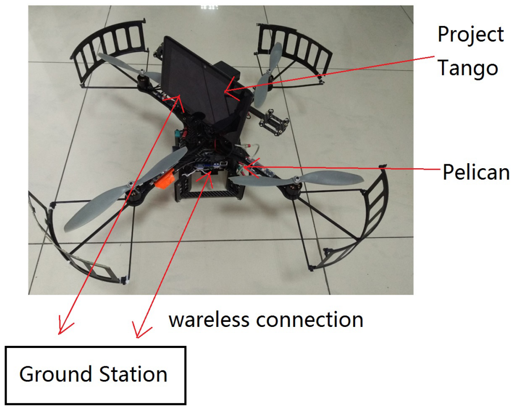

Figure 2: Hardware structure.

3.2 Software structure

The autonomous flight controlled system of the Pelican is designed to be a double loop controlled system, as shown in Figure 3. A proportional integral derivative (PID) controller in the inner control loop is fixed in the hardware of Pelican and receives information from the remote control (RC), while an integral separation PID controller is adopted in the external loop, where Project Tango acts as a visual sensor.

Being attached to the Pelican, the tablet can monitor its real-time status and compare it to the desired status. The external controller receives these error items as its own input while its output can be the given amounts of the internal controller instead of the control information from the RC. Finally, the autonomous control and navigation of the UAV platform can be enhanced by integrating the estimated information of surrounding environments.

As previously mentioned, three parts of this system are connected to each other over a wireless connection (see Figure 4). A android application routine is enabled on the tablet for pose and depth data streaming to the ground station using the user datagram protocol (UDP) which transfers data more efficiently and is more reliable.

As shown in Figure 5, the UDP packets are received by the ground station and are subsequently processed into usable commands which are eventually transferred to Pelican through universal asynchronous receiver/transmitter (UART). The Pelican receives commands from ground station to replace the control information from the RC to control the flight.

4 Control method

4.1 Pose estimation

The ability to estimate pose accurately and efficiently is very important to navigation in GPS-denied environments.

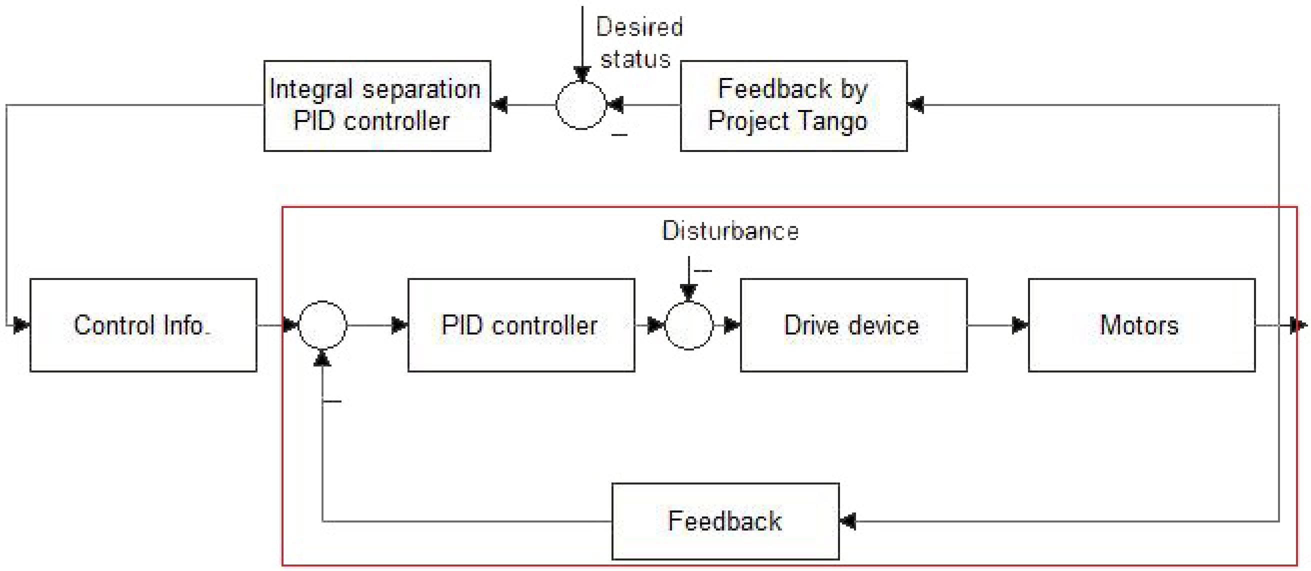

Figure 3: Autonomous flight control of the Pelican.

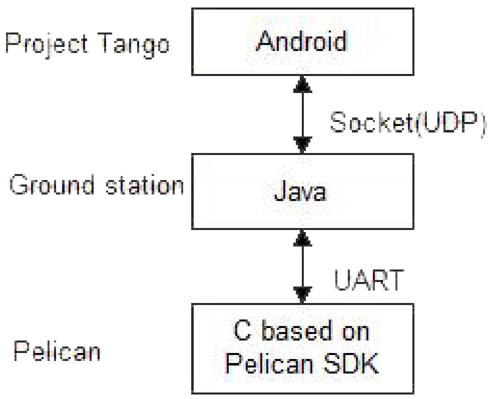

Figure 4: Software structure.

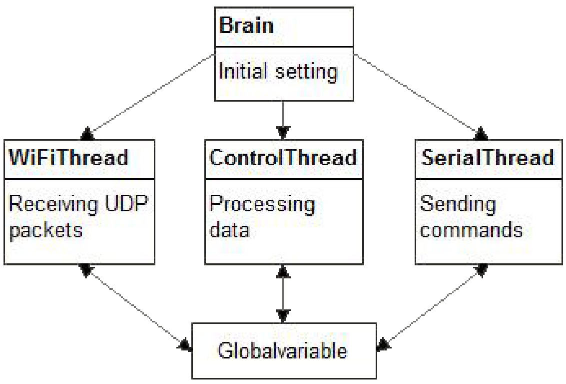

Figure 5: Ground station.

Pose data is accessible from Project Tango and so is Pelicans pose by processing these data since they are attached to each other and in the same coordinate system.

It is available to read position data with respect to the s-tart point from Project Tango. By mean filter of continuous returned N data $p_i, i = 1, 2, , N$ to eliminate the influence of random disturbance [7] , the current position p_{now} of the Pelican with respect to the start point can be expressed as

$$p_{now} = \frac{1}{N} \sum_{i=0}^{N} p_i \tag{3}$$

Similarly, quaternion data can also be obtained from Project Tango which characterizes the attitude of Pelican. Doing the real-time math to solve the quaternion [7] , the attitude information will be obtained from

$$\phi = \arctan \frac{2(q_0 q_1 + q_2 q_3)}{1 - 2(q_1^2 + q_2^2)}$$

$$\theta = \arcsin(2(q_0 q_2 - q_1 q_3))$$

$$\psi = \arctan \frac{2(q_0 q_3 + q_1 q_2)}{1 - 2(q_2^2 + q_3^2)} \tag{4}$$

where q_0 is the real part of the quaternion and q_1, q_2, q_3 are the imaginary parts, ϕ is the yaw angle, θ is the roll angle, and ψ is the pitch angle, respectively. These angles are relative angles between the inertial reference frame and the body reference frame.

4.2 *Navigation strategy*

This paper uses the depth perception technology of Project Tango to put forward a navigation strategy. A point cloud can be obtained by developing the Project Tango, like Figure 6.

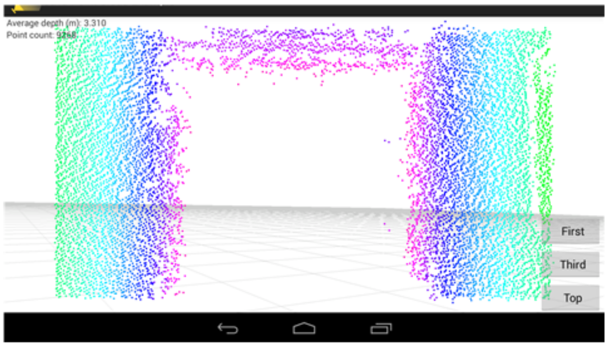

Figure 6: Point cloud.

The cloud point shown in Figure 6 means a three-dimensional (3-D) coordinate estimation which presents the relative positional relationship between the mapped point in real world and the Project Tango. According to this, the platform can realize the depth information of surroundings. Fixed by a recognition algorithm, it can learn which direction has enough space to go forward, additionally, detect and avoid obstacles.

4.3 *Control algorithm*

The platform can obtain its current state through pose estimation, and the future state it should achieve through navigation strategy. The next task is to consider how to control the conversion of state.

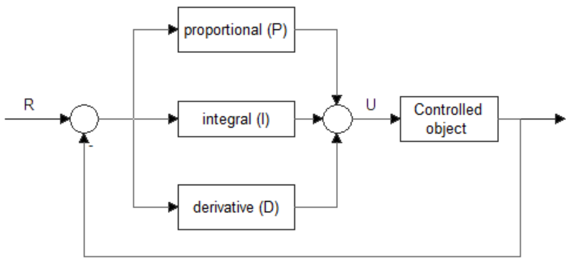

Figure 7: PID algorithm.

$$U(k) = k_p e(k) + k_i \sum_{j=0}^{k} e(j) + k_d(e(k) - e(k-1)) \quad (5)$$

What can be found is that the approach which is put forward above can always convert into some kinds of error control, which can be achieved by using the PID control algorithm (see Figure 7 and (5)) where $e(k)$ is the error, $U(k)$ is the output control amount, k_p is for present values of the error, k_i is for past values of the error, k_d is for possible future values of the error, respectively. However, there is one common problem resulting from the ideal PID implementations, which is the integral windup. Following a large change in set point, the integral term can accumulate an error much larger than the maximal value for the regulation variable (windup), thus the system overshoots and continues to increase until this accumulated error is unwound.

So, an integral separation PID control is adopted (see Figure 3), which is the improvement of the ideal PID and can be expressed as

$$U(k) = k_p e(k) + \beta k_i \sum_{j=0}^{k} e(j) + k_d(e(k) - e(k-1))$$

$$\beta = 1, e(k) \leq \varepsilon$$

$$\beta = 0, e(k) > \varepsilon \quad (6)$$

where ε is the separation limit, integral action will be performed only when $e(k) \leq \varepsilon$ to eliminate the integral windup.

4.4 *Experiment*

The experiment is based on the following assumptions:

(1)Choosing narrow space like corridor as experiment environment.

(2)Setting the initial position of Pelican artificially.

(3)The planning track for Pelican is flying in the horizontal midline while maintaining a certain height and returning if it reaches the end of the corridor.

Based on the above assumptions, this paper puts forward a specific recognition algorithm. What should to do first is to abstract Figure 6 to Figure 8 as shown below.

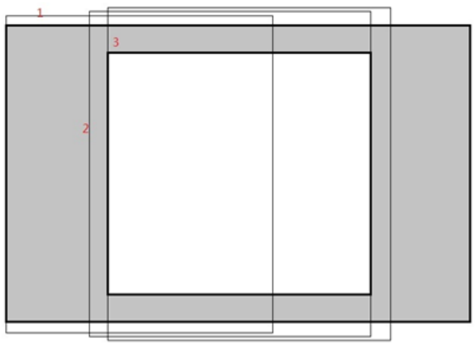

Figure 8: Abstract image.

The followings name the white rectangle unknown rectangle (UR) and the rectangle, which is filled with shadow and UR, known rectangle (KR). There exists UR because there is a limit of detection distance of the Project Tango so that it can not detect all the points in front of itself. Conversely, those points covered by the shadow are those that can be detected.

Assuming that the head of Pelican is always parallel to the axis of the corridor, the problem that should be solved is the relative positional relationship between UR and KR. Doing this by following these steps.

(1)Recording 95% of the counts of those points that can not be detected in KR as Nc.

(2)Constructing a rectangle which is big enough to cover the UR, moving it from left to right with fixed step and recording the counts of those points that can not be detected in it as Ni.

(3)Recording the left boundary of the moving rectangle when Ni is bigger than Nc for the first time as L and the right boundary of the moving rectangle when Ni is bigger than Nc for the last time as R. $m = \frac{L+R}{2}$ will be the midline of UR. With the already known midline M of KR, we can obtain the relative positional relationship between UR and KR.

Additionally, to make sure that the head of Pelican is always parallel to the axis of the corridor, what should be done is to operate real-time monitor of the change of Pelicans yaw angle and make the change always smaller than a threshold.

In order to detect the end of the corridor, we can record the real-time average depth in the field of view. Actually, the average depth will hop when Pelican reaches the end of the corridor (see Figure 9).

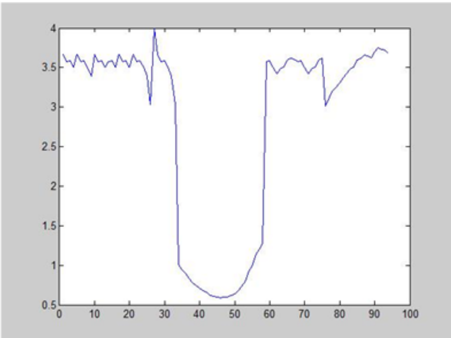

Figure 9: Average depth in the field of view.

As previously mentioned, the strategy which is put forward above can convert into error control, which can be achieved by using the integral separation PID algorithm.

In this experiment, the height of Pelican can stay in a relatively small range near a fixed value (see Figure 10). Additionally, the platform can find and reach the midline of the corridor, and the track of its horizontal position can coincide with the axis of the corridor in a small margin of error (see Figure 11).

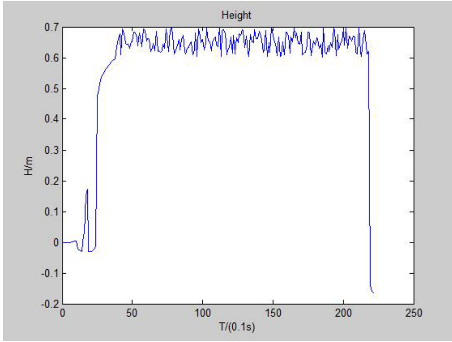

Figure 10: Height curve during the experiment.

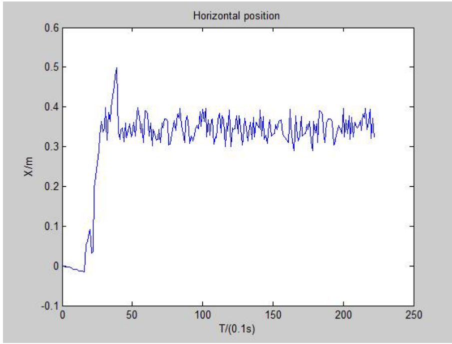

Figure 11: Horizontal position curve during the experiment.

5 Conclusion

This paper constructs a programmable UAV platform and demonstrates the hardware and software structure with the proposed control method to enable the autonomous positioning and navigation for the platform in simple GPS-denied environments. Serval experiments are carried out and some certain results are obtained.

Future work will take more complicated GPS-denied environments into account.

References

[1] Albert S Huang, Abraham Bachrach, Peter Henry, Michael Krainin, Daniel Maturana, Dieter Fox, and Nicholas Roy. Visual odometry and mapping for autonomous flight using an rgb-d camera. In *International Symposium on Robotics Research (ISRR)*, volume 2, 2011.

[2] Giuseppe Loianno, Vincenzo Lippiello, and Bruno Siciliano. Fast localization and 3d mapping using an rgb-d sensor. In *Advanced Robotics (ICAR), 2013 16th International Conference on*, pages 1–6. IEEE, 2013.

[3] Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Monocular-slam–based navigation for autonomous micro helicopters in gps-denied environments. *Journal of Field Robotics*, 28(6):854–874, 2011.

[4] Lan Huang, Jian Mei Song, Pu Hua Chen, and Gao Hua Cai. Visual navigation for uav using optical flow estimation. In *Control Conference (CCC), 2014 33rd Chinese*, pages 816–821. IEEE, 2014.

[5] Mo Shan, Fei Wang, Feng Lin, Zhi Gao, Ya Z Tang, and Ben M Chen. Google map aided visual navigation for uavs in gps-denied environment. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 114–119. IEEE, 2015.

[6] Tianmiao Wang, Chaolei Wang, Jianhong Liang, Yang Chen, and Yicheng Zhang. Vision-aided inertial navigation for small unmanned aerial vehicles in gps-denied environments. *International Journal of Advanced Robotic Systems*, 10, 2013.

[7] Nathan Michael, Daniel Mellinger, Quentin Lindsey, and Vijay Kumar. The grasp multiple micro-uav testbed. *IEEE Robotics & Automation Magazine*, 17(3):56–65, 2010.