

A Trajectory Planning and Control System for Quadrotor Unmanned Aerial Vehicle in Field Inspection Missions

Gang Chen¹, Rong Wang², Wei Dong¹, and Xinjun Sheng¹(✉)

¹ State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
xjsheng@sjtu.edu.cn

² Shanghai Pujiang Bridge and Tunnel Management Co. Ltd., Shanghai 200231, China

Abstract. In this work, a trajectory planning and control system designed for quadrotor unmanned aerial vehicles (UAVs) in automatic field inspection missions is presented. This system controls the UAV by two steps. First, a ground station laptop generates a complete coverage flight path regarding the specific working field. Subsequently, an onboard computer controls the UAV in real-time with position, velocity and acceleration setpoints calculated by a trajectory planning algorithm. The algorithm considers both the dynamic property of quadrotor UAVs and the acceleration and velocity limitations in real working condition. UAV controlled by this system performs a rapid and stable flight in windy environment and can be widely utilised in various domains.

Keywords: Quadrotor UAV · Complete coverage path · Trajectory planning · Trajectory tracker

1 Introduction

The quadrotor UAV is a popular type of flying robot with compact mechanical structure, high load capacity, capability of hovering and low cost. Due to its unique advantages, the quadrotor UAV is utilized in various application scenarios. In industrial inspection field, it is applied in power line inspection [1], bridge defect detection [2] and solar power plant measurement [3]. With UAVs, inspection work can be safer, lower-cost and much more efficient. Also, there are great demands for quadrotor UAVs in precision agriculture. Loading appropriate equipment, they can monitor and analyse the quality and health condition of growing crops [4, 5]. Furthermore, quadrotor UAVs are utilized in traffic monitoring [6] and public safety [7] domains.

In some of the scenarios mentioned above, such as bridge defect detection, solar power plant measurement and crop health monitoring, the UAV is desired

to fulfil an inspection mission in specified field. The whole field should be covered with minimal time cost. Stable and smooth flight is also required in order to achieve continuous inspection result. Currently, most UAVs are manually controlled to perform such kind of missions, which requires professional UAV pilots and causes high manpower cost. Therefore, the corresponding automatic flight control system is demanded.

The main algorithm in this control system is a trajectory planning algorithm. First a complete coverage path should be generated to cover the working field. Coverage path planning algorithm research starts from ground robots, such as cleaning robots [8] and farming robots [9]. Later, Valente J et al. designed a coverage path planning algorithm for quadrotor UAVs working on irregular fields measured by images. The algorithm divides the field into squares and treat the issue as a travel salesman problem [10]. Franco et al. proposed an energy-aware coverage path planning method for UAVs [11]. These methods concern the general path to cover the field, but ignore the real-time control of the quadrotor UAV. Y. Bouktir et al. researched on the dynamic properties of quadrotor UAVs and raised a trajectory planning method based on B-spline functions [12]. Mark W. Mueller et al. proposed a computationally efficient trajectory generation algorithm for quadrotor UAVs flying along a straight line [13]. These algorithms perform excellently in simulation or indoor experiment environment. In view of the state-of-art, the control of quadrotor UAVs in real working condition is still required to be thoroughly considered to improve the performance.

Motivated by this consideration, a trajectory planing and control system for quadrotor UAV is proposed. The system has competence to generate complete coverage flight path and calculate real-time setpoints to control the UAV. The hardware of this system is composed of a flight control unit (FCU), an onboard computer and a ground station laptop. The ground station laptop generates coverage path based on the Global Position System (GPS) geography information of the working field in a practical approach. The onboard computer calculates real-time trajectory setpoints through an improved algorithm and controls the UAV to realize a stable and efficient flight.

The rest of this paper is arranged as follows. Section 2 introduces how the whole system functions. Section 3 describes the complete coverage polyline path generation method. Section 4 presents a trajectory planning algorithm on each line and a trajectory tracker for real-time control. Section 5 shows the experiment results and Sect. 6 states the conclusion.

2 System Overview

The basic flight of a quadrotor UAV is controlled by FCU. An open-source FCU named PX4 is chosen in this program [14]. With OFFBOARD control mode in PX4, the UAV can be controlled by acceleration setpoints received from serial port. One approach is to transmit the real-time setpoints calculated by the ground station laptop with wireless transmitter, such as XBee. However, smooth flight can not be achieved because of the terrible packet dropout phenomenon. Therefore, an onboard computer is utilized to calculate the real-time setpoints.

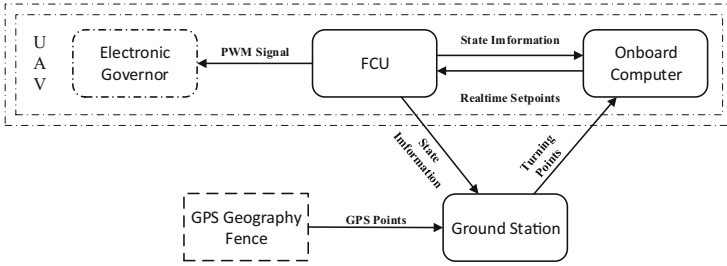


Fig. 1. System framework

The whole system is operated as Fig. 1 shows. Firstly, a GPS geography fence of the working field is measured and imported to the ground station laptop. Then the ground station generates a complete coverage polyline path based on the working field and the current position of the UAV. The generated turning points, namely the endpoints of each line, are transmitted to the onboard computer through wireless transmitter. After that, the onboard computer calculates the real-time trajectory setpoints on the basis of the path generated by ground station and the flight state information given by FCU. Finally, the FCU receives the setpoints and controls the UAV to fly as expected. When the mission changes, the ground station generates another coverage path and sends to the onboard computer to proceed new mission. This system allows the UAV operators to change missions on the ground anytime they want and realize smooth automatic flight control.

3 Complete Coverage Ployline Path Generation

To cover the selected working field, a complete coverage polyline path is generated on the ground station laptop with geography fence. Geography fence is usually measured by a smartphone or a professional GPS device. The fence consists of GPS points and forms a polygon after converting from global position system to local position system. In Fig. 2, the blue polygon is the imported fence. The flight height of the UAV is always settled in particular mission, thus only a two dimensional path is considered in this paper.

Firstly, the line equation defined by two adjacent corner points on the fence polygon can be calculated. Denote corner point $A = (x_1, y_1)$ and $B = (x_2, y_2)$. To avoid infinite result during calculation, the equation of Line AB is described as the standard form

$$ax + by = c \tag{1}$$

where $a = (y_2 - y_1)$, $b = (x_1 - x_2)$ and $c = x_1y_2 - x_2y_1$. The endpoints of each line are also recorded in a matrix in the same order as lines parameters.

Denote reference working width d_r and the flight direction $\vec{\omega} = (x_\omega, y_\omega)$, where d_r is often given by the property of the UAV and $\vec{\omega}$ is given according to the requirement in specific situation. For example, solar panels inspection mission

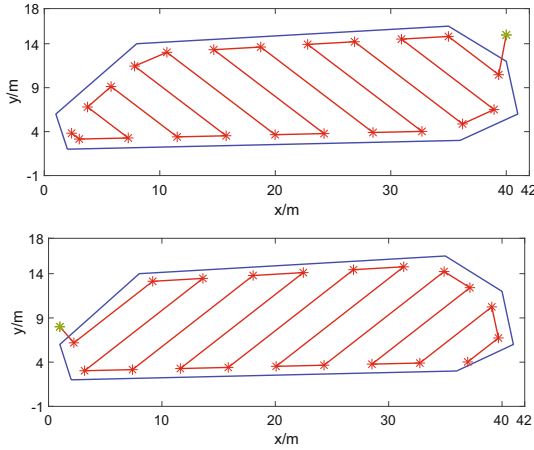


Fig. 2. Complete coverage polyline path. Blue polygon in the figure is the geography fence for the working field and red polylines are the generated flight path. Fly direction is $(-1, 1)$ in the upper figure and $(1, 1)$ in the below figure. Green star represents the position of the UAV at this moment. Red stars are the turning points. (Color figure online)

in solar power plant requires the UAV to fly along the direction the panels are ranged and d_r is width that a image of the inspection camera could cover. In crop inspection or fertilization mission, fly direction is along the direction the crops are planted.

Any line along direction $\vec{\omega}$ can be written as

$$y_\omega x - x_\omega y = m \tag{2}$$

Substitute corner points of fence polygon into Eq. 2 and find the maximum and minimum value of variable m , namely m_{max} and m_{min} . The number of the parallel lines required in the ployline flight path n can be calculated by

$$n = \left\lceil \frac{m_{max} - m_{min}}{\sqrt{y_\omega^2 + x_\omega^2} \times d_r} \right\rceil + 1 \tag{3}$$

From small to large value of m with interval $\delta m = (m_{max} - m_{min}) \div n$, crossover points of the parallel lines and fence polygon are given by solving the following equation for each parallel path line and boundary line of the polygon.

$$\begin{pmatrix} a & b \\ y_\omega & -x_\omega \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} c \\ m_{min} + \frac{\delta m}{2} + k\delta m \end{pmatrix} \tag{4}$$

where a and b are the variables in Eq. 1, $k \in \{1, 2, \dots, n\}$ and valid result (x, y) should be on the boundary of the fence polygon. The precondition of solving Eq. 4 is

$$\begin{vmatrix} a & b \\ y_\omega & -x_\omega \end{vmatrix} \neq 0 \tag{5}$$

which indicates that the flight path line and the boundary line can not be parallel. After the crossover points are acquired, move those points a distance of $\frac{\delta m}{2}$ from boundary to cut the overlap area. Finally, rearrange the sequence of crossover points based on the current position of the UAV. The result is shown in Fig. 2.

4 Trajectory Planning and Real-Time Control

After the ground station generates polylines flight path and transmits to the onboard computer, trajectory planning and real-time control threads start in sequence. These threads are realized in the onboard computer. Trajectory planning is an off-line work to generate the desired acceleration, velocity and position of the UAV on the whole flight path, while real-time control adjust and send set-points to FCU to control the actual behaviour of the UAV.

4.1 Trajectory Planning on Line

The polyline flight path can be divided into lines. On each line, the UAV should fly through three periods, acceleration period, uniform velocity period and deceleration period. In a real working situation, uniform velocity is always expected to achieve a excellent inspection result. This part focus on the algorithm to realize a smooth and rapid flight with long uniform velocity period on each line.

The algorithm Mark W. Mueller et al. proposed is established on the dynamic characteristic of quadrotor UAVs by employing Pontryagin’s minimum principle [13]. Three parameters α , β and γ are derived to acquire position, velocity and acceleration setpoints on a single line path as follows

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \frac{1}{T^5} \begin{pmatrix} 60T^2 & -360T & 720 \\ -24T^3 & 168T^2 & -360T \\ 3T^4 & -24T^3 & 60T^2 \end{pmatrix} \begin{pmatrix} \delta a \\ \delta v \\ \delta p \end{pmatrix} \tag{6}$$

where T is the expected flight time, a, v and p represent acceleration, velocity and position respectively. Denote initial state $s_0 = \{p_0, v_0, a_0\}$ and final state $s_f = \{p_f, v_f, a_f\}$. $\delta a, \delta v$ and δp in Eq. 6 are given by

$$\begin{pmatrix} \delta a \\ \delta v \\ \delta p \end{pmatrix} = \begin{pmatrix} a_f - a_0 \\ v_f - v_0 - a_0T \\ p_f - p_0 - v_0T - \frac{1}{2}a_0T^2 \end{pmatrix} \tag{7}$$

Then the flight jerk, namely the derivative of acceleration, is calculated by

$$j(t) = \frac{1}{2}\alpha + \beta t + \gamma, t \in [0, T] \tag{8}$$

Furthermore, position, velocity and acceleration setpoints at time t can be expressed as

$$\begin{pmatrix} p(t) \\ v(t) \\ a(t) \end{pmatrix} = \begin{pmatrix} \frac{\alpha}{120}t^5 + \frac{\beta}{24}t^4 + \frac{\gamma}{6}t^3 + \frac{a_0}{2}t^2 + v_0t + p_0 \\ \frac{\alpha}{24}t^4 + \frac{\beta}{6}t^3 + \frac{\gamma}{2}t^2 + a_0t + v_0 \\ \frac{\alpha}{6}t^3 + \frac{\beta}{2}t^2 + \gamma t + a_0 \end{pmatrix}, t \in [0, T] \tag{9}$$

The algorithm realizes a smooth and rapid accelerating and decelerating process, but it is not really suitable for the inspection mission. The flight time T in this algorithm is calculated by dividing flight distance by average velocity. If the average velocity increases, T would be smaller and a large maximum velocity would be generated. For example, when $s_0 = \{0, 0, 0\}$, $s_f = \{20, 0, 0\}$ and $T = 5$ s, the average velocity is 4 m/s while the maximum velocity is up to 7.5 m/s. However, in the real working condition, there are always limitations on both velocity and acceleration due to the safety and specific mission requirements in outdoor flight environment, such as stability in wind, emergency brake dealing with unexpected obstacles, and the maximum detection speed determined by real-time image processing methods. Under velocity and acceleration limitations, we hope to achieve a high average velocity to increase working efficiency. Therefore, a piecewise trajectory generation algorithm is proposed and discussed in this paper.

When the length of a single line on the path is short, with acceleration limitation a_{max} , the quadrotor UAV would not reach the allowed maximal velocity v_{max} . Therefore, suppose the length threshold is s_{th} , when the real line length is shorter than s_{th} , the algorithm described from Eqs. 6 to 9 can be applied simply by calculating time T , which will be given in Eq. 15. If the real line length is longer than s_{th} , the newly proposed piecewise trajectory generation algorithm is utilized.

In our mission, the UAV flies along the path line by line, thus the initial and final acceleration and velocity are all zero. Denote the initial position p_0 and the final position p_f , Eqs. 8 and 9 can be respectively transformed into

$$j(t) = 360 \frac{d}{T^5} t^2 - 360 \frac{d}{T^4} t + 60 \frac{d}{T^3}, t \in [0, T] \tag{10}$$

$$\begin{pmatrix} p(t) \\ v(t) \\ a(t) \end{pmatrix} = \begin{pmatrix} \frac{6d}{T^5} t^5 - \frac{15d}{T^4} t^4 + \frac{10d}{T^3} t^3 1ex \\ \frac{30d}{T^5} t^4 - \frac{60d}{T^4} t^3 + \frac{30d}{T^3} t^2 1ex \\ \frac{120d}{T^5} t^3 - \frac{180d}{T^4} t^2 + \frac{60d}{T^3} t 1ex \end{pmatrix}, t \in [0, T] \tag{11}$$

In order to get the maximum acceleration on a line flight, let the jerk, the derivative of acceleration be zero. Eq. 10 can be rewritten as

$$\frac{360s_{th}}{T^5} (t - Tt + \frac{T^2}{6}) = 0 \tag{12}$$

Solve Eq. 12 and take accelerating period into consideration, the answer is

$$t_m = \frac{3 - \sqrt{3}}{6} T \tag{13}$$

Substitute t_m into acceleration equation in Eq. 11, the maximum acceleration can be described as

$$a(t_m) = \frac{120s_{th}}{T^5} (t_m)^3 - \frac{180s_{th}}{T^4} (t_m)^2 + \frac{60s_{th}}{T^3} t_m \approx 5.77 \frac{s_{th}}{T^2} \tag{14}$$

$a(t_m)$ should be smaller than a_{max} , thus we can get the minimum flight time T_s to satisfy the acceleration limitation.

$$T_s = \sqrt{\frac{5.77s_{th}}{a_{max}}} \quad (15)$$

According to symmetry of velocity in Eq. 11, the maximum velocity exists at time $t = \frac{T}{2}$.

$$v\left(\frac{T}{2}\right) = \frac{30d}{T^5} \frac{T^4}{2} - \frac{60d}{T^4} \frac{T^3}{2} + \frac{30d}{T^3} \frac{T^2}{2} = \frac{15}{8} \frac{s_{th}}{T} \leq v_{max} \quad (16)$$

where $\frac{s_{th}}{T}$ represents the average velocity. Hence the maximum velocity is $\frac{15}{8}$ times of the average velocity. Replace T with T_s given by Eq. 15 and consider the threshold state, the relational expression between a_{max} , v_{max} and s_{th} is expressed as

$$s_{th} = \frac{8}{15} v_{max} \sqrt{\frac{5.77s_{th}}{a_{max}}} \quad (17)$$

Then the distance threshold s_{th} can be calculated by

$$s_{th} = \frac{64}{225} v_{max}^2 \frac{5.77}{a_{max}} \approx 1.64 \frac{v_{max}^2}{a_{max}} \quad (18)$$

Therefore, given the limitations v_{max} and a_{max} by experience in real flight environment, distance threshold s_{th} can be derived. When the coverage polyline path in Sect. 3 is generated, compare the length of each line between two turning points. If the length s is smaller than s_{th} , we use Eq. 11 and T is acquired from Eq. 15 by replace s_{th} with actual length s . Otherwise, the flight trajectory is calculated separately in three periods as the following Period 1 to Period 3. The total fight time is $T_s + T'$. An example state curve is shown in Fig. 3.

Period 1 Accelerating period. The position, velocity and acceleration setpoints are calculated by

$$\begin{pmatrix} p(t) \\ v(t) \\ a(t) \end{pmatrix} = \begin{pmatrix} \frac{\alpha}{120} t^5 + \frac{\beta}{24} t^4 + \frac{\gamma}{6} t^3 + p_0 \\ \frac{\alpha}{24} t^4 + \frac{\beta}{6} t^3 + \frac{\gamma}{2} t^2 \\ \frac{\alpha}{6} t^3 + \frac{\beta}{2} t^2 + \gamma t \end{pmatrix}, \quad t \in [0, \frac{T_s}{2}] \quad (19)$$

where T_s is derived by Eq. 18 and α , β and γ are given by

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \frac{1}{T_s^5} \begin{pmatrix} 720s_{th} \\ -360s_{th}T_s \\ 60s_{th}T_s^2 \end{pmatrix} \quad (20)$$

Period 2 Uniform velocity period. In this period, the acceleration remains zero and velocity remains v_{max} . The state values are

$$\begin{pmatrix} p(t) \\ v(t) \\ a(t) \end{pmatrix} = \begin{pmatrix} p_0 \frac{s_{th}}{2} + v_{max}(t - \frac{T_s}{2}) \\ v_{max} \\ 0 \end{pmatrix}, \quad t \in [\frac{T_s}{2}, \frac{T_s}{2} + T'] \quad (21)$$

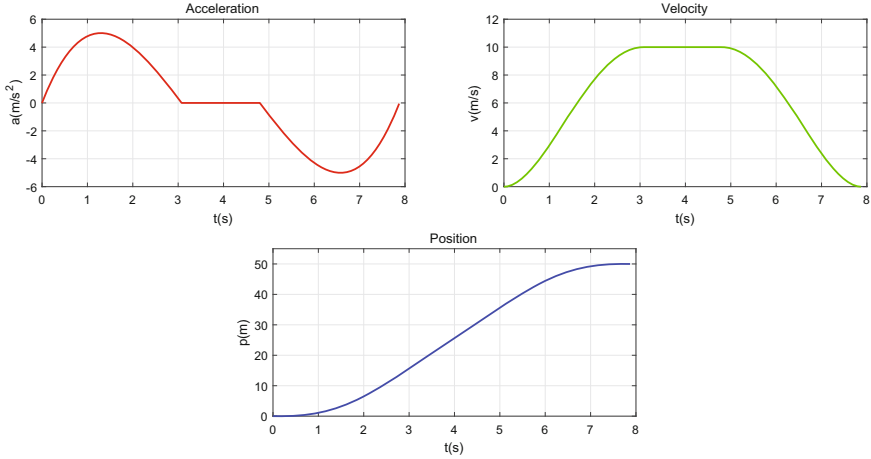


Fig. 3. A trajectory curve on a line 50 m in length. The red curve, green curve and blue curve respectively represent acceleration, velocity and position. Set $v_{max} = 10$ m/s and $a_{max} = 5$ m/s². The derived s_{th} is 32.8 m and T_s is 6.15 s. (Color figure online)

where T' is the time for this period and is described as

$$T' = \frac{s - s_{th}}{v_{max}} \tag{22}$$

Period 3 Decelerating period. The velocity decelerate from v_{max} to 0. With the same α , β and γ in Eq. 20, the state values are expressed as

$$\begin{pmatrix} p(t) \\ v(t) \\ a(t) \end{pmatrix} = \begin{pmatrix} \frac{\alpha}{120}(t-T')^5 + \frac{\beta}{24}(t-T')^4 + \frac{\gamma}{6}(t-T')^3 + p_0 + v_{max}T' \\ \frac{\alpha}{24}(t-T')^4 + \frac{\beta}{6}(t-T')^3 + \frac{\gamma}{2}(t-T')^2 \\ \frac{\alpha}{6}(t-T')^3 + \frac{\beta}{2}(t-T')^2 + \gamma(t-T') \end{pmatrix}, \tag{23}$$

$$t \in [\frac{T_s}{2} + T', T_s + T']$$

4.2 Trajectory Tracker for Real-Time Control

When the flight trajectory is derived, the position, velocity and acceleration setpoints are saved in the onboard computer. The FCU receives only acceleration setpoints, which is directly related to the attitude of a quadrotor UAV. Thus we need a trajectory tracker to control the UAV to follow the desired position, velocity and acceleration regardless of wind effect. The output of the tracker is only acceleration setpoints. The tracker is designed as Fig. 4 shows.

5 Experiment

To verify the algorithm in real environment, a quadrotor UAV with onboard computer Raspberry Pi is utilized, as is shown in Fig. 5. The weight of the UAV

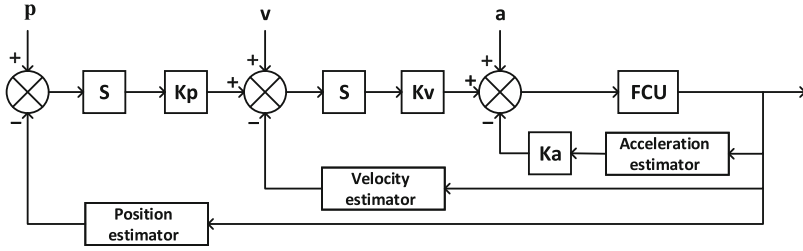


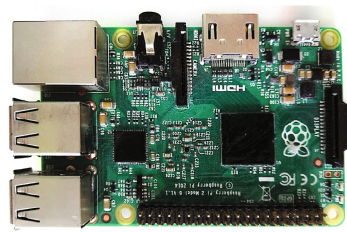
Fig. 4. Trajectory tracker. p , v and a are respectively position, velocity and acceleration setpoint. k_p , k_v and k_a are amplification coefficients.

is 2.1 kg and the axle base is 400 mm. Raspberry Pi runs the algorithm introduced in Sect. 4. Besides, a ground station software is programmed to generate the complete coverage path introduced in Sect. 3 and to monitor the real-time status of the UAV. The software is shown in Fig. 6. In a GPS available outdoor environment, a piece of grass land is chosen as the test field. In real working condition, the UAV is always loaded with particular equipment and becomes less flexible. Therefore, 1.0 kg load is mounted on the UAV and the thrust weight ratio is reduced from 2.38 to 1.61. Considering the flight safety in the wind, the velocity limitation is set to be 5 m/s and acceleration limitation is set to be 3 m/s². Firstly, the geography fence is measured by a GPS gauge and transmitted into the ground station software. Then the path and trajectory is generated as Sect. 2 introduced. Finally, the UAV is controlled by the setpoints received from onboard computer.

The experiment results are shown in Fig. 7, which are the position maps drawn in ground station software. Due to the manual take-off, the start position is a little different from the planned position. In Fig. 7(a), the average position error in gentle breeze condition (3°) is less than 0.2 m. The maximum position error is about 0.3 m when a gust of wind occurred. In Fig. 7(b), the maximum position error is about 0.4 m in strong wind condition (5–6°). The error is acceptable in application scenarios.



(a) The experiment quadrotor UAV

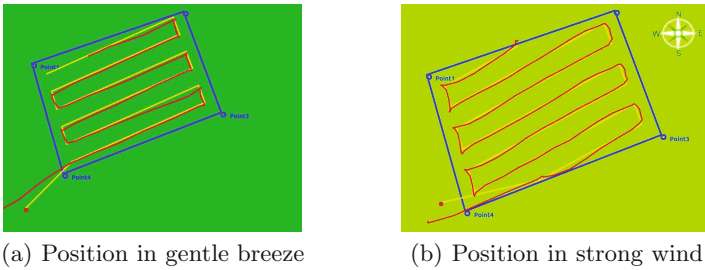


(b) The onboard computer Raspberry Pi 2B+

Fig. 5. Experiment hardware



Fig. 6. Ground station software



(a) Position in gentle breeze

(b) Position in strong wind

Fig. 7. Fly position map in experiments. Blue lines are the GPS fence. Yellow lines are the expected path and red lines are the real flight path. (Color figure online)

The current UAV products with automatic field inspection control system are very few. Here, *MG-1* designed by the famous multirotor UAV company DJI is chosen as the contrast. *MG-1* is utilized in agricultural domain. It has an automatic flight mode which flies polylines based on a given fixed length of parallel lines. Hence it only fits rectangle fields. It can fly as fast as 8 m/s. Choose the identical maximum velocity and apply in our test UAV. The size of the test field is 24 m × 55 m. Set the working width to 4 m and 6 m, the working speed result of *MG-1* and our system is presented in Table 1. The thrust weight ratio of *MG-1* is 1.78, which is higher than our UAV. However, with this control system, the working speed of our UAV are respectively 22.2% and 25.0% higher than that of *MG-1*.

Table 1. Working speed comparison

| UAV system | Working width | Working speed | Working width | Working speed |
|------------|---------------|---------------|---------------|---------------|
| MG-1 | 4 m | 2.7 ha./h | 6 m | 4.0 ha./h |
| MG-1 | 4 m | 3.3 ha./h | 6 m | 5.0 ha./h |

6 Conclusion

In this paper, a trajectory planning and control system for quadrotor UAVs working in field inspection missions is proposed. The hardware of this system is composed of a FCU, a ground station laptop and an onboard computer. In order to achieve an efficient inspection work, a complete coverage path generation method operating in the ground station laptop and a piecewise trajectory planning algorithm running in the onboard computer are respectively designed. With this system, the UAV could fly stably and rapidly to cover the whole working field. The maximum position error in windy environment is less than 0.5 m and the working efficiency is higher than that of commercial UAV products. In future, further researches will be carried to reduce the position error and the time cost. More experiments and improvements will be made to acquire more excellent performance in real working condition.

References

1. Luque-Vega, L.F., Castillo-Toledo, B., Loukianov, A., Gonzalez-Jimenez, L.E.: Power line inspection via an unmanned aerial system based on the quadrotor helicopter. In: Mediterranean Electrotechnical Conference, pp. 393–397 (2014)
2. Chen, J., Wu, J., Chen, G., Dong, W., Sheng, X.: Design and development of a multi-rotor unmanned aerial vehicle system for bridge inspection. In: Kubota, N., Kiguchi, K., Liu, H., Obo, T. (eds.) ICIRA 2016. LNCS (LNAI), vol. 9834, pp. 498–510. Springer, Cham (2016). doi:[10.1007/978-3-319-43506-0_44](https://doi.org/10.1007/978-3-319-43506-0_44)
3. Matsuoka, R., Nagusa, I., Yasuhara, H., Mori, M., Katayama, T., Yachi, N., Hasui, A., Katakuse, M., Atagi, T.: Measurement of large-scale solar power plant by using images acquired by non-metric digital camera on board uav. ISPRS - Int. Arch. Photogram., Remote Sens. Spat. Inf. Sci. **39**, 435–440 (2012)
4. Tabanlıoğlu, A., Yucedag, A.C., Tuysuz, M.F., Tenekeci, M.E.: Multicopter usage for analysis productivity in agriculture on gap region. In: Signal Processing and Communications Applications Conference, pp. 1102–1105 (2015)
5. Malveaux, C., Hall, S.G., Price, R.: Using drones in agriculture: unmanned aerial systems for agricultural remote sensing applications. In: 2014 Montreal, Quebec Canada, 13 July–16 July 2014, p. 1. American Society of Agricultural and Biological Engineers (2014)
6. Bizin, I., Vlasov, V., Demidov, A., Kanatnikov, N.: The use of multicopters for traffic monitoring in dense urban areas. In: Proceedings of the 2014 Conference on Electronic Governance and Open Society: Challenges in Eurasia, pp. 42–44. ACM (2014)
7. Baker, R.E.: Combining micro technologies and unmanned systems to support public safety and homeland security. J. Civil Eng. Archit. **6**(10), 1399 (2012)
8. De Carvalho, R.N., Vidal, H., Vieira, P., Ribeiro, M.: Complete coverage path planning and guidance for cleaning robots. In: 1997 Proceedings of the IEEE International Symposium on Industrial Electronics, ISIE 1997, vol. 2, pp. 677–682. IEEE (1997)
9. Zuo, G., Zhang, P., Qiao, J.: Path planning algorithm based on sub-region for agricultural robot. In: 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR), vol. 2, pp. 197–200. IEEE (2010)

10. Valente, J., Sanz, D., Del Cerro, J., Barrientos, A., de Frutos, M.Á.: Near-optimal coverage trajectories for image mosaicing using a mini quad-rotor over irregular-shaped fields. *Precision Agric.* **14**(1), 115–132 (2013)
11. Di Franco, C., Buttazzo, G.: Energy-aware coverage path planning of UAVs. In: 2015 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), pp. 111–117. IEEE (2015)
12. Bouktir, Y., Haddad, M., Chettibi, T.: Trajectory planning for a quadrotor helicopter. In: 2008 16th Mediterranean Conference on Control and Automation, pp. 1258–1263. IEEE (2008)
13. Mueller, M.W., Hehn, M., D’Andrea, R.: A computationally efficient algorithm for state-to-state quadrocopter trajectory generation and feasibility verification. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3480–3486. IEEE (2013)
14. Meier, L., Honegger, D., Pollefeys, M.: PX4: a node-based multithreaded open source robotics framework for deeply embedded platforms. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 6235–6240. IEEE (2015)