Available online at www.sciencedirect.com
ScienceDirect
Progress in Natural Science 19 (2009) 1357–1363
Progress in Natural Science
www.elsevier.com/locate/pnsc

Mission planning of the flying robot for powerline inspection

Chang'an Liu *, Lei Wang, Chunyang Liu

School of Computer Science and Technology, North China Electric Power University, Beijing 102206, China

Received 9 January 2009; received in revised form 21 February 2009; accepted 22 February 2009

Abstract

Mission planning is an important step in the powerline inspection flying robots. To deal with the tasks of powerline flying robots, a flying robot mission-planning system is established based on the unmanned mini-helicopter model described in this paper. It can determine the best checking order, the optimal space path and the best flight trajectory. The objective function and the constraints are described by establishing the mathematical model, and the mission-planning system is decomposed by using the tiered methods. Multi-agent technology is chosen to solve the mission-planning issue. Finally, simulation is conducted to verify that the mission-planning system is feasible and efficient.

Keywords: Powerline inspection flying robot; Mission planning; Multi-agent technology

1. Introduction

Powerline inspection is the basic step to ensure the minimum fault rate. Using such methods as manual inspection and record has some flaws, such as it cannot ensure the inspectors go to the right position and the description of the fault could not be precise or standard. Using the manned helicopter to carry out the inspection, the staff should have the technology and experience of the helicopter and wear particular uniform, so it needs a great deal of manpower and material resources [1]. Therefore, the research on the flying robot for overhead powerline inspection (FROPI) causes growing concern [2]. FROPI is a robot that considers the unmanned mini-helicopter as the carrier and can autonomously fly along the powerline, using its own portable image acquisition and thermal imaging equipment to inspect the powerline [3].

The purpose of mission planning of FROPI is to analyse the geographic information and lines information, to consider certain limits and constraints of the flying robot itself and to determine the best checking order, the optimal flight path space and the best flight trajectory for the robot to implement the inspection on the powerline [4,5]. Thus, it can realize intelligent flight of the FROPI, so as to efficiently complete the task of checking the powerlines. There has been few published works about it till now.

In order to realize the mission planning of FROPI, the following key issues need to be solved: first, human–computer exchange. To realize the real-time unmanned control to the inspection robot, the robot can conduct real-time information exchange with a ground base station [6,7]. The second issue is the path choice. The flying robot may need time to adjust during the processes of the mission, so it is necessary to consider the issue of minimizing the path and also the issue of selecting the intermittent point in the path in the planning tasks. The third issue is real-time flight control. The robot must in real-time adjust its pose in the flight, so that the robot can fly along the preplanned flight routes as short as possible [8,9].

This paper presents a method for mission planning of FROPI so that the flying robot mission-planning problem

* Corresponding author. Tel.: +86 10 51963387; fax: +86 10 80798575.
E-mail address: liuchangan@ncepu.edu.cn (C. Liu).

1002-0071/$ - see front matter
doi:10.1016/j.pnsc.2009.02.006

is solved by establishing mathematical models and using the multi-agent technology.

2. Mathematical description

2.1. The objective function and planning variables

It is assumed that there are n sections between the start point and the target point in the flight, recorded as $\{b_i\}$, $i = 1, \ldots, n$. In each section of the flight, the robot should always keep real-time communication with the ground control system, maintain good postures, avoid the dangerous zones, get to the intermittent points and complete the inspection mission successfully. The objective function of the flying robot mission planning is:

$$Z = mN + \sum_{i=1}^{n}(pL_i + qA_i + wS_i) + QT \quad (1)$$

where $L = L_f - L_t$, $A = [\Delta\alpha, \Delta\beta, \ \Delta\gamma]$, $i = 1, \ldots, n$, and α, β, γ are the angles (in degrees) in the level and the vertical planes. The basic variables are N, L, A, S and T. N is the reception and recognition of mission objectives; L_f is the distance that the current flying robot can fly; L_t is the distance between the robot and the nearest intermittent point; and L is the difference between L_f and L_t.

This paper assumes that $L_i > b$ (b is a constant greater than zero) to determine the selection of intermittent points.

A is the difference between the actual flight angle degree of the robot and the angle degree that was planned before, and the value of A_i within a range is used to control the shortest flight distance of the robot.

S is the distance from the robot to the dangerous regions or dangerous points. This paper assumes that $S_i > R$ (R is the radius of the dangerous area that the robot can fly through safely) to ensure the safety of the flying robot.

T is the postures that the flying robot should keep when hovering space. The postures should be changed according to the environments. In formula (1), m, p, q, w *and* Q represent weights.

2.2. Constraints conditions

To complete a specific inspection task successfully, the planned path should pass through the dangerous area as soon as possible and not exceed the physical limits of the flying robot. Fuel consumption is not over the limits and satisfies other relevant parameters [10,11] that the specific mission determined. In general, the path generated from the mission planning should satisfy the following constraints:

(1) *Constraints of the longest flight distance.* This limits the path of each section and the total length must be less than or equal to a pre-set maximum distance. It depends on the fuel the flying robot carried and the allowed flight time for reaching the goal. It is assumed that L is the maximum length of the path and l_i is the length of each section, and then the constraints can be expressed as:

$$l_i \le L_i, \quad \sum_{i}^{n} l_i \le L \quad (2)$$

(2) *Obstacles constraints.* Let S_i be the dangerous index of the path of section i, which is used to choose the least dangerous path, calculated as follows:

$$S_i = \sum_{j=1}^{N_S} K_j/(R_{Sj})^3 \quad (3)$$

where K_j is the dangerous strength of the section j. R_{Sj} is the distance between the flying robot and the dangerous source. N_S is the number of dangers found in the path i.

(3) *The maximum climb and fall angle.* This angle is determined by the flying robot's own mobility, which limits the maximum angle [12] of the rise and fall of the path in the vertical plane. It is assumed that γ is the allowable climbing/fall angle, and the constraints can be expressed as:

$$\frac{|z_j - z_{j-1}|}{|x_i|} \le \tan\gamma \quad (4)$$

where z_j is the coordinate component on the vertical plane of the path point j and x_i is the coordinate component on the level of the surface.

(4) *The minimum path length.* When the flying robot implements its mission, the robot should fly along the shortest flight path under the condition of keeping a safe distance from the powerline. The distances from the powerline and the angle to the powerline are controlled to ensure the shortest path. It is assumed that the offline distance is D, the farthest distance is s and the shortest distance is f. The path deviation angle is A_i. The constraints can be expressed as:

$$s \le D \le f, \quad 0 \le A_i \le \alpha. \quad (5)$$

where α is a vector greater than zero.

(5) *Flight speed.* The camera equipment needs sufficient processing time when the flying robot executes the inspection mission. Flight speed should be fast and meet the performance parameters [13] that airborne equipment requires.

(6) *Hovering space.* In the implementation of its mission, if there are suspicious or damaged lines, the flying robot should act accordingly. At this time, the flying robot should be in a safe position and posture to ensure that the appropriate task is completed successfully.

3. Tiered approach of planning

Tiered planning uses the simplified model to abstract the planning. It compresses the details and focuses on the vital issues domain. In this paper, the tiered planning is used to simplify the solving process of the mission planning. The advantage is that some of the options are excluded from consideration at a higher level of abstraction, and these options do not need to be considered at a lower level of abstraction, so that the planning system will not solve the issues that have nothing to do with it [14]. This paper divides the planning system into three levels: the mission-planning level, the path-planning level and the trajectory control level. Each level determines its own planning variables and the objective function. They are also the planning constraints for the next level.

(1) *Mission-planning level.* This level conducts long-term planning such as determining intermittent points and avoiding the dangerous signs according to the line information, topographic information, the danger constraints, etc. The planning variables include take-off points, goal, grade goal, take-off time and the time taken to reach the goal. Its objective function is as follows:

$$H_1 = \sum_{i=1}^{n}(d_1 G + d_2 Y^2) \tag{6}$$

where G expresses that the robot maintains the distance without interference between the powerline and the robot as far as possible and Y expresses the distance that should be increased between the flight path and the dangerous region as far as possible. The weights of G and Y are represented by d_1 and d_2, respectively. The constraints of this level are to choose the minimum path length to safely reach the goal. This level sends information such as planned take-off points, goals, intermittent points, take-off time and road code to the path-planning level.

(2) *Path-planning level.* This level conducts the behavioural planning, and its purpose is to choose a flight path where the robot can complete the scheduled task safely according to the mission requirements, constraints on fuel, weather and other information, and the planning variable includes the information of intermittent points. Its objective function is as follows:

$$H_2 = \sum_{i=1}^{n}(r_1 C_i^2 + r_2 U^2) + hJ \tag{7}$$

where C is the difference between the fuel consumption in the course of the robot flying along the actual flight path and the fuel limits of the robot, U is the distance between the flight path and the abnormal environment, J is the information on intermittent points, and r_1, r_2, and h are, respectively, the weights of the three items.

The corresponding path lengths of each path sequence are outputted to the mission planning. The flight path points are outputted to the trajectory control level to smooth, so that the path can be flown [15].

(3) *Trajectory control level.* The flight path point sequence is smoothed according to the robot's flight characteristics, the traffic situation around the powerlines and the temporary risk situation. It determines the control variable [16]. Its objective function is as follows:

$$H_3 = \sum_{i=1}^{n}(n_1 h_i + n_2 \partial_i + n_3 \beta_i + n_4 \gamma_i) + QT \tag{8}$$

where h expresses that the flying robot should be at the same degree of level with the powerlines. The second, third and fourth items express that the posture should be adjusted to real-time in the flight. ∂, β, and γ are the control parameters of the flying robot in the flight. The fifth item expresses the position that the flying robot should maintain when hovering space [17]. The weights of the items are represented as n_1, n_2, n_3, n_4, and Q.

The information of planned flight, path altitude, speed and route in this level will be sent to the flying robot's executive part to carry out.

4. Multi-agent technology solves the mission planning

Agent is the entity with high self-control ability running in a dynamic environment, and its fundamental purpose is to accept the commission of another entity and to provide help and services for that [18]. Multi-agent system (MAS) theory is the new way to design and realize complex software systems and control systems. Generally, the collaborative capacity of MAS for solving the issue is better than a single agent. MAS has characteristics such as the interoperability between itself and existing systems or software, solving the data and control with distribution character, so it can improve efficiency and robustness. As a result, it is used to solve this issue, collaboration and consultations that a single agent cannot deal with. In general, MAS has four characteristics: (1) adequate capacity or information is available to solve the issue; (2) the overall control agencies do not exist in the entire system; (3) the data are distributed; (4) calculations are not necessarily synchronous. From the flying robot mission-planning model established above, it can be seen that it is a feasible way to use the MAS to solve the mission-planning issue.

4.1. The components of the multi-agent system

The flying robot inspection mission-planning system is a functional system composed of the management agent (MA), the coordination agent (CA) and the resources agent (SA) based on a specific communication and coordination mechanism, as shown in Fig. 1. The sharable content of the blackboard is the structured data that puts the targeted as a unit. The number of the local blackboard depends on the number of system function modules, the amount of computation and mutual connection. According to the function and the scale of the mission-planning system, it needs three to four local blackboards corresponding to CPU and configures local monitors. The entire functional module (knowledge source KS) is also divided into three or four groups, and each group corresponds to a partial blackboard [19]. It uses single-entity cooperation among the single agents and uses the federal collaborative approach among the module agents. The agent has its own resources

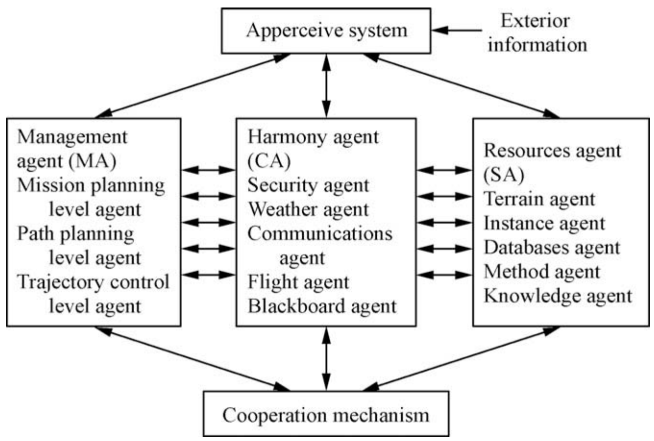

Fig. 1. The multi-agent mission system composition of the inspection flying robot.

and task-sharing capability, so that the entire multi-agent system has a strong ability to adapt to the environment.

4.2. The functions of the different parts in the multi-agent system

In the flying robot mission-planning system, MA includes management agents of the mission-planning level, the path-planning level and the trajectory control level.

(1) *The management agents of the mission-planning level.* Aiming at a planning mission, it decomposes the task in accordance with the minimum cost principle of the objective function as Eq. (1) and the planning constraints conditions. Then, the decomposed sub-tasks queue is released to the CA such as the security agent and the flight agent and the SA such as the databases agent, the communications agent and the terrain agent through the blackboard agent, so as to urge MA, CA and SA in accordance with their capacity to compose the coordination mechanism implementation sub-tasks queue. MA sends the take-off point, the target point, the main route point and route codes and other information to the agent of the path-planning level through the blackboard agent after colligating the solution results.

(2) *The management agent of the path-planning level.* The path-planning management agent receives the long-term planning route information produced by the mission-planning level from the blackboard agent. It decomposes the task in accordance with the minimum cost principle of the objective function as Eq. (1) and the planning constraints conditions. Then, the decomposed sub-tasks queue is released to the CA such as the weather agent and flight agent and the SA such as the databases agent, the communications agent and the terrain agent through the blackboard agent, so as to urge MA, CA and SA in accordance with their capacity to compose the coordination mechanism implementation sub-tasks queue. The management agents of the path-planning level send more detailed flight path route information to the trajectory control agent after colligating the solution results.

(3) *The management agents of the trajectory control level.* The management agent of the trajectory control level decomposes the task in accordance with the minimum cost principle of the objective function as Eq. (1) and the planning constraints conditions after obtaining the more detailed path route information. Then, the decomposed sub-tasks queue are released to the CA and SA, such as the security agent and the flight agent through the blackboard agent, so as to urge MA, CA and SA in accordance with their capacity to compose the coordination mechanism implementation sub-tasks queue. Afterward, the management agent of the trajectory control level will distribute the detailed route information to the flight agent and the security agent through the blackboard agent. They, respectively, solve the sub-task in accordance with planning models and restrictive conditions. The flight agent smooths the flight path in accordance with the flight characteristics of the flying robot. The security agent optimizes the flight path in accordance with the air traffic situation and the temporary danger. Finally, the management agents of the trajectory control level colligate the solved results of the sub-task and send the detailed flight path information (altitude, speed and direction) to the automatic driving system of the flying robot to implement.

CA is an agent that completes the part functions of the system. According to its objectives and requirements, CA carries out the corresponding parts of the work through acquiring the necessary resources, including the establishment of a viable plan and the harmony of processing the related resources. The security agent of the CA optimizes the pre-flight path in accordance with the situation of dangerous signs, temporary risks and the situations of the flight path. In the process of inspection, the flight agent optimizes the path again in accordance with the fuel constraints, range bound, the largest yaw angle, the biggest pitch angle, hovering in the air and mobile performance of the flying robot to ensure that the planned path could be flown.

SA is an agent that manages the resources of the system including the library of knowledge, the models, the methods, databases, the terrains, the goals, and the situation. The performance data of the flying robot, planning constraints and the necessary data in the process of the inspection are stored in the database, and the dangerous data are stored in the situation library. When the mission-planning system is run, an SA could be called by another agent in the scheduling of the MA. SA has its own algorithm (for orders, some of choice, collaboration, etc.) and knowledge (for instructions, such as sub-handling similar situations) [16]. When collaborating with other agents, SA supports their decision-making.

The collaboration mechanisms among the agents: agents in the system contain free and controlled collaboration and implementation mechanisms. There exists horizontal and vertical collaboration and communication means: agents between the lower and upper level progress the collaboration, while those at the same level also need mutual cooperation.

5. Solution and simulation

In order to verify the validity and feasibility of this mission-planning system, this paper proceeds with the solution and tasks simulation for the flying robot on the implementation of the checking tasks. This paper sets up the input and output matrix for mission planning in accordance with the composed, stratified situation and the interaction between the sub-tasks of the MAS in the flying robot mission-planning system for powerline inspection.

The set A against sub-tasks r is composed of CA and SA with indexes m and n: $\{CA_1, CA_2, \ldots, CA_m, SA_1, SA_2, \ldots, SA_n\}$. Suppose that the consumption of the CA_i or SA_j implementation of tasks r is Cr ($Cr > 0$) and that non-participation in the implementation of the task r is zero. The output Re ($\mathrm{Re} \geqslant 1$) is the returned numbers of data identifier after the CA_i or SA_j implementing the task r. Integrating all the situations of CA or SA of A implementing the sub-task r, we can obtain the Input Vector and Output Vector.

$$\text{Input Vector}: C(r, A) = [Cr_1 Cr_2 \ldots Cr_3]. \tag{9}$$

$$\text{Output Vector}: \mathrm{Re}(r, A) = [\mathrm{Re}_1 \mathrm{Re}_2 \ldots \mathrm{Re}_n] \tag{10}$$

Vectors will be extended to the tasks $R_i(t) = \{r_1, r_2, \ldots, r_k\}$ composed of the sub-task queue form. Then, the input matrix and output matrix can be calculated using:

$$\text{Input matrix}: C(R_i(t), A) = [C(r_1, A)\mathrm{C}(r_2, A) \ldots C(r_k, A)] \tag{11}$$

$$\text{Output matrix}: \mathrm{Re}(R_i(t), A) = [\mathrm{Re}(r_1, A)\mathrm{Re}(r_2, A) \ldots \mathrm{Re}(r_k, A)] \tag{12}$$

When the sub-tasks queue is published in A, each CA or SA calculates the consumption and submits it to the MA aiming at each sub-task. MA inputs the consumptions obtained from CA or SA into the input matrix $C(R_i(t), A)$ and derives the coordination mechanism A according to the principle of non-zero consumption. Moreover, when the coordination mechanism implementing the tasks is over and returns the data identifiers, MA counts the number of data identifiers in each of the CA or SA to form the output matrix $\mathrm{Re}(R(t), A)$ so as to help evaluate the effect of implementing the coordination mechanism [17].

This paper gives an example aiming at the composition of the coordination mechanism and the access efficiency of the data identifier. For instance, the task of the planning level, the path-planning level and the trajectory control level are implemented in order and they are controlled and managed by the MA of the mission-planning level, path-planning level and the trajectory control level as three separate tasks. This paper puts the current target point and the next target of a path in an executable cycle.

In the implementation of a cycle, MA selects the corresponding CA and SA composes the coordination mechanism according to the objective function and the constraints of the task r. MA requests the random data within CA and SA memory and forms coordination mechanisms and then returns the data identifier matched with these data. At the same time, it records the composition and execution time of each implementation cycle.

Fig. 2 is the set A of the CA and SA in the implementation of a cycle, $A = \{CA_1, CA_2, CA_3, CA_4, CA_5, SA_1, SA_2, SA_3\}$. It obtains the input matrix $C(R_i(t), A)$ aiming at the sub-task $R_i = \{r_1, r_2, r_3\}$ and obtains A_i accordingly: $A_i = \{CA_1, CA_2, CA_3, CA_4, CA_5, SA_1, SA_2, SA_3\}$.

Fig. 2 is the implementation of the input matrix $C(R_i(t), A)$. Fig. 3 is the implementation of the output matrix $\mathrm{Re}(R_i(t), A)$.

In Figs. 2 and 3, CA_1–CA_5 represent the communications agents, SA_1, SA_2 and SA_3 represent the data agent, the goals agent and the resources agent, respectively.

According to the method, this paper simulates the established system in the two-dimensional environment with the Matlab6.5 simulation platform.

The experiment configurations are as follows: CPU: AMD Turion X2 1.6 GHz. Memory: 1.00 GB the physical address of the memory expansion. The simulation tool: Matlab6.5 English.

The courses of the experiment are as follows:

(1) First, it is assumed that the agents described in the text can run well and the communications coordination mechanisms among the entire agents work properly.
(2) This paper proposes a new path-planning method aiming at the characteristics of the flying robot mission for powerline inspection. When the flying robot did not arrive at the powerline facilities, it will fly along the shortest and safest flight path [20,21]. When the powerline flying robot is in the implementation of its tasks, it remains at a safe distance between the powerlines and itself so as to ensure that it can successfully complete the inspection tasks. When the flying robot moves close to dangerous zones, it can identify and avoid the zone in time.
(3) The flying robot can identify target points (including intermittent) and can accurately reach them.

Based on the above methods, this study simulated the task of the flying robot. The corresponding two-dimen-

$$\begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \end{array} \begin{array}{c} CA_1, CA_2, CA_3, CA_4, CA_5, SA_1, SA_2, SA_3 \\ \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

Fig. 2. Input matrix $C(R_i(t), A)$.

$$
\begin{array}{c|cccccccc}
 & CA_1 & CA_2 & CA_3 & CA_4 & CA_5 & SA_1 & SA_2 & SA_3 \\
r_1 & 8 & 0 & 8 & 1 & 1 & 1 & 1 & 1 \\
r_2 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
r_3 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1
\end{array}
$$

Fig. 3. Output matrix $\mathrm{Re}(R_i(t), A)$.

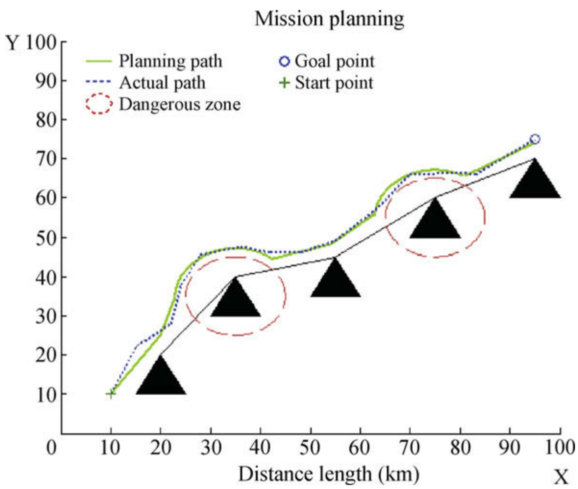

Fig. 4. Two-dimensional simulation of the mission planning.

Table 1
The main referenced data of the experiment.

Obstacles	Adjust times	Adjust angles (°)	If overstep the limits	If pass safely
Before no. 1	2	68–34, 36–65	No	Yes
No. 1	3	65–54, 53–26, 27 to −13	No	Yes
Before no. 2	3	−12 to −8, −6 to 11, 12 to −38	No	Yes
No. 2	4	39–49, 50–61, 60–14, 15–18	No	Yes
After no. 2	2	16 to −4, −6 to 29	No	Yes

sional simulation is shown in Fig. 4, and the experimental data are shown in Table 1.

It can be seen that the flying robot is able to independently recognize the difference between its own state and the advance planning state and make effective adjustments in a timely manner and also in real-time to identify dangerous areas and avoid them safely (Fig. 4). Table 1 shows that the robot can adjust itself in a timely manner and accurately so as to go through the dangerous zone safely and successfully complete the inspection tasks on the premise of not exceeding the performance of its own. Throughout the process, the flying robot maintains a shorter flight path and reaches the target point successfully. Through the data and results mentioned above, it can be seen that the mission-planning system and its solution methods are effective.

6. Conclusions

The mission-planning system of FROPI is used for the powerline inspection flying robot to identify the best detection sequence, the optimal flight path space and the best flight path. In this paper, the mission-planning system is a functional one composed of the management agents (MA), coordination agents (CA) and resources agents (SA) that are based on a specific communication and coordination mechanism. The solution of the powerline inspection flying robot mission-planning system based on multi-agent technology simplifies the complex issues to a simpler single distributed issue through stratifying and decomposing them. The experiments show that using multi-agent interaction, it can reduce the system's total consumption, improve the efficiency of data acquisition and processing, strengthen the system adaptive capacity and make the system more efficient. In the future, the collaboration between the agents, more stringent operating standards and the overall system performance will be further studied and the resources will be improved to apply the actual tasks of mission planning.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 60775058) and the Key Project of the China Ministry of Education (Grant No. 107028).

References

[1] Zhao P, Deng C, Yuan Y. Patrol of transmission line with helicopter. North China Electric Power 2002;27(10):1–3 (in Chinese).
[2] Jones DI, Earp GK. Camera sightline pointing requirements for aerial inspection of overhead powerlines. Electric Power Syst Res 2001;57(2):73–82.
[3] Whitworth CC, Duller AWG, Jones DI, et al. Aerial video inspection of overhead powerlines. Power Eng J 2001;15(1):25–32.
[4] Barrientos A, Campoy P. An autonomous helicopter guided by computer vision for inspection of overhead power cable. In: Proceedings of IROS conference on aerial robotics. Lausanne, Switzerland; 2002. p. 1026–30.
[5] Peungsungwal S, Pungsiri B, Chamnongthai K, et al. Autonomous robot for a power transmission line inspection. In: Proceedings of the IEEE international symposium on circuits and system. Sydney, Australia; 2001. p. 121–4.
[6] Campoy P, Barrientos A. An autonomous helicopter guide by computer vision for visual inspection of overhead power cable. In: Proceedings of the fifth international conference on live maintenance. Madrid, Spain; 2000. p. 344–9.
[7] Hartmut H. Remote controlled flying robot platform. In: Proceedings of third international conference on digital information management. London, UK; 2008. p. 920–1.
[8] Shi H, Zhao Z. Mission-planning system of low observation aircraft based on multi-agent technique. J PLA Univ Sci Technol 2005;6(4):337–41 (in Chinese).
[9] Rejon V, Aranda-Bricaire E. Discrete-time stabilization of a remotely controlled flying robot in real-time without velocities measurement. In: Proceedings of 33rd annual conference of the IEEE on IECON. Taipei, Taiwan; 2007. p. 756–61.
[10] Oliver T, Hans PG. Two-degree-of-freedom robust controller for an autonomous helicopter. In: Proceedings of the American control conference. Denver, United States; 2003. p. 993–8.
[11] Stefan H, Gauray S. Omnidirectional vision for an autonomous helicopter. In: Proceedings of IEEE international conference on robotics and automation. Taipei, Taiwan; 2003. p. 558–63.

[12] Golightly I, Jones D. Corner detection and matching for visual tracking during powerline inspection. Image Vis Comput 2003;21(9):827–40.
[13] Jones DI. Aerial inspection of overhead powerlines using video: estimation of image blurring due to vehicle and camera motion. IEE Process Vis Image Signal Process 2000;147(2):157–66.
[14] Williams M, Jones DI, Earp GK. Obstacle avoidance during aerial inspection of powerlines. Aircr Eng Aerosp Technol 2001;73(5):472–9.
[15] Zhou W, Yuan J, Luo J. Flying robots fast circumnavigation trajectory design and guidance. In: Proceedings of IEEE international conference on robotics and biomimetics. Kunming, China; 2006. p. 263–7.
[16] Luo P, Pei H. An autonomous helicopter with vision based navigation. In: Proceedings of IEEE conference on control and automation. Guangzhou, China; 2007. p. 2595–9.
[17] Joerq SD, Eric NJ. Multi-sensor navigation system for an autonomous helicopter. In: Proceedings of AIAA/IEEE digital avionics systems. CA, United States; 2002, 8C11-9.
[18] Zhang K, Shao C, Qiang W. A study of mission planning systems based on agent-oriented technique. High Technol Lett 2002;12(5):82–6 (in Chinese).
[19] Shen LC, Gao GH, Chang WS. An open system approach to mission route planning. J Astronaut 1998;19(2):13–8 (in Chinese).
[20] Koenig S, Likhachev M. Improved fast replanning for robot navigation in unknown terrain. In: Proceedings of IEEE international conference on robotics and automation. Washington, DC, United States; 2002. p. 968–75.
[21] Peng C, LaValle SM. Resolution complete rapidly-exploring random trees. In: Proceedings of IEEE international conference on robotics and automation. Washington, DC, United States; 2002. p. 267–72.