Vision-based map building and path planning method in unmanned air/ground vehicle cooperative systems

Xi a’Xing¹, Zhao Jin¹,², Fu Haolong¹, Zhou Tao¹, Liu Dongjie¹
¹Department of Mechanical Engineering, Guizhou University, Guiyang 550025, People’s Republic of China
²E-mail: zhaoj@gzu.edu.cn

Abstract: To reduce the casualties and improve the efficiency of robot, unmanned air/ground vehicle (UAV/UGV) cooperative systems are proposed. In this study, mathematical algorithms are applied in a typical-rescue scenario. As a ‘flying eye’, the UAV provided the global information by vision sensor. Then, the images were processed with SURF algorithm and image binarisation to model the environment. Based on the UAV’s information, the optimised A* algorithm was proposed. Finally, a feasible path was designed for UGV to rescue. Experiments are performed to evaluate the performance of the proposed method. Results show that SURF algorithm and image binarisation can realise the accuracy and robustness of map building. The optimised A* algorithm can provide a real-time and feasible path.

1 Introduction

At present, unmanned air/ground vehicle (UAV/UGV) cooperative systems can replace human beings to approach these dangerous missions, such as fire rescuing and military [1, 2]. A typical scenario of UAV/UGV cooperative systems is shown in Fig. 1, where the target needs to be rescued. From Fig. 2, we can see that the UGV is easily obscured by obstacles and cannot get the information of the target. By contrast, the UAV can easily recognise the target, but the UAV is limited by flight time. Therefore, the UAV and the UGV should work cooperatively.

Visual sensor is one of the main sensors to solve the problem. Chaimowicz et al. [3, 4] had carried out a series of studies based on UAV/UGV cooperative system and studied the visual target detection and localisation algorithm. The verification experiment had been carried out successfully. However, they did not consider information exchange between UAVs and UGVs.

Li et al. [5] studied the ground map building and put forward a path planning in UAV/UGV cooperative systems. They processed image with image denoising, image correction, and obstacle recognition to construct the ground map automatically. Nevertheless, the results did not convert the map in pixels to a metric map. Feng et al. [6] explored the method of using robot to achieve cooperative navigation. The recognition of ground obstacles was realised by using UAV vision sensor. Based on the YCrCb colour space, the moving target was detected and the tracking of the ground robot was realised by using the Windows Tracking. However, this method was verified only in a special environment.

In the study of path planning, Aghaeeyan et al. [7] put forward the system of leader and follower in UAV/UGV cooperative systems and designed a UGV sliding mode controller. However, the path planning of UGV is planned under the local strategy and the accuracy cannot achieve the desired result in the process of target following. Ganeshmurthy et al. proposed a heuristic-based method to search the feasible initial path efficiently to solve the problem of dynamic environments [8]. Zhu et al. proposed a global path planning of robots by multi-objective algorithms [9, 10]. Zhao et al. [11] proposed modified artificial potential field in a complex environment and used the adjustment factor to improve the objective function of the gravitational point. Finally, they realised the path planning of UAV. However, the artificial potential field method has different objective functions for different obstacles, resulting in large amount of computations.

In this paper, the main contributions as follows: (i) With a vision sensor, images are acquired from outside environment by UAV. Then, the target and obstacles are found and a traversable map is built. (ii) Three path planning algorithms are compared. Then, the optimised A* algorithm is proposed, which can plan a feasible path. (iii) Two groups of experiments are designed to evaluate the performance of the proposed algorithms.

Fig. 1 Typical UAV/UGV cooperative systems

Fig. 2 Information from different angle of view (a) UGV angle of view, (b) UAV angle of view
2 System overview

Based on the automated highway system [12], a hierarchical system is proposed in Fig. 3. The management layer is mainly responsible for strategy and supervise the UAV and UGV. The communication layer is mainly responsible for information exchange among inter-UAVs, inter-UGVs, intra-UAV/UGV. The function of execution and control layer is to sense the environment and complete the mission by controlling the UAV/UGV.

To more clearly illustrate the proposed structure framework, a simplified-rescuing scenario is shown in Fig. 4: The UAV is mainly equipped with a visual sensor, which acquires environmental information. The UGV is mainly equipped with data transmission and performs a rescue mission. The start point is the location of UGV, the end point is the location of rescuer. Others are treated as obstacles.

3 Mapping with vision sensor

Map-building method includes target searching and obstacle recognition.

3.1 Target searching in a simplified-rescuing scenario

The method of target searching is to find an optimal transformation by comparing the histogram, feature, texture, structure, greyscale etc. [13]. The general process of target searching is shown in Fig. 5.

3.2 Image greyscale

The image is acquired, when the UAV hovering on the target, the image is a RGB model. In order to adapt human sensitivity, it is necessary to quantify the colour value of image [14]. Greyscale is a weighted average method. \[ R = G = B = (0.3R + 0.59G + 0.11B)/3 \] (1)

3.3 SURF algorithm

As the SURF algorithm has strong robustness and scale invariance in image features extracting [15]. The general idea of SURF algorithm is shown as follows:

i. Integral image. The integral image value of \( i(i', j') \) is shown as follows:

\[ i(i, j) = \sum_{i' \leq i, j' \leq j} p(i', j') \] (2)

where \( p(i', j') \) is the greyscale value.

ii. Determinant of Hessian approximation. The size of template is \( 9 \times 9 \) and \( \sigma = 1.2 \) are used to filter and detect. \( D_{xx}, D_{yy}, D_{xy} \) are the convolution results of template and image. Thus, the Hessian matrix \( H \) is given as

\[ \text{Det}(H_{\text{approx}}) = D_{xx}D_{yy} - (0.9D_{xx})^2 \] (3)

iii. Scale space representation. The box filter and integral image are adopted. Indirect method of increasing the size of box filter template is applied to filter.

iv. SURF algorithm feature descriptors. Taking the feature point as the centre, the \( 20 \times 20 \) image is divided into \( 4 \times 4 \) pixels with a \( 5 \times 5 \) rectangle region. \( \sum d_x, \sum d_y, \sum |d_x|, \sum |d_y| \) are the Haar wavelet response values of each rectangle in each region.

v. Matching the target. In order to achieve rapid matching, the Laplace response sign of feature points is added. Only the same response point can match successfully.

By the above-mentioned steps, the result of target searching is shown in Fig. 6.

3.4 Obstacle recognition by image segmentation

Image segmentation is to divide the image into several regions with unique properties and find interesting regions [16].
i. Spatial transformation. The environmental information collected by UAV is easily affected by light. Therefore, the hue, saturation, value (HSV) spatial model is applied.

ii. Image binarisation. In order to show the image information of obstacles, the image binarisation is used to mark obstacle \[17\]. The image binarisation is shown as follows:

\[
f(x, y) = \begin{cases} 
1 & f(x, y) \in T \\
0 & \text{else} 
\end{cases}
\]

where \(f(x, y)\) is the image grey value. \(T\) is the grey value threshold. The result in HSV space is shown in Fig. 7.

According to the SURF algorithm and image segmentation, the image information has transformed into executable map through a coordinate transformation. The map is shown in Fig. 8.

4 Coordinate transformation

In Section 3, the pixelated map has to be gained. In order to realise the global path planning, the map in pixels should be converted to a metric map. UAV and UAV coordinate systems are transformed into the same world coordinate system \[18\]. The main coordinate systems are shown in Fig. 9: \(O_{uv}\) is the pixel coordinate system, \(O_{X,Y,Z_C}\) is the image coordinate system, \(O_{X,Y,Z_G}\) is the camera coordinate system, \(O_{X,Y,Z_U}\) is the UGV coordinate system, and \(O_{X,Y,Z_W}\) is the world coordinate system.

4.1 UAV's coordinate transformation

To avoid errors caused by camera motion, the camera coordinate system has simplified to the UAV coordinate system and the UAV is taken as a rigid motion model. The Homogeneous transformation is given as

\[
Z_{P_{w}} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overrightarrow{K}(R_{w}P_{w} + t_{w}) = \overrightarrow{K}P_{w}^{*}
\]

where \(\overrightarrow{K}\) is the Camera Intrinsics. Rotation matrix \(R_{t}\) and translation vector \(t_{w}\) are the exterior of camera.

4.2 UGV's coordinate system transformation

The Homogeneous transformation of the UGV is shown as follows:
\[ \begin{bmatrix} x_G \\ y_G \\ 0 \end{bmatrix} = R_2 \begin{bmatrix} x_w \\ y_w \\ 0 \end{bmatrix} + t_2 \]  

(6)

where \( R_2 \) is the rotation matrix and \( t_2 \) is the translation vector.

### 5 Global path planning

In Section 4, the traversable map is gained. In this section, the global path planning of UGV is discussed. First, the traditional Bidirectional Rapidly exploring Random Trees and the \( A^* \) algorithm are introduced and analysed. Then, the optimised \( A^* \) algorithm is proposed.

#### 5.1 Bidirectional rapidly exploring random trees

Bidirectional rapidly exploring random trees (BRRT) is an improved algorithm based on rapidly-exploring random trees (RRT) [19]. It is aim to find a feasible path from the start point \( q_{init} \) to the goal \( q_{goal} \). The basic structure of the RRT is shown in Fig. 10.

#### 5.2 \( A^* \) algorithm

\( A^* \) algorithm is a heuristic algorithm with optimality and high efficiency. \( A^* \) balances the distance between the start and the goal [20]. The function of the \( A^* \) algorithm is shown as follows:

\[ f(n) = g(n) + h(n) \]  

(7)

where \( f(n) \) is the total movement cost function, \( g(n) \) is the exact cost of the path from the starting point to the node \( n \), and \( h(n) \) is the heuristic estimated cost from the node \( n \) to the goal.

#### 5.3 Optimised \( A^* \) algorithm

For the UGV to track the feasible path, the main step is to ensure the path smoother. The optimised algorithm \( A^* \) algorithm is to eliminate the influence of the vertices [21]. The basic function of the optimised algorithm \( A^* \) algorithm is shown as

\[ Z = \alpha (p_i' - p_{i-1})^2 + \beta ((p_i' - p_i)^2 + (p_i' - p_{i+1})^2) \]  

(8)

where \( p_i \) is the coordinate position before optimisation, \( p_i' \) is the coordinate position after optimisation. \( \alpha \) and \( \beta \) are constants and \( Z \) is the minimum distance.

For \( \alpha \), the following iteration is used:

\[ p_{i+1}' = p_i' - \alpha (p_i' - p_i) \]  

(9)

For \( \beta \), the following iteration is used:

\[ p_{i+1}' = p_i' - \beta ((p_i' - p_{i-1})^2 + (p_i' - p_{i+1})^2) \]  

(10)

Considering the maximum curvature limit of the UGV, \( \alpha = 0.5 \) and \( \beta = 0.2 \) are chosen to optimised \( A^* \) algorithm.

### 6 Simulation results

#### 6.1 Simulation in a simple experiment

In order to verify the superiority of the optimised \( A^* \) algorithm, a simple experiment is obtained. Simulation results are shown in Fig. 11.

In this experiment, the real time and feasibility of the three path-planning algorithms mentioned in Section 5 are tested. Twenty experimental results are compared and analysed with the same starting point (50, 650) and the same ending point (650, 50) in 700 \( \times \) 700 pixels experiment. Simulation results are shown in Fig. 12 and Table 1.

Table 1 shows the average results of 20 groups. From the results, we can see that the average shortest path length is the optimised \( A^* \) algorithm, the average shortest computing time is \( A^* \)
algorithm. Although the optimised $A^*$ algorithm adds a little calculating time compared to the $A^*$ algorithm, most important is that the optimised $A^*$ algorithm reduces the number of vertices to 0. Therefore, the optimised $A^*$ algorithm has a stable performance in shortening length and smoothing path. It is more suitable to design a real time and feasible path for $UGV$ to track in a real environment.

6.2 Simulation in a complicated experiment

In order to further verify and apply the proposed algorithms. More complicated conditions are set. In the complicated experiment, the information of environmental is obtained by $UAV$ in real environment. Fig. 8 has showed the result of mapping. Figs. 13 and 14 show the trajectory in a complex environment.

Fig. 13 Result of the trajectory in the complex environment

Fig. 14 Partial trajectory of the yellow border in Fig. 13

algorithm. Although the optimised $A^*$ algorithm adds a little calculating time compared to the $A^*$ algorithm, most important is that the optimised $A^*$ algorithm reduces the number of vertices to 0. Therefore, the optimised $A^*$ algorithm has a stable performance in shortening length and smoothing path. It is more suitable to design a real time and feasible path for $UGV$ to track in a real environment.

6.3 Simulation in a physical experiment

All the algorithms are applied in a physical experiment. The result of target recognition is shown in Fig. 15. Based on the result, the optimised $A^*$ algorithm was used to plan a feasible trajectory, the result is shown in Fig. 16.

7 Conclusion

In this paper, a typical rescue application is considered to study the map building and path planning. Image pre-processing methods were used to improve the recognition accuracy of the target and obstacles. Especially, the $SURF$ algorithm and image segmentation have the strong robustness in map building. In addition, the $HSV$ spatial model is more suited for illumination environment. Furthermore, the coordinate transformed system is to convert the map in pixels to a metric map. Finally, the designed path by the optimised $A^*$ algorithm is shorter and smoother for the $UGV$ to track. All experimental results prove that the practicability of the proposed algorithms are useful.

8 Acknowledgments

This research was supported by The Outstanding Young Researcher of Guizhou Province [2017] 5630 and Technology Research and Development Program of Guizhou Province [2017] 2027.
References

[1] Kim, J., Kim, Y.: ‘Moving ground target tracking in dense obstacle areas using UAVs’, IEEE Proc. Vol., 2008, 41, (2), pp. 8552–8557.

[2] Tang, Z., Oguzner, U.: ‘Sensor fusion for target track maintenance with multiple UAVs based on Bayesian filtering method and hospitality map’. Proc. IEEE Conf. on Decision and Control, 2003, Maui, Hawaii, 2004, vol. 1, pp. 19–24.

[3] Chaimowicz, L., Grocholsky, B., Keller, J., et al.: ‘Experiments in multirobot air-ground coordination’. IEEE Int. Conf. on Robotics and Automation, 2004. Proc. ICRA, New Orleans, USA, 2004, vol. 4, pp. 4053–4058.

[4] Chaimowicz, L., Cowley, A., Gomez-Ibánez, D., et al.: ‘Deploying air-ground multi-robot teams in urban environments’, in ‘Multi-robot systems: From swarms to intelligent automata volume III’ (Springer, Netherlands, 2005), pp. 223–234.

[5] Li, J., Deng, G., Luo, C., et al.: ‘A hybrid path planning method in unmanned air/ground vehicle (UAV/UGV) cooperative systems’, IEEE Trans. Veh. Technol., 2016, 65, (12), pp. 9585–9596.

[6] Feng, G., Zheng, W., Qi, S., et al.: ‘Theoretical and experimental study of air-ground cooperative navigation’, J. Univ. Sci. Technol. China, 2012, 42, (5), pp. 398–404.

[7] Aghaeeyan, A., Abdollah, F., Talebi, H.A.: ‘Robust cooperative control in the presence of obstacles’. Electrical Engineering, Mashhad, Iran, 2013, pp. 1–6.

[8] Ganeshmurthy, M.S., Suresh, G.R.: ‘Path planning algorithm for autonomous mobile robot in dynamic environment’, Proc. 3rd Int. Conf. on Signal Processing, Communication and Networking (ICSCN), Chennai, India, 2015.

[9] Zhu, Z., Xiao, J., Li, F., et al.: ‘Global path planning of wheeled robots using multi-objective memetic algorithms’, Integr. Comput.-Aided Eng., 2015, 22, (4), pp. 387–404.

[10] Zhu, Z., Wang, F., He, S., et al.: ‘Global path planning of mobile robots using a memetic algorithm’, International Journal of Systems Science, 2015, 46, (11), pp. 1992–1993.

[11] Hu, Q., Zhao, J., Han, L.: ‘Cooperative path planning for intelligent vehicle using unmanned air and ground vehicles’. Proc. of 2017 Chinese Intelligent Systems Conf., Mudanjiang, China, 2018.

[12] Horowitz, R., Varaiya, P.: ‘Control design of an automated highway system’, Proc. IEEE, 2000, 88, (7), pp. 913–925.

[13] Wang, F., Zhu, Z.: ‘Global path planning of wheeled robots using a multi-objective memetic algorithm’, Integr. Comput. Aided Eng., 2015, 22, (4), pp. 387–404.

[14] Shi, Q., Zhao, J., Han, L., et al.: ‘Dynamic lane tracking system based on multi-model fuzzy controller’. IEEE Int. Conf. on Mechatronics and Automation, Takamatsu, Japan, 2016, pp. 873–877.

[15] Yang, S.: ‘Improved SURF image registration algorithm’ (University of Electronic Science and Technology of China, China, 2008).

[16] Yufeng, Z.: ‘Research on moving object recognition and matching of the crane hoisting system’ (Hebei University, China, 2014).

[17] Xiaochuan, Z.: ‘Interpretation and improvement of modern digital image processing technology’ (Beihang University Press, China, 2012).

[18] Wei, Z.: ‘Quad-rotor aircraft target recognition and tracking based on vision’ (Nanjing University of Aeronautics and Astronautics, China, 2014).

[19] Cheng, J., Jun, H., Kang, A.: ‘Dynamic path planning based on an improved RRT algorithm for RoboCup robot’. Robot, 2017, 39, (11), pp. 8–15.

[20] Belt, M.: ‘Hyperstar: a multi-path Astar algorithm for risk averse vehicle navigation’. Transp. Res. B Methodol., 2009, 43, (1), pp. 97–107.

[21] Qi-lei, Z., Tai-liang, G.: ‘Global path planning algorithm of robot based on multistage decision’. Comput. Eng., 2016, 42, (10), pp. 296–302.