Relative pose estimation for vision-based UAV vertically landing on the ship

S Dou, J J Zhang, X Y Wang and Q Zhang
School of Instrument Science and Opto-electronics Engineering,
Beihang University, Beijing 100191, China
zhangjingjuan@buaa.edu.cn

Abstract. A method based on monocular vision guidance of autonomously and vertically landing on the ship is proposed aiming to estimate the relative pose information between the UAV and ship and to provide the important information for the follow-up flight control system. This paper mainly involves corner points extraction in the region of interest (ROI) and pose estimation based on corner points. Through collecting the self-designed feature plane images, the method extracts the target points of the ROI by using HSV space and invariant moments. Then the relative pose information based on the principle of homography matrix is estimated by the information of points. In the measurement experiment, the rotation of rotary table imitates the movement of ship. The relative pose information is estimated by extracting 12 corner points with the 800×600-pixel monocular camera. Experiments show that the method can achieve maximum 1.7° relative posture measurement error and 1.4% relative position measurement error. This method has good robustness and high accuracy.

1. Introduction
Vertically landing on ship is a very dangerous and demanding process in the whole automatic flight of the UAV, which is mainly caused by the bad landing environment. If the UAV lands on ship accurately, there must be a stable navigation system. At home and abroad, the study of landing accurately on ship navigation system mainly includes: inertial navigation system (INS) [1], radio navigation system (RNS), global positioning system (GPS) [2] and Doppler navigation system. Compared with the traditional approaches of the UAV landing, the approach of vision guidance [3] has the following advantages: Firstly, this method has strong anti-interference ability. The visual sensor belongs to the passive sensor, which can image the light reflected by the object without the influence of electromagnetic interference or accumulated errors. Secondly, the sensor is small in size and highly adaptable, capable of capturing large amounts of information and sensitive to movement information. The most important step of the approach based on vision guidance is to estimate the relative pose. In paper [4], a special landing platform was selected to estimate pose information through corner points but this method has weak anti-interference ability and is easy to match the wrong points. A fish-eye camera and a lens camera constituted a hybrid stereoscopic vision system where the height of the UAV was estimated by plane scanning [5]. In addition, the multi-rotor UAV installed two cameras to match the histogram and template of the collected images and achieved real-time estimation of the height [6]. However, the above two methods involve the tedious calibration of binocular camera and the calculation process of pose takes a lot of time. The micro-infrared thermal imager detects spots on the ground, so as to calculate the relative pose of the UAV [7]. Moreover, if
the UAV realizes to land on ship autonomously, adding feedback control to this scheme is essential because the situation of sea level is complex, it is necessary to accurately extract the target area and corner points and feedback to the system in real time.

In view of the above problems, this paper designs a set of autonomously landing on ship system of the UAV. Firstly, the second chapter introduces the coordinate system required by the algorithm and the overall control strategy framework. Secondly, the third chapter designs a feature plane that contains a large amount of corner point information, and furthermore an algorithm that uses the matching points to estimate the relative pose information between the ship and the UAV. Thirdly, the fourth chapter is the experimental verification of pose estimation algorithm. Finally, the fifth chapter is the conclusions.

2. Overview of vertically landing on the ship system

2.1. Necessary coordinate system definitions

To facilitate the introduction of the system framework, this section lists the required coordinate systems as follows.

As is shown in figure 1(a), the ship body coordinate system is $O_s - X_s Y_s Z_s$. The $Y_s$-axis is in the symmetrical plane and points straight ahead of the ship, the $Z_s$-axis is also in the symmetrical plane and vertical with the $Y_s$-axis and the $X_s$-axis is defined according to the right-hand rule. According to self-designed feature image, the world coordinate system is established as $O_w - X_w Y_w Z_w$. The original point $O_w$ is located at a corner point, the $X_w$-axis is in the feature plane and parallels to the $X_s$-axis, the $Y_w$-axis is also in the feature and opposite the $Y_s$-axis and $Z_w$ is defined by the right-hand rule. So there’s a relationship here:

$$p_i = C_w^o p_o + T_w^o$$

In the figure 1(b), the UAV body coordinate system is fixed to the UAV. $X_u$-axis points to the front of UAV, $Y_u$-axis points to the right and $Z_u$-axis follows the right-hand rule. Likewise, the camera coordinate system controlled by the cloud platform is defined as follows: $Z_c$-axis is perpendicular to the lens and points away, $X_c$-axis points to the right of the camera and $Y_c$-axis is defined according to the right-hand rule. Through the messages of the camera cloud platform, the relationship between the UAV and camera is:

$$p_b = C_b^c p_c + T_b^c$$

In any images from camera, there are two coordinate systems including image coordinate system $O - xy$ and pixel coordinate system $O_u - uv$ from figure 2. The relationship between them is:
Figure 2. Image coordinate system and pixel coordinate system

The gray area represents the collected image and the origin $O$ of the image coordinate system is the point where the light source point is perpendicular to the image plane. Moreover, $x$-axis and $y$-axis are parallel respectively with $u$-axis and $v$-axis.

\[
\begin{bmatrix}
    u \\
    v \\
    1
\end{bmatrix} =
\begin{bmatrix}
    1 / dx & 0 & u_o \\
    0 & 1 / dy & v_o \\
    0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
    x \\
    y \\
    1
\end{bmatrix}
\] (3)

2.2. The Overall flow diagram of the system

The flow diagram of the system is shown in figure 3. The system architecture can be described in brief as ROI extraction (step1), corner points extraction (step2), and pose estimation (step3).

Figure 3. The flow diagram of the system

The next content is the introduction to the three-step relationship. The step 1 is to extract the ROI from acquired image. If this step goes well, the ROI will be used for the next step. Otherwise, the platform will rotate again. As for the step 2, after extracting the corner points, the system judges whether the target corner points are successfully matched. If the match is successful, the step 3 will be taken. If not, rough information will be sent to the flight control system. In this paper, the last step uses the points to estimate precise information such as position and posture to the final control system.

3. Pose estimation system

3.1. ROI extraction and corner points extraction system

In this paper, the design features are shown below:
The feature image is made up of three black rectangles and a red circle and also shows the actual dimensions between the corner points. Let the point 1 be the origin of the world coordinate system. The coordinates of each point are shown in Table 1.

Generally speaking, the way to extract colour space is to use HSV space which collected image is transformed from RGB space to. Threshold segmentation is performed for H space. The result is shown in Figure 5(a). Threshold binarization is carried out by using S space in order to identify ROI. The original image is then converted to grayscale image to facilitate corner point extraction. The ROI extraction is shown in Figure 5(b).

| Label | Coordinate(mm) | Label | Coordinate(mm) |
|-------|----------------|-------|----------------|
| 1     | (0,0)          | 2     | (0,23)         |
| 3     | (6.5,33)       | 4     | (6.5,51)       |
| 5     | (10,61)        | 6     | (10,70.5)      |
| 7     | (45.5,75.5)    | 8     | (45.5,61)      |
| 9     | (49.5,61)      | 10    | (49.5,33)      |
| 11    | (55.5,23)      | 12    | (55.5,0)       |

Figure 4. The feature image

Figure 5. ROI extraction result

There are many ways to extract corners and this paper uses the method of Harris because it is robust and easy to calculate. However, it’s not enough to find corner points. The step that every extracted point corresponds to every corner point in the world coordinate system is necessary. Taking the characteristics of the three rectangle in this feature graph, the invariant moment principle is used to mark the corner points. The result of marking corner points is shown in the Figure 6.
3.2. Pose estimation system
As the figure 7 shows, the pose estimation system mainly includes solving for homography matrix and pose model. In this paper, the homography matrix (H) represents the correspondence between the world coordinate system and image coordinate system. H is a 3*3 matrix which contains posture and position information. From the known corner points, this matrix can be easily obtained by DLT algorithm [8].

According to the principle of monomorphic matrix, the following can be obtained:

\[ p_w = Hp \]  

(4)

In equation (4), \( p_w \) and \( p \) are homogeneous expressions of coordinate points of two coordinate systems. In this paper, the most important section is to build pose models. \( \phi \), \( \theta \) and \( \psi \) represent the relative pitch, roll and yaw angle of two planes respectively in the figure 8.

About the model of \( \phi \), there is a point on the x-axis of the image coordinate system, which makes the projection infinitely far into the world coordinate system in the figure 9(a). By the same logic, the relative pitch angle can be obtained by using the y-axis projection. As for the model of \( \psi \), there is an angle called \( \psi \) between the y-axis of the image coordinate system after projection and the \( Y_w \)-axis of the world coordinate system in the figure 9(b). Moreover, the projection of the line and line can be derived from the projection of the point and point. The yaw angle is calculated by using the slope solution of the line equation which is projected on the world plane from y-axis. The specific solution is shown in equation (5), (6), and (7).
Figure 9. The models of φ and ψ

\[
\begin{bmatrix}
X_w \\
Y_w \\
0
\end{bmatrix} =
\begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{bmatrix}
x_p \\
0 \\
1
\end{bmatrix} \Rightarrow x_p = -\frac{h_{33}}{h_{31}} \Rightarrow \arctan \varphi = -\frac{f_{h_{31}}}{h_{33}}
\]

(5)

\[
\begin{bmatrix}
X_w \\
Y_w \\
0
\end{bmatrix} =
\begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{bmatrix}
y_p \\
0 \\
1
\end{bmatrix} \Rightarrow y_p = -\frac{h_{33}}{h_{32}} \Rightarrow \arctan \theta = -\frac{f_{h_{32}}}{h_{33}}
\]

(6)

\[
y' = \begin{bmatrix}
h_{11} & h_{21} & h_{31} \\
h_{12} & h_{22} & h_{32} \\
h_{13} & h_{23} & h_{33}
\end{bmatrix}(1,0,0)^T \Rightarrow y' = (h_{11}, h_{12}, h_{13}) \Rightarrow k = -\frac{h_{11}}{h_{12}}
\]

(7)

Figure 10. Position model

In the figure 10(a), the origin of the world coordinate system is projected at \(O_w\) in the \(\pi_1\). To figure out the distance of the line \(O,O_w\), it’s necessary to solve for the coordinates of \(p, O_w,\) and \(q\). And then through using the similarity between the triangle \(O,O_w, p\) and \(O_w, O, q\), the distance of \(O,O_w\) is figured out in equation (8), (9) and (10). Finally, as the figure 10(b) shows, the focal length information can be used to obtain the three dimensional position of \(O_w\) in camera coordinate system by the model of keyhole imaging.
\[
\begin{align*}
\begin{bmatrix} x_{O_x} & y_{O_y} & \lambda_1 \end{bmatrix}^T &= H^{-1}[1 0 0]^T \Rightarrow d_{O_{x_O}x} = \frac{1}{\lambda_1} \left[ (\lambda_1 f)^2 + x_{O_x}^2 + y_{O_y}^2 \right]^{1/2} \\
\begin{bmatrix} x_p & y_p & \lambda_2 \end{bmatrix}^T &= H^{-1}[1 0 0]^T \Rightarrow d_{O_{y_O}y} = \frac{1}{\lambda_2} \left[ (\lambda_2 f)^2 + x_p^2 + y_p^2 \right]^{1/2} \\
\begin{bmatrix} x_q & y_q & 0 \end{bmatrix}^T &= H^{-1}[1 0 0]^T, \quad \lambda O_pO_p \, p - \lambda O_qO_q \, q \Rightarrow d_{O_{z_O}z} = \frac{d_{O_{x_O}x}}{d_{O_{y_O}y}} | X_q |
\end{align*}
\]

4. The experiment verification of pose estimation

4.1. Experiment preparation

In order to better verify the validity of the algorithm, the rotary table is used instead of ship motion and accurate calibration of the camera is required. The higher the resolution, the longer the processing time. At the same time, there will be a positive proportional relationship between the resolution and corner point extraction. Considering the relationship between time and extraction effect, the resolution of image is 800*600. The paper uses MATLAB calibration kit of camera for calibration. The calibration parameters such as focal length, principal point and distortion of the small camera in this paper are shown in the table 2. Considering that the distortion is small, there is no need to compensate the corner points after extraction.

| Focal length(pixel) | Principal point(pixel) | distortion |
|--------------------|------------------------|------------|
| (721,723)          | (351,316)              | (0.00175,-0.00586) |

4.2. Experimental results and analysis

In this paper, the experimental of pose estimation mainly include the acquisition and verification of three relative attitude angles and the relative position. Based on the above algorithm and the existing rotary table, four experiments are set up in this paper.

4.2.1. Relative attitude angle experiment verification. In the actual experiment, the real relative angel is not easy to obtain. In order to verify the effect of the algorithm, the quantitative rotation angle of the rotary table is compared with the variation of the results.

Figure 11. Attitude angle measurement environment

In the figure 11, the red circle represents the camera, the red arrow represents the axis of rotation and the blue arrow does the direction of rotation. The difference of (a) and (b) is that image along its normal vector rotates 90° and the camera also does 90° through its \( Z \)-axis. Furthermore, the difference between (a) and (c) is that they use different axes to rotate image. Through three experiments, the relative pitch, roll and yaw angle can be obtained respectively.
As the result shows, each experiment used four different angles of rotation. As the graph shrinks, each rotation line of each image appears horizontal. However, in fact these angles fluctuate a bit. Through calculating the mean and variance of angle at each stage, data statistics are shown in the below tables.

### Table 3. the result of relative pitch angle.

| Position       | Mean(°)  | Standard Deviation(°) | Variation of pitch angle(°) |
|----------------|----------|-----------------------|----------------------------|
| Initial position | -35.4573 | 0.3116                | 11.1289                    |
| 10° rotating    | -24.3284 | 0.1313                | 9.7513                     |
| 20° rotating    | -14.5771 | 0.2530                | 9.0567                     |
| 30° rotating    | -5.5204  | 0.8312                |                             |

In the table 3, as the rotary table is in each position, the variation of pitch angle is around 10°. The maximum difference of variation is 1.1° which adapts to landing requirements.

### Table 4. the result of relative roll angle.

| Position       | Mean(°)  | Standard Deviation(°) | Variation of roll angle(°) |
|----------------|----------|-----------------------|----------------------------|
| Initial position | 30.8440  | 0.3132                | -11.6808                   |
| 10° rotating    | 19.1632  | 0.5395                | -9.2022                    |
| 20° rotating    | 9.9610   | 0.8895                | -10.9917                   |
| 30° rotating    | -1.0307  | 0.0597                |                             |

From the above table, as for roll angle, the analysis method is similar to the table 3.

### Table 5. the result of relative yaw angle.

| Position       | Mean(°)  | Standard Deviation(°) | Variation of yaw angle(°) |
|----------------|----------|-----------------------|----------------------------|
| Initial position | -4.1072  | 0.0106                | 90.8470                    |
| 90° rotating    | 86.7398  | 0.0286                | 89.3157                    |
| 180° rotating   | 176.0555 | 0.0136                |                             |
| 270° rotating   | 267.0185 | 0.0487                |                             |

As the table 5 shows, the change of yaw angle is around 90° and the standard deviation is under 0.05°. As the rotary table rotates, the line has good polarity, so it is sufficient and effective to use this algorithm to calculate relative angles.

#### 4.2.2. Relative position experiment verification.

Similar to the above experimental conditions, the camera and feature surface remain motionless in the process of position estimation. The distance between the camera and the origin of the plane is obtained by using a steel ruler. In this paper, two relative position information measured are 637 mm and 890 mm. The position information of frame by frame analysis which is carried out through the collected video is shown in the figure 13.
The data obtained is tabulated as shown in the table blow:

| Mean(mm) | Standard deviation(mm) | Maximum(mm) | Minimum(mm) | Percentage |
|----------|------------------------|-------------|-------------|------------|
| 637 mm   | 640.7843               | 641.5412    | 638.0245    | 0.470%     |
| 890 mm   | 902.9889               | 905.8412    | 890.7145    | 1.35%      |

To sum up, the information (637 mm) of relative position estimation is 640.7843 mm while the other one is 902.9889 mm. In other words, about the 890 mm information, its error with the actual accuracy is 1.35%. This means that the algorithm has good position measurement. However, the two graphs also have massive burr which does harm to the whole system. After repeated experiments, the reasons for this phenomenon are as follows: 1. The effect of light and the poor performance of the camera affect the quality of collected image. 2. The size of the feature image is relatively small. 3. The effect of colour target grey zone blurs the corner points.

The root of this phenomenon is the extraction of corner points. Although the effect on attitude angles is small, these problems should be avoided. There are several ways to reduce the impact: 1. The corner points are further strengthened and the size and spacing of feature graphs are selected appropriately. 2. Using subpixel angular point extraction and coordinate normalization can improve the accuracy. 3. Taking higher resolution will reduce the errors of corner point extraction.

5. Conclusions
In this paper, we designed a vision-based UAV vertically landing on the ship and the most important of these is the estimation of relative posture and position. First, we use the image information to extract the areas of interest and the target points by using the HSV space and Harris corner detection. Next, the extracted feature points are used for pose estimation. In this section, by using the relationship between two planes and building coordinate systems and mathematical model, the pose information is solved with geometric relation. The extraction time and estimation precision of this method can satisfy the UAV system. Finally, using the two known coordinate relationship between the characteristic plane and the ship, the camera and the UAV, we can get the relative posture and position information of the UAV and the ship.

References
[1] Zhang X, Liu P and Zhang C 2016 *Math. Probl. Eng.* 2016 1-13
[2] Koifman M and Bar-Itzhack I Y 2002 *IEEE Trans. Control Syst. Technol.* 7(4) 487-493
[3] Kong W et al. 2014 *IEEE/RSJ International Conf. on Intelligent Robots and Systems (Chicago)* 2014 American Elsevier pp 4797-4804
[4] Sharp C S, Shakernia O and Sastry S S 2001 *IEEE/RSJ International Conf. on Intelligent Robots and Systems (Hawaii)* vol 2 pp 1720-1727
[5] Eynard D et al. 2010 *IEEE/RSJ International Conf. on Intelligent Robots and Systems 2010* pp 646-651
[6] Jeon D, Cho K and Kim D H 2011 *Vision-Based Autonomous Landing for Small-Scale Unmanned Rotorcraft* (Moscow: Moscow State University Press) p 35
[7] Wenzel K E, Rosset P and Zell A 2010 *J. Intell. Robot. Syst.* 57(1-4) 297-311
[8] Hartley R and Zisserman A 2003 *Multiple View Geometry in Computer Vision* (Cambridge: Cambridge University Press) p 274