

# Research on Precise Landing of Rotor UAV Based on Multi-Sensor Fusion

Jintang Li \*

Engineering Technology College, Hubei University of Technology, Wuhan, Hubei, 430068, China

\*Corresponding Author: [ljt021113@outlook.com](mailto:ljt021113@outlook.com)

## ABSTRACT

With the rapid development of drone technology, the autonomous landing of drones has become a current research hotspot. This article introduces an autonomous drone landing system based on multi-sensor fusion. The system combines GPS, IMU, visual sensors, LiDAR, and other sensor data through Kalman filtering to achieve data fusion and improve the positioning accuracy and stability of the drone during landing.

## KEYWORDS

Drone; Multi-sensor fusion; Lidar; Visual sensor; Kalman filter

## 1. INTRODUCTION

With the rapid development of computer technology, rotorcraft drones have been widely used in multiple fields such as aerial photography, agricultural plant protection, power inspection, film and television shooting, and post-disaster search and rescue. These applications require drones to have high-precision autonomous landing capabilities to ensure task success and safe recovery of the drone. However, the problem of low precision in autonomous drone landing has become increasingly prominent, especially in situations where GPS is unavailable or the environment is complex.

Traditionally, drones rely on inertial navigation systems and GPS for positioning, but these methods are challenged by environmental limitations and positioning accuracy. In recent years, computer vision-based autonomous drone landing technology has received attention. This technology achieves rapid detection and recognition of landing markers by capturing real-time target and environmental information through an onboard computer, providing accurate position and attitude information for the drone. However, visual methods are also affected by factors such as lighting, weather, and visual perception uncertainty.

In order to overcome these limitations, LiDAR, a sensor with high environmental adaptability, is considered for use in autonomous landing of unmanned aerial vehicles. LiDAR can acquire accurate three-dimensional information about the surrounding environment, but its beam is extremely narrow and the search range is limited. Therefore, combining multiple sensors such as vision and LiDAR for external environmental perception and self-shape monitoring of unmanned aerial vehicles, and improving the accuracy and robustness of UAV position and attitude estimation through sensor fusion algorithms, has become a hot topic in current research.

In summary, the development of autonomous landing technology for unmanned aerial vehicles requires comprehensive consideration of factors such as sensor selection, data processing algorithms,

and environmental adaptability, in order to achieve a high-precision, high-robustness autonomous landing system.

## 2. OVERALL FRAMEWORK

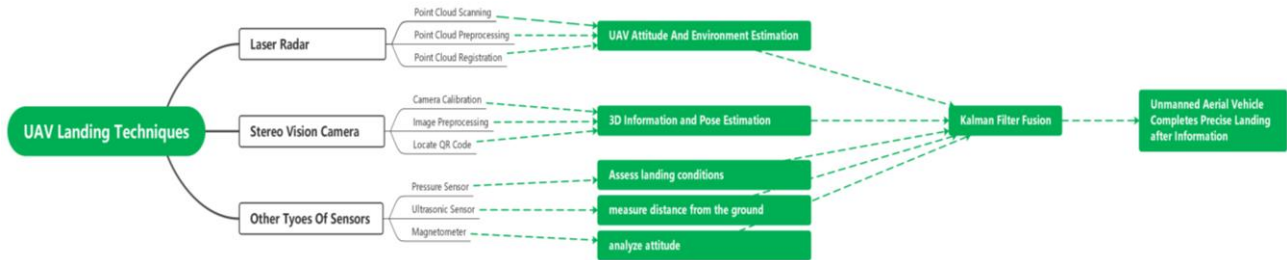


Figure 1. Frame Diagram

## 3. LIDAR RADAR

This article adopts the LiDAR to obtain the landing environment of the unmanned aerial vehicle. Firstly, the point cloud of the unmanned aerial vehicle landing area obtained by LiDAR is analyzed, then the point cloud data of the landing area is preprocessed, and finally the registration of the point cloud data is completed. The process of point cloud data processing is shown in Figure 1.

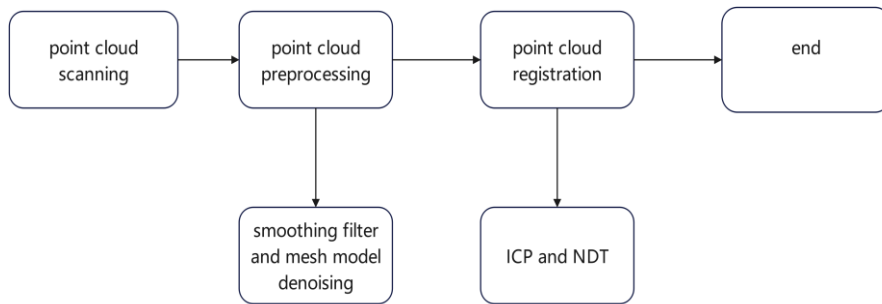


Figure 2. LiDAR Procedure Diagram

Lidar measures the distance, position, angle, and other information of the target object, obtaining the three-dimensional spatial data of the scanned area. The pulse ranging method is where lidar first emits a light pulse and simultaneously starts a timer. When the receiving system detects the light pulse reflected back from an obstacle, the timer stops. The time recorded by the counter is the time taken for the light pulse to travel from emission to reception [1]. The method used in this paper is the pulse ranging method, with the formula:

$$L = ct/2.$$

Here,  $c$  represents the speed of light,  $t$  is the time interval of reflection, and  $L$  is the measured distance to the target object.

### 3.1. Characteristics of Point Cloud Data

Lidar consists of a laser ranging system, control recording unit, optical scanning unit, and imaging equipment, among other components. Point cloud data is primarily composed of a collection of three-dimensional point coordinates distributed on the surface of objects, which include coordinate information in the X, Y, and Z dimensions. Each point represents a specific location on the surface of an object, and a large number of points can precisely describe the shape and structure of the object[2]. As the lidar moves with the drone, it performs a wide-area scan of the landing position,

resulting in point cloud data that exhibits the following five characteristics: discrete distribution, uneven data distribution, sparsity and disorder, a large number of repeated point clouds, and when the laser does not produce an echo signal, it causes missing point cloud data.

### **3.2. Processing of Point Cloud Data Sampling**

In general, the point cloud data model captured by three-dimensional scanning equipment presents significant challenges in storage, transmission, and computation due to its large data volume and dense information points. Therefore, it is particularly important to perform down-sampling processing on these data. This article will use a voxel grid filter to down-sample the point cloud. This method first constructs a three-dimensional voxel grid coordinate system based on the original point cloud data, and then, within each voxel, uses the centroid of all points within that voxel to approximate the other points in the voxel, i.e., all points within each voxel are ultimately represented by a single centroid point[3]. Compared to the simple voxel center approximation method, although this approach is slower in processing all voxels to obtain the point cloud, it can more accurately preserve the surface details corresponding to the sampled points.

### **3.3. Denoising of Point Cloud Data**

During the process of data acquisition on site, due to the influence of instruments and environment, there will be some errors in the collected data, which is called point cloud noise data. It reduces the quality of point cloud data and also affects the accuracy of the reconstructed model in later modeling.

When denoising point cloud data, the characteristic distribution of the collected points should be considered. Suitable denoising methods should be selected based on these characteristic distributions. Currently, point cloud data is mainly classified into linear point cloud data, arrayed point cloud data, triangulated point cloud data, and scattered point cloud data according to their characteristic distribution. Among them, linear point cloud data and arrayed point cloud data belong to ordered point cloud data. The triangulated point cloud data is in a state between ordered and unordered point cloud data, belonging to unordered point cloud data. For ordered point cloud data and partially ordered point cloud data, due to the clear topological relationships between the data, we can effectively remove noise by using smoothing filter operators [4]. However, for scattered point cloud data, since there is no regular pattern to follow and the arrangement is flexible, and there is no topological structure between the data, we can first grid the scattered point cloud data during the denoising process and then apply the grid model denoising method; alternatively, we can directly use denoising methods designed for scattered point clouds for denoising.

### **3.4. Point Cloud Registration**

During the scanning process, due to the limitations of its scanning range, the point cloud information obtained by LiDAR is often incomplete. To achieve a more comprehensive understanding of the environmental space [5], it is necessary to scan with the LiDAR from multiple different angles. Subsequently, we need to precisely align the point clouds obtained from each angle to form a complete three-dimensional environmental model. There are mainly three methods for point cloud registration: point-based registration, feature-based registration, and math feature-based registration [6]. Currently, the two most commonly used methods are the ICP (Iterative Closest Point) algorithm and the NDT (Normal Distributions Transform) algorithm. In this paper, we will also employ these two algorithms for point cloud registration, with the aim of obtaining a more accurate and complete three-dimensional environmental model.

#### **3.4.1. Nearest Neighbor Iteration Algorithm**

The ICP algorithm, based on data registration methods, employs the nearest point search technique to address registration issues involving free-form surfaces. The core concept of the algorithm is to

iteratively optimize the minimum square error between two point clouds, thereby identifying the optimal point cloud registration outcome [7]. Initially, the input point cloud data is preprocessed, which includes operations such as outlier removal and downsampling to enhance the stability and efficiency of the algorithm. One of the point clouds is selected as the reference point cloud, and all points of the other point cloud are projected onto the coordinate system of the reference point cloud to achieve an initial alignment result. For each point in the reference point cloud, the algorithm finds the nearest point in the target point cloud to establish a correspondence.

Based on the point correspondence relationship, the rotation and translation transformation matrix is solved by the least squares method, which is used to align the target point cloud to the reference point cloud coordinate system. The target point cloud is applied with the transformation matrix to obtain the updated target point cloud.

The mean square error (MSE) or other similarity metrics are calculated to determine whether the preset convergence condition is met. If not, the point cloud matching step is returned to continue iterating and optimizing. When the convergence condition is satisfied, the final registration result is outputted, which achieves the best match in shape and position between the two point clouds.

### 3.4.2. Normal Distributions Transform (NDT) Algorithm

The Normal Distributions Transform (NDT) algorithm is a widely used registration algorithm in point cloud data processing[8]. It is a point cloud registration algorithm based on probability density distribution. NDT maps the target point cloud and the source point cloud data onto multi-dimensional normal distributions, describes the spatial distribution of the point cloud using probability density functions, and finds the best match between the two point clouds through optimization algorithms. Preprocessing such as filtering and downsampling is performed on the source and target point clouds to improve registration efficiency.

The space where the source point cloud is located is divided into several small grids, and the number of points in each grid is counted to construct a probability density distribution model for the source point cloud. Based on the initial rigid body transformation parameters (such as rotation matrix and translation vector), the target point cloud is mapped into the coordinate system of the source point cloud. For each point in the mapped target point cloud, its probability density value in the grid, i.e., the probability of the point belonging to the source point cloud, is calculated. The optimal rigid body transformation parameters are iteratively solved through optimization algorithms such as Newton's method and Gauss-Newton method, minimizing the error between the mapped target point cloud and the probability density distribution model of the source point cloud. The algorithm is judged to converge based on preset convergence conditions such as the number of iterations and error threshold. If it converges, the optimal rigid body transformation parameters are outputted; otherwise, further iterations are performed to optimize.

## 3.5. Chapter Summary

In this chapter, the spatial point cloud data is obtained by using LiDAR, and the method of scanning the minimum landing area in the hovering stage with LiDAR installed on the drone is introduced. At the same time, some methods of point cloud preprocessing and the registration of point cloud data after preprocessing operations are introduced, mainly including the Iterative Closest Point (ICP) algorithm and Normal Distribution Transform (NDT) for point cloud registration.

## 4. BINOCULAR VISION

Image acquisition is carried out through parallel binocular cameras in computer vision, and the location and recognition of specific landing QR code, the calculation of the center point of the sign, and the estimation of the attitude angle parameters of the unmanned aerial vehicle.

#### 4.1. Distortion Correction and Calibration of Parallel Binocular Cameras

The primary purpose of binocular camera calibration is to obtain the camera's intrinsic and extrinsic parameters, as well as distortion parameters, to facilitate three-dimensional reconstruction and measurement. This involves determining the intrinsic parameters of each camera (such as focal length, optical center, etc.) and the extrinsic parameters (such as the camera's position and orientation in the world coordinate system), and correcting the lens distortion of the camera. Methods such as Zhang's calibration method or other similar calibration techniques [9] are used. These methods involve capturing multiple images of a calibration board with known geometric features (such as a checkerboard) from different angles and positions, and using image processing techniques to identify the corner points, thereby calculating the camera's intrinsic and extrinsic parameters. Initially, several photos of the calibration board at various angles and locations are taken, then the corner points on the calibration board are extracted, and finally, the extracted corner point information is used for camera calibration to compute the camera's intrinsic and extrinsic parameters and distortion parameters.

Due to manufacturing and installation errors, camera lenses can cause image distortion, such as radial and tangential distortion. These distortions can affect the accuracy of the image and the effectiveness of three-dimensional reconstruction. The distortion parameters obtained through the calibration process can be used to correct the image [10]. This typically involves remapping each pixel in the image to eliminate the effects of distortion. After distortion correction, the image can more accurately reflect the shape and positional relationships of objects in the real world, thereby improving the accuracy of three-dimensional reconstruction and measurement.

#### 4.2. Study on QR Code Positioning Detection Method for Landing Landmarks

When selecting landing landmarks, we emphasize that they must be rich in landing feature information and uniqueness. Therefore, QR codes become an ideal choice [11]. During the landing process, the drone uses long-range positioning algorithms and stereo binocular camera technology to determine the three-dimensional position of the QR code. Subsequently, the flight control system guides the drone towards the landmark until it approaches the predetermined distance. At this point, the system switches to close-range positioning and estimates the drone's attitude angle to ensure a safe and precise landing.



Figure 3. Landing Target QR Code

#### 4.3. Research on Short-Range Positioning Method Using QR Codes

In the autonomous landing system for drones based on parallel binocular computer vision, when the drone approaches the landing marker, the parallel binocular camera mounted on the drone can clearly capture the edges of the landing marker's QR code. By extracting the edge contours using the Canny operator in image processing [12], and through the nested information of the edge contours, the position detection pattern of the QR code can be obtained, preparing for the close-range positioning of the QR code. The specific implementation steps are as follows:

- (1). Image input, perform image preprocessing such as Gaussian filtering to remove noise.

- (2). Process the preprocessed image with the Canny edge detection algorithm to extract the edges in the image.
- (3). Search for contours in the edges[13], identify edges with more than 4 nested layers to obtain the Position Detection Pattern.
- (4). If the number of position detection images obtained in step 3 is not three, error correction must be performed to detect them.
- (5). Once three positioning detection images are obtained, the smallest bounding rectangle of the marker can be determined. By calculating the outermost vertices of the detection pattern, the position of the fourth vertex is obtained, ultimately identifying the QR code area.

#### **4.4. Research on Long-Range Positioning QR Code Method**

When the QR code is at a distance, edge detection information cannot obtain the nested information of the graphic contour. At this point, without using the three Positioning Mark (localization markers) to find the QR code position, treat the QR code as a whole:

- (1). Load the image and perform preprocessing by applying high-pass filtering to remove noise, etc.
- (2). Convert to grayscale, perform histogram equalization, enhance contrast, and filter.
- (3). Apply the OTSU thresholding method [14] to binarize the image.
- (4). Conduct three rounds of dilation and erosion morphological processing [15] to fully merge the black and white areas of the QR code into a single object.
- (5). Use a connected component analysis algorithm to count the sizes of connected components, set a minimum size for connected components, and perform an initial screening to obtain potential regions.
- (6). Then, make a judgment on the potential regions: calculate the centroid position of the connected component. Find the outermost square boundary of the connected component and calculate the ratio of the total area of the connected component to the area of the foreground within the outermost square boundary. If the ratio is greater than 0.9, it is determined to be the location of the QR code, and the centroid obtained is the position of the QR code. Otherwise, exclude that region and proceed to the next potential region judgment until the centroid position of the QR code is obtained.

#### **4.5. Chapter Summary**

In this chapter, the focus is on the study of distance measurement methods using a stereo camera system, and an analysis and introduction of the content structure features of QR codes are provided. Centered around the localization of QR codes, algorithms for localizing QR codes at close range and long range are proposed.

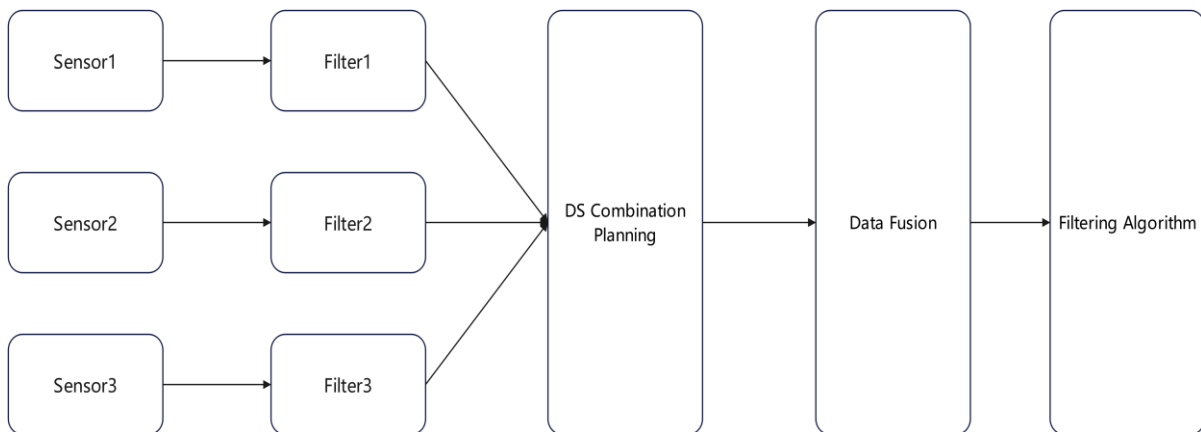
### **5. RESEARCH ON FUSION ALGORITHMS FOR OTHER SENSORS AND MULTISENSOR INTEGRATION**

In addition to lidar and binocular vision cameras, there are also other sensors that can provide accurate information for drone landings. For instance, pressure sensors can determine the drone's landing conditions, ultrasonic sensors measure the distance of the drone from the ground, and IMUs analyze the drone's attitude information.

## 5.1. Study on Multi-Sensor Fusion Methods

Kalman Filter: Utilizes linear system state equations and observation data to continuously update the estimated state values through recursive methods.

Based on the system dynamic model, it predicts the current state using the state estimate from the previous moment and the control input. By combining the predicted state with the observation data and using the Bayesian update rule, it calculates the optimal state estimate. DS Evidence Theory [16]: Used for fusing measurement data from different sensors to obtain dynamic weighting values. By calculating the confidence (or credibility) of each sensor's data and weighting the measurement data based on these confidences [17]. Combining the predicted values from the Kalman filter with the weighted values obtained from the DS evidence theory, it processes the measurement information received by the sub-sensors. After the fusion of the measurement information, it reassigns the values to further improve the accuracy and reliability of the data. After data fusion and reassignment, the Kalman filter [18] is applied again to the processed data to filter out any possible noise and interference [19]. Through these steps, combining the Kalman filter with the DS evidence theory significantly improves the estimation accuracy of the sensor data on unmanned aerial vehicles, thereby providing more accurate data support for the autonomous landing of UAVs.



**Figure 4.** Flowchart of Multi-Sensor Fusion Algorithm

## 5.2. Summary of This Chapter

The introduction mentioned that various other types of sensors can provide more accurate information for drones. By processing the measurement data with Kalman filters and Dempster-Shafer (DC) evidence theory, dynamic weighting values for the measurements are obtained. The fusion processing of the measurement information received by the sub-sensors is carried out, and a secondary assignment of the measurement information is performed. The data processed in this way, after filtering, significantly improves the estimation accuracy.

## 6. CONCLUSION

This paper proposes a multi-sensor fusion-based drone landing method by integrating various sensor data and deep learning techniques. Future research will further optimize sensor configurations and algorithm design to enhance the system's real-time performance and robustness. Additionally, it will explore drone landing technologies across more application scenarios to promote the comprehensive development of drone technology.

## REFERENCES

- [1] Pan, L. (2021) Research on 3D Scene Reconstruction Based on Airborne LiDAR. Thesis of Dalian University of Technology.
- [2] Li, Y. (2020) Research on the Extraction of Existing Track Centerline and Plane Alignment Fitting Method Based on Point Cloud Data. Thesis of Southwest Jiaotong University.
- [3] Zou, J. (2021) Voxel Filtering Point Cloud Denoising Algorithm Based on Region Growing. *Surveying and Mapping Geography Information*, 46(02):63-66.
- [4] Li, M. (2022) Research on Denoising Algorithms for 3D Laser Point Cloud Data. *Information Technology and Informatization*, (09), 47-50
- [5] He, H., Lai, J., Lv, P., et al. (2021) High-precision registration and localization method for LiDAR point clouds in structured scenes. *Navigation Positioning and Timing*, 8(01): 133-142.
- [6] Zhang, X., Yang, B., Li, Y., et al. (2018) A method of partially overlapping point clouds registration based on differential evolution algorithm. *Plos One*, 13 (12):209-227
- [7] Wu, P., Li, W., Yan, M. (2020) 3D scene reconstruction based on improved ICP algorithm. *Microprocessors and Microsystems*, 75(12):103064.
- [8] Ula, C., Temelta, H. (2013) 3D multi-layered normal distribution transform for fast and long range scan matching. *Journal of Intelligent & Robotic Systems*, 71(1):85-108.
- [9] Zhang, Z. (2000) A flexible new technique for camera calibration. *Tpami*, 22(11):1330-1334.
- [10] Chen, X., Jia, R. (2011) Calibration Method of Binocular Vision Cameras for Underground Mobile Robots. *China Coal*, 37(11): 59-63.
- [11] Cheng, S. (2016) Research on QR Code Localization Algorithm Based on Convolutional Neural Network. Thesis of Zhejiang University, 34-36.
- [12] Lu, Y. (2012) Research on Encoding and Image Processing Technology Based on Two-Dimensional Barcode QR Code. Thesis of Taiyuan University of Science and Technology, 44-46.
- [13] Li, C., Wong, T., Xu, N., Saxena, A. (2011) Feccm for scene understanding: Helping the robot to learn multiple tasks. In Video contribution in ICRA, 532-538.
- [14] J. Niebles, C.W. Chen, and L. Fei-Fei. (2010) Modeling temporal structure of decomposable motion segments for activity classification. In ECCV, 911-917.
- [15] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk. (2014) Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences*, 7(3): 1588-1623.
- [16] Dempster, A.P. (1967) Upper and Lower probabilities Induced by a Multivalued Mapping. *Annals of Mathematical Statistics*, 38(2):325-339.
- [17] Shafer, G.A. (1976) *Mathematical Theory of Evidence*. Princeton University Press, Princeton.
- [18] Li, L., Ji, H., Luo, J., et al. (1976) Iterative Extended Kalman Particle Filter. *Journal of Xidian University: Natural Science Edition*, 34(2): 233-238
- [19] Zhang, K., Fu, L., Fan, Y., et al. (2006) Research on Optimization of Information Fusion Algorithm Based on Kalman Filter. *Journal of Chinese Inertial Technology*, 14(5): 32-35