drones

MDPI

Article

N-Cameras-Enabled Joint Pose Estimation for Auto-Landing Fixed-Wing UAVs

Dengqing Tang *, Lincheng Shen, Xiaojia Xiang, Han Zhou and Jun Lai

The College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; lcshen@nudt.edu.cn (L.S.); xiangxiaojia@nudt.edu.cn (X.X.); zhouhan@nudt.edu.cn (H.Z.); laijun@nudt.edu.cn (J.L.)

* Correspondence: tangdengqing09@nudt.edu.cn

Abstract: We propose a novel 6D pose estimation approach tailored for auto-landing fixed-wing unmanned aerial vehicles (UAVs). This method facilitates the simultaneous tracking of both position and attitude using a ground-based vision system, regardless of the number of cameras (N-cameras), even in Global Navigation Satellite System-denied environments. Our approach proposes a pipeline consisting of a Convolutional Neural Network (CNN)-based detection of UAV anchors which, in turn, drives the estimation of UAV pose. In order to ensure robust and precise anchor detection, we designed a Block-CNN architecture to mitigate the influence of outliers. Leveraging the information from these anchors, we established an Extended Kalman Filter to continuously update the UAV's position and attitude. To support our research, we set up both monocular and stereo outdoor ground view systems for data collection and experimentation. Additionally, to expand our training dataset without requiring extra outdoor experiments, we created a parallel system that combines outdoor and simulated setups with identical configurations. We conducted a series of simulated and outdoor experiments. The results show that, compared with the baselines, our method achieves 3.0% anchor detection precision improvement and 19.5% and 12.7% accuracy improvement of position and attitude estimation. Furthermore, these experiments affirm the practicality of our proposed architecture and algorithm, meeting the stringent requirements for accuracy and real-time capability in the context of auto-landing fixed-wing UAVs.

Keywords: pose estimation; auto-landing fixed-wing UAVs; ground vision system; block convolutional neural networks

Citation: Tang, D.; Shen, L.; Xiang, X.; Zhou, H.; Lai, J. N-Cameras-Enabled Joint Pose Estimation for Auto-Landing Fixed-Wing UAVs. *Drones* **2023**, *7*, 693. https://doi.org/10.3390/ drones7120693

Academic Editor: Oleg Yakimenko

Received: 12 September 2023
Revised: 15 November 2023
Accepted: 16 November 2023
Published: 30 November 2023

1. Introduction

Unmanned aerial vehicle (UAV)-based systems have recently gained prominence as highly efficient solutions across various application domains. Among the pivotal functionalities required for UAV operations, autonomous landing stands out as a critical capability. Nonetheless, achieving autonomous landing still presents formidable technical challenges. In addition to the intricate control aspect, which has been widely acknowledged as a challenging problem [1,2], the accurate state estimation of the aircraft poses another formidable task. For fixed-wing aircraft, compared to rotary-wing aircraft, landing is additionally complicated by the non-zero airspeed at the moment of touchdown. This circumstance causes the necessity to generate and realize control actions in a very short time. The important basis for the formation of these control actions is the pose (position and attitude) of the aircraft.

There are two alternatives for the aircraft's pose estimation. In the first one, onboard sensors provide the required information; in particular, the Global Navigation Satellite System (GNSS) combined with inertial navigation systems (INS). The second alternative relies on ground equipment to estimate the aircraft's pose, which is transmitted to the aircraft. The onboard approach often relies on stable satellite signals [3,4]. In addition,

conventional INS can be sensitive to magnetic fields in the environment and are influenced by temperature variations. Traditionally, addressing this issue necessitates the integration of supplementary sensors like visual navigation systems to enhance the aircraft's pose estimation during the landing process [5]. The offboard approach allows for significant reduction in the complexity and cost of the aircraft. This paper prefers the second option and proposes a pose estimation method for an aircraft's auto-landing based on a ground vision system, regardless of the number of cameras. A parallel ground vision system that combines outdoor and simulated setups with identical configurations is established for the method performance evaluated. The results indicate a significant improvement in pose estimation accuracy compared to state-of-the-art methods. In addition, the results show that the proposed architecture and method satisfy the stringent requirements for accuracy and real-time capability in the context of auto-landing fixed-wing aircraft. Compared with our previous work [6], which only focuses on the position estimation of the UAV, this paper aims to realize a joint estimation for the position and attitude without dependence on the number of cameras. In summary, the contributions of this paper are as follows:

(1) Independence of the Number of Cameras: Our proposed pose estimation method is versatile and compatible with ground vision systems utilizing any number of cameras, whether it is a monocular vision system ($N = 1$) or a stereo vision system ($N > 1$). Generally, the inclusion of more cameras enhances the system's resilience to measurement errors, such as anchor detection inaccuracies and pan-tilt unit (PTU) attitude errors.

(2) Elimination of Excessive Outdoor Data Requirement: Traditional approaches often entail an extensive and labor-intensive process of outdoor fixed-wing landing experiments to collect essential data, incurring significant time and resource costs. Our method, however, achieves accurate and robust outdoor UAV anchor detection by utilizing just 730 frames of data from two outdoor landings. This approach provides a viable solution for scenarios characterized by high outdoor experimental costs.

(3) Robust and Accurate Pose Estimation: Autonomous landing necessitates quick and dynamic movement of the UAV in a three-dimensional space, resulting in rapid changes in visual appearance, imaging backgrounds, and more. These spatial and temporal variations present formidable challenges to ground vision-based UAV pose estimation. Our method adeptly addresses these challenges, enabling more precise and robust anchor detection and pose estimation than state-of-the-art methods. This improvement has been validated through the replacement of the onboard GPS-INS positioning system with our method as the sole source of position and attitude data during the outdoor UAV auto-landing process.

(4) Simulated and Real Auto-landing Dataset: We have constructed a comprehensive dataset comprising eight simulated and four real landing videos, complete with labels such as target bounding boxes, anchors, and ground truth UAV pose information. This dataset, encompassing diverse conditions including varying wind directions and landing paths, serves as an invaluable resource for UAV detection and pose estimation research.

This paper is organized as follows: Section 2 provides a review of different options for aircraft pose estimation and analyzes their capabilities. In Section 3, we present the problem of UAV pose estimation based on ground vision. The module details, including CNN training, anchors detection, and 6D pose estimation, are then presented in Section 4. In Section 5, three simulated experiments and two outdoor experiments are presented to validate the feasibility, real-time capability, and the robustness of the proposed pose estimation method. The paper is finally concluded in Section 6.

2. Related Works

The GNSS and INS-integrated navigation system is still the most common means for aircraft pose estimation during auto-landing. However, due to weather factors and multi-wave effects in the landing area, the risk of safe landing accidents for aircraft significantly

increases when GNSS signals are occasionally interrupted or continuously disturbed. To improve the robustness of the auto-landing system, a GNSS-independent auto-landing guidance system has attracted researchers' attention. The early systems usually used radio, radar, or lidar to measure the distance between the aircraft and the landing area. Thales, a company in France, built a radio-based system to assist in UAV auto-landing on deck. This system was deployed on a H-6U "Bird" rotary-wing aircraft and a French Navy "Raphael" class frigate, completing a series of technical verifications such as long-range alignment and the mutual measurement of moving platforms. Yang et al. [7] provided a comprehensive review and analysis about radio frequency-based position estimation technology for aircraft. In 1999, The Swiss aerospace company RUAG developed the Object Position and Tracking System (OPATS). It uses laser measurement technology to measure the position and angle of drones and can support the guided landing of Swiss Air Force "patrol" drones. Kim et al. [8] used a lidar mounted on a ground vehicle and realized a lidar-guided aircraft auto-landing on the ground vehicle. The radar-based guidance system is common in both military and civilian fields. In 1996, Sierra Nevada Corporation (SNC) established a millimeter wave radar-based universal automatic recovery system for aircraft for the US military. Pavlenko et al. [9] proposed a 24 GHz secondary radar sensors-based aircraft localization method. By dropping several active beacons, the position is estimated via distance and angle information to the deployed beacons. Most of the above systems have been well applied, especially in military fields, which shows great accuracy and robustness. However, most of them are sensitive to magnetic fields and smog in the environment. Furthermore, some onboard auxiliary devices are necessary, which means modifications to the aircraft system are required.

Visual navigation systems offer an attractive alternative capable of mitigating drift issues by fusing prior knowledge with real-time data. The integration of a vision system augments the amount of environmental information accessible and enhances the robustness of self-state estimation. Using an onboard or offboard camera, the vision navigation system provides aircraft with accurate and real-time self-states, such as visual odometry (VO) [10], and visual simultaneous localization and mapping (VSLAM) [11]. Therefore, visual navigation is emerging as a viable auto-landing solution due to its intrinsic ability to incorporate rich environmental information.

Onboard vision: For the autonomous landing of rotary-wing aircraft, the aircraft's pose is often estimated by detecting the markers painted on the static platform utilizing an onboard camera [12,13]. For more complex scenarios such as landing on a ship deck, Wang et al. [14] realized that the autonomous landing of a Parrot AR Drone on a vessel deck platform only relies on onboard sensors. They simulated the movement of a ship deck with an attitude-programmable plate. Landing on a moving target [15–17] is more challenging compared with ship landing in terms of localization, trajectory planning, and control. For fixed-wing UAVs, however, it is challenging to track the ground marker throughout the entire landing process. This is because, unlike a rotary-wing aircraft, a fixed-wing aircraft is unable to hover. The landing of fixed-wing UAVs is further challenged [18] because even small errors in the guidance system may lead to system damage. The onboard vision was often used to detect the runway [19] and to estimate the relative aircraft's pose to the runway for autonomous landing. However, runway detection is often sensitive to the change in runway appearance. More importantly, for a successful landing, the closer the UAV is to the ground, the higher the accuracy requirement of the UAV pose. Nevertheless, when the UAV approaches the ground, the limited onboard field of view makes it difficult to obtain comprehensive visual information on the runway, which affects the accuracy of pose estimation. Without runway detection, the landmarks on the platform are tracked by the aircraft to provide information on the relative poses [20]. For the runway or landmarks, achieving accurate, robust, and real-time detection often requires abundant computing and storage resources, which is often unattainable for small UAVs. In addition, although an onboard solution can directly provide estimated pose for a control system without wireless

communication support, it usually requires modification of the aircraft itself, which is not feasible in many application scenarios such as military fields.

Ground vision: An alternative to the onboard vision-based system is the ground vision-based guidance system. It estimates the UAV pose and the pose data is then transmitted to the UAV for auto-landing control. Generally, the ground systems are equipped with a vast computational capacity which enables real-time pose estimation. In addition, using ground-based systems also reduces the load of onboard processing resources, which is often limited, especially for small aircraft. Furthermore, onboard systems such as the barometer and inertial measurement unit (IMU) are significantly sensitive to temperature and magnetic field variations, which does not affect the UAV pose estimation by the ground vision system. Y. Gui [21] proposed a relay guidance scheme to land a UAV by placing three groups of cameras on both sides of the runway so that the total field of view of the cameras covers the whole landing area. The *AUTOLAND* project [22] focused on the solutions that enable the autonomous landing of a fixed-wing aircraft on a Fast Patrol Boat. The ground monocular vision system has been tested to generate the relative pose of the aircraft concerning the camera. Relying on the ground stereo vision system, a saliency-inspired method [23] and a cascaded deep learning model [24] were proposed and developed to detect and track the aircraft in the images and then used the Extended Kalman Filter (EKF) proposed in our previous work [6] to estimate the position of the aircraft. Paying attention to the ground stereo vision-based fixed-wing aircraft detection and localization for autonomous landing, we design a ground stereo vision guidance system for validation. Unlike the multiple camera groups configuration in work [21], a pan-tilt rotary system was built to extend the ground camera's field of view by controlling the rotary to track the landing aircraft. Offline and online experiments demonstrate the feasibility and robustness of the proposed system [6,25]. Only focusing on the 3D position estimation of the aircraft, the above works explored vision-based guidance system schemes including monocular vision and infrared stereo vision.

The integration of multiple sensors utilizes the advantages of different types of sensors, thereby improving the adaptability to the environment. T. Nguyen et al. [26] built a system combining an ultra-wideband (UWB) ranging sensor with a camera to localize the aircraft by using the distance and relative displacement measurements. A vision/radar/INS-integrated shipboard landing guidance system was developed [20]. This system consisted of an onboard camera/INS-based motion estimator and an offboard radar-based relative position generator. X. Dong et al. [27] proposed an integrated UWB-IMU-Vision framework for autonomous approaching and landing of aircraft. Using simulated and real-world experiments in extensive scenes, the proposed scheme satisfied the accuracy requirement of auto-landing.

In addition to the position, attitude also plays an important role in the fixed-wing aircraft landing guidance and control system [28]. For accurate attitude estimation, the INS is commonly used [29]. Aided by the onboard inertial sensors, Yang et al. [29] proposed a bioinspired polarization-based attitude and heading reference system to self-determine the heading orientation in GNSS-denied environments. However, the INS measurement is often affected by internal or external factors such as drift, magnetic field, and temperature variation [30]. To improve robustness to environmental factors, the vision system is well applied [10,31]. To realize the tanker-UAV relative pose estimation during aerial refueling, Mammarell et al. [31] combined GPS and a machine vision-based system for a reliable estimation, where at least one order of magnitude improvement was achieved by using the EKF instead of other fusion algorithms. Using an off-board camera, an EKF with a nonlinear constant-velocity process model was proposed to estimate position and attitude for rotary-wing aircraft [32].

In this research, taking inspiration from advancements in pose estimation techniques for humans [33], human heads [34], and rigid objects [35], we have developed a ground vision system-based fixed-wing aircraft 6D pose estimation method that is independent of the number of cameras and boasts exceptional accuracy, robustness, and real-time

performance. However, existing state-of-the-art methods are primarily tailored for the pose estimation of slow-moving objects in a single-frame image. In a parallel vein, an attitude estimation method for fixed-wing aircraft is introduced [28], also leveraging a ground vision system and validating its accuracy and real-time capabilities through experiments. In contrast to these approaches, our method excels in that it not only estimates attitude but also jointly determines the position and attitude of the aircraft. Furthermore, during our outdoor experiments, the aircraft accomplished successful autonomous landings in environments where GPS and INS data were unavailable, highlighting that the onboard autonomous landing control system solely relied on our ground-based aircraft pose estimation.

3. Problem Formulation

Accurate pose estimation and flight control are the main two challenges for the autonomous landing of UAVs. This paper focuses on the first component, which is estimating the UAV poses $\{\boldsymbol{P_u}, \boldsymbol{A_u}\}$ with high accuracy and strong robustness according to the ground sensor data. To guarantee that the UAV remains in the camera's field of view throughout the entire landing, the camera is mounted on a pan-tilt unit (PTU), and the PTU has the ability to automatically search and track the UAV. Therefore, in addition to the images I, the PTU attitudes $\boldsymbol{A_P}$ are also included in the ground sensor data.

In general, the complete procedure of the pose estimation includes obtaining the object region of interest (ROI) first by object detecting, followed by detecting the object's features and estimating the object's poses. The procedure of the proposed pose estimation algorithm is summarized as the following mapping:

$$\{\mathrm{ROI}, \boldsymbol{A}_p, P_a\} \rightarrow \{\boldsymbol{P}_u, \boldsymbol{A}_u\} \tag{1}$$

where P_a denotes the system parameters gained by offline calibration.

In computer vision, the commonly extracted features are points, lines, and planes. During auto-landing, there are several points of the UAV that are always in the field of view and remarkably distinctive. Since the anchors' distribution has a significant impact on the pose estimation accuracy, the selection of the anchors considers their characteristics and distribution (for details, see Section 5). Here, we consider the following five UAV anchors: the endpoints of the left wing (*LW*), right wing (*RW*), left tail (*LT*), right tail (*RT*), and the front tripod (*FT*), as shown in Figure 1.

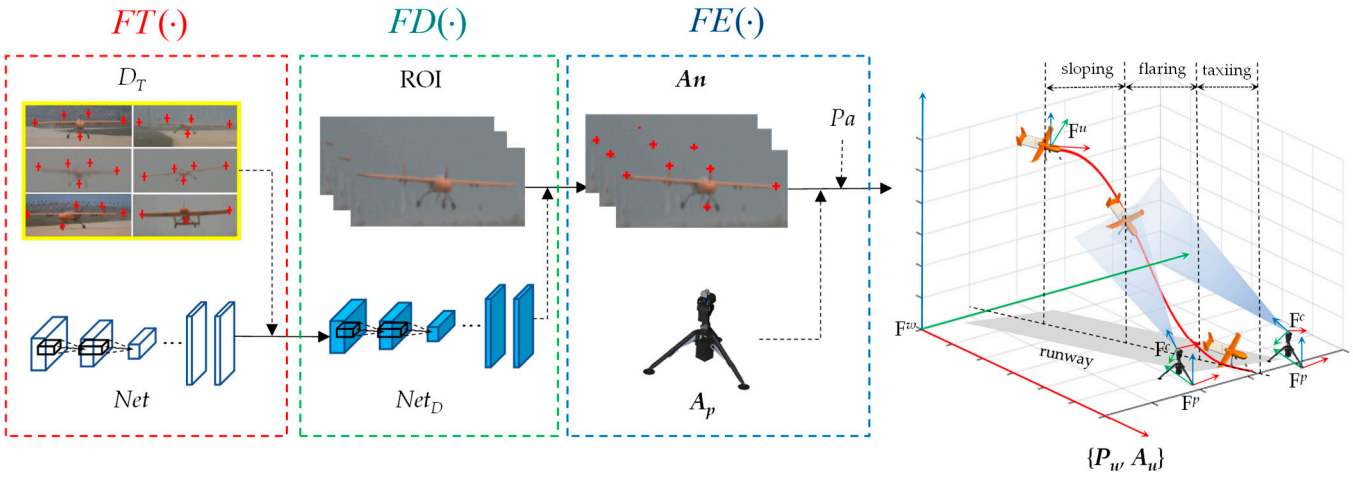

Figure 1. The diagram of the proposed algorithm is composed of the training operators $FT(\cdot)$, anchors detection operator $FD(\cdot)$, and pose estimation operator $FE(\cdot)$. The right trajectory plot shows the involved coordinate frames and three main periods for the auto-landing of the UAV.

We divide the UAV pose estimation into the three following operators: CNN training operator $FT(\cdot)$, anchors detection operator $FD(\cdot)$, and 6D pose filtering operator $FE(\cdot)$:

$$FT(\cdot) : \{D_T, Net\} \rightarrow Net_D \tag{2}$$

$$FD(\cdot) : \{ROI, N_D\} \rightarrow \boldsymbol{An} \tag{3}$$

$$FE(\cdot) : \{\boldsymbol{An}, \boldsymbol{A}_p, P_a\} \rightarrow \{\boldsymbol{P}_u, \boldsymbol{A}_u\} \tag{4}$$

As depicted in Figure 1, through offline training based on the training dataset D_T, the initial network *Net* evolves to be the network Net_D, which gains the ability to detect the anchors. The operator $FD(\cdot)$ detects the anchors and obtains the anchor locations $\boldsymbol{An}$ in image I. This is mainly performed by the anchor detection network Net_D. The final operator $FE(\cdot)$ estimates the UAV poses $\{\boldsymbol{P_u}, \boldsymbol{A_u}\}$ according to the anchor locations $\boldsymbol{An}$, system parameters $P_{a,}$ and real-time PTU attitude $\boldsymbol{A_p}$.

The right part of Figure 1 displays the involved coordinate frames in the autonomous landing system. The objective of the proposed algorithm is to estimate the quickly varied transformation between the world coordinate frame and the UAV body coordinate frame. For the left and right PTU coordinate frames, their origins in the world coordinate frame are known and constant. Since the camera is fixed on the PTU, the transformation between the camera coordinate frame and the corresponding PTU coordinate frame is also constant.

4. Methodology

This part provides a detailed description of the three operators: anchors detection operator $FD(\cdot)$, CNN training operator $FT(\cdot)$, and 6D pose filtering operator $FE(\cdot)$. The design of a Block-CNN for anchor detection is first given. Then, the details of training data generation and network training are presented. An EKF-based 6D pose estimation algorithm is finally described.

4.1. Anchor Detection Operator $FD(\cdot)$

Anchor detection is one of the core operators of the proposed pose estimation algorithm and its accuracy directly affects the pose estimation accuracy. Conventional anchor detection methods are often based on classical feature points such as Scale Invariant Feature Transform (SIFT) [36] and Oriented FAST and Rotated BRIEF (ORB) [37]. These handcrafted representations are, however, suboptimal compared to statistically learned features. The CNN learns the features and, therefore, archives significant advantages in computer vision applications in terms of accuracy and robustness [38]. One of the disadvantages is that the training of the CNN is computationally expensive and often needs GPU-like Compute Unified Device Architecture. This is, however, an essential disadvantage that limits the applications of CNNs [39]. In contrast, such computational resources can be easily made available on the ground for vision-based applications. This justifies the use of a CNN for accurate and robust anchor detection in ground vision-based systems.

Figure 2 illustrates the designed CNN: F with a 36×36 input and a 10×1 vector output. In general, the deeper the convolutional layer the greater the accuracy. In practice, the trade-off between accuracy and real-time capability needs to be incorporated into the final design. Considering the above factors, we employed 4 convolution layers and 1 fully connected layer. The output vector represents the positions of the five anchors in the image. Conventionally, the anchors' positions are estimated by the network F and can be directly used for estimating the next pose. Here, to improve the detection accuracy, we introduced the block strategy. Through partitioning the ROI, more pixel details are preserved after resizing as the network input, which helps the network extract more useful features. On the other hand, the block strategy enables the repeated detection of the same anchors. Using score-weighting averaging on the repeated detection results, the outliers' negative impact on detection accuracy is also reduced.

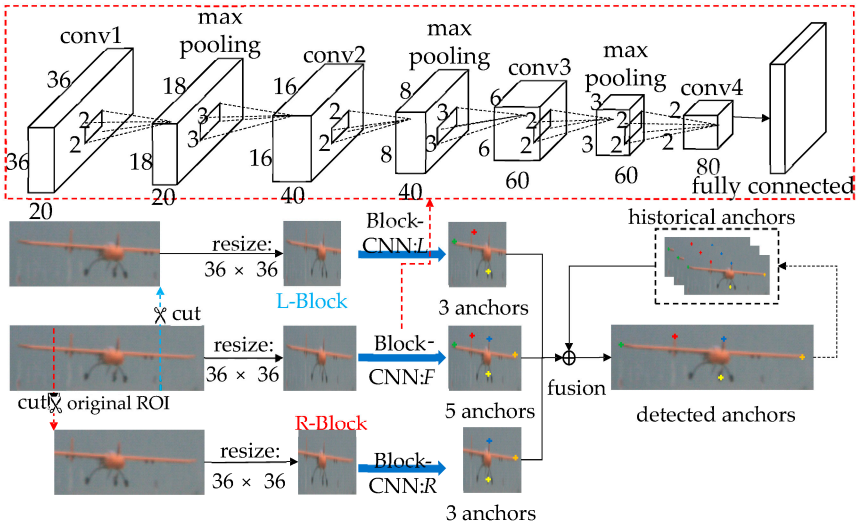

Figure 2. The Block-CNN architecture for anchor detection. The ROI is cut into 3 sub-ROIs. The anchors are then detected in sub-ROIs using the networks with the same structure. The network consists of 4 convolutional and 1 fully connected layer. The final detection results are generated after the fusion.

Since each anchor was distributed in a relatively fixed region of the ROI, several blocks are obtained by dividing the ROI. Each block contains some of the five anchors. As shown in Figure 2, two blocks are cut from the original ROI and then resized to be the size of 36 × 36. The L-Block contains the anchors *LW*, *LT*, and *FT*. Also, the anchors *RW*, *RT*, and *FT* are within the R-Block. Another two networks (*L* and *R*) have almost the same structure as network *F*, and they are used to detect part of the anchors in the R-Block and L-Block, respectively. Compared with *F*, the only difference in the structure is that the outputs are six-dimensional vectors. To promote detection accuracy, the anchors' locations in the ROI are then obtained by computing the score-weighted average of the Block-CNN outputs as described below.

Considering the movement continuity of the UAV, the location relationships among all of the anchors remain almost constant. For example, the tail anchors cannot move below the tripod anchor in the ROI. Therefore, we first set up the following constraints:

$$u_{LW} < u_{FT} < u_{RW} \tag{5}$$

$$u_{LW} < u_{LT} < u_{RT} < u_{RW} \tag{6}$$

$$v_{FT} < v_{LT} \tag{7}$$

$$v_{FT} < v_{RT} \tag{8}$$

where (u, v) denotes the image coordinates and their upper and lower indexes indicate the network and anchor categories, respectively. The outputs of the networks are ignored if they do not satisfy the above constraints. For the frame k, the final FT location (u_{FT}, v_{FT}) is the average of all the networks outputs:

$$(u_{FT}, v_{FT})_k = \frac{(u^F_{FT}, v^F_{FT})_k + (u^L_{FT}, v^L_{FT})_k + (u^R_{FT}, v^R_{FT})_k}{p} \tag{9}$$

Since three networks are used in this paper, the value of p is 3. Another 4 anchor locations are also predicted first, according to the historical anchor locations at step $k-1$, hence

$$(u,v)_{k|k-1} = (u,v)_{k-1} + ((u_{FT},v_{FT})_k - (u_{FT},v_{FT})_{k-1}) \tag{10}$$

Therefore, the final *LW* location computing steps are:

$$\Delta F_{LW} = \left\| (u^F_{LW},v^F_{LW})_k - (u_{LW},v_{LW})_{k|k-1} \right\| \tag{11}$$

$$\Delta L_{LW} = \left\| (u^L_{LW},v^L_{LW})_k - (u_{LW},v_{LW})_{k|k-1} \right\| \tag{12}$$

$$(u_{LW},v_{LW})_k = \frac{\Delta L_{LW}(u^F_{LW},v^F_{LW})_k + \Delta F_{LW}(u^L_{LW},v^L_{LW})_k}{\Delta F_{LW} + \Delta L_{LW}} \tag{13}$$

Another three anchor locations are also computed following the same steps as above.

4.2. CNN Training Operator $FT(\cdot)$

This module consists of training data generation and network training. Generating the data in the simulated system significantly improves data production efficiency due to the high labor and time costs of conducting outdoor landing experiments. Considering the great simulation performance of Gazebo for UAV dynamic characteristics during landing, we construct a Gazebo-based simulated environment following the same configuration as the outdoor environment. Using this method, data under different conditions such as different wind directions and different landing paths are efficiently generated. Furthermore, since the states of all objects, including UAV, PTUs, cameras, and involved parameters are known accurately in the simulation, the data supports autonomous labeling, and almost no manual labeling is needed.

Here we define the loss of network training by the Euclidean distance between the estimation and the ground truth. The Stochastic Gradient Descent (SGD) is employed to be the optimizer for the training. Before training, the samples were shuffled and every 10 samples were packed into a batch. We then train the network on a PC with 2 GPUs (GTX 3070). The network is trained with the maximum number of iterations of 40 k and an initial learning rate of 0.03, which is decreased by 10 at every 15 k iterations, and we use the weight decay of 0.0001 in the training process.

4.3. 6D Pose Estimation Operator $FE(\cdot)$

This module aims to recover UAV spatial pose from several anchors. The Perspective-N-Points (PNP) problem solution for monocular vision and the triangulation method for stereo vision are commonly used for the above problem. However, it does not consider the sensor data noise and takes advantage of the historical UAV states. To improve the robustness of the measurement error, such as the anchors' detection error, we establish an EKF to estimate the UAV position (x, y, z) and attitude (Euler angle (ψ, ϕ, θ)) in the world coordinate frame.

Let state $\mathbf{x}$ be defined as:

$$\mathbf{x} = \left[x, y, z, \dot{x}, \dot{y}, \dot{z}, \psi, \phi, \theta, w_\psi, w_\phi, w_\theta\right]^T \tag{14}$$

where $(\dot{x}, \dot{y}, \dot{z})$ and $(w_\psi, w_\phi, w_\theta)$ are linear and angular velocities, respectively. The state $\mathbf{x}$ at step k is predicted by the process model F_k:

$$\bar{\mathbf{x}}_{k|k-1} = F_k \bar{\mathbf{x}}_{k-1|k-1} \tag{15}$$

$$F_k = \begin{bmatrix} I_{3\times3} & \Delta t_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & \Delta t_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} \end{bmatrix} \tag{16}$$

where $\Delta t_{3\times3}$ denotes the 3×3 diagonal matrix with the diagonal element Δt defined as the time interval between steps $k-1$ and k. The state covariance matrix P is then obtained as:

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + G_k Q_k G_k^T \tag{17}$$

The measurement $\mathbf{z}$ contains the detected anchor locations in the images captured by the N cameras:

$$\mathbf{z} = \left[\begin{bmatrix} u_{LW}^l & \vdots & u_{RT}^l \\ v_{LW}^l & \vdots & u_{RT}^l \\ 1 & \vdots & 1 \end{bmatrix}_{3\times5} \cdots \begin{bmatrix} u_{LW}^r & \vdots & u_{RT}^r \\ v_{LW}^r & \vdots & u_{RT}^r \\ 1 & \vdots & 1 \end{bmatrix}_{3\times5} \right]_{3\times5\times N} \tag{18}$$

where N is the number of cameras, and each camera provides a set of 3×5 measurements of the detected anchor locations. The upper index marks the left and right cameras. In terms of the measurement model $h(\bullet)$, the pinhole camera model is employed to project the anchors into the images according to the predicted state $\bar{\mathbf{x}}_{k|k-1}$:

$$\mathbf{z} = h(\bar{\mathbf{x}}_{k|k-1}) = \frac{1}{\lambda} N_C T_P^C T_W^P T_U^W P_U \tag{19}$$

where λ is a scaling factor and P_U is the anchor locations matrix in the UAV body coordinate frame. Matrix T indicates the homogeneous transformation between the two coordinate frames, which is composed of the rotation matrix R and the translation vector $\mathbf{t}$:

$$T = \begin{bmatrix} R & \mathbf{t} \\ 0_{1\times3} & 1 \end{bmatrix} \tag{20}$$

and N_c is the intrinsic matrix of the camera:

$$N_C = \begin{bmatrix} 1/dx & 0 & u_0 \\ 0 & 1/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{21}$$

where f, u_0, v_0, dx, and dy are the camera-intrinsic parameters, which can be obtained in advance through offline calibration.

Due to the nonlinearity of the measurement model $h(\bullet)$, the Jacobian matrix $H(\bullet)$ is:

$$H_k = \frac{\partial h(\bar{\mathbf{x}}_{k|k-1})}{\partial \bar{\mathbf{x}}_{k|k-1}} \tag{22}$$

Hence, the Kalman gain K_k is:

$$S_k = H_k P_{k|k-1} H_k^T + G \tag{23}$$

$$K_k = P_{k|k-1} H_k^T (S_k)^{-1} \tag{24}$$

where G is the sensor's Gaussian noise covariance matrix for each measurement. The final step is to update the state $\mathbf{x}$:

$$\bar{\mathbf{x}}_{k|k} = \bar{\mathbf{x}}_{k|k-1} + K_k(\mathbf{z}_k - h(\bar{\mathbf{x}}_{k|k-1})) \tag{25}$$

5. Experiments

To validate the proposed algorithm and generate a training dataset, we built a parallel system as shown in Figure 3. This system comprised the guidance system and the fixed-wing UAV. The guidance system included two 2-freedom pan-tilt units (PTUs), two cameras mounted on the PTUs, and a laptop with i9-9900k (CPU) and NVIDIA GTX2080 (GPU)). In the outdoor experiment, the PTUs were placed on both sides of the runway with a 10.77 m baseline. The PTU attitude measurement resolution reached 0.00625 degrees, and its highest rotary speed was 50 degree/s. The camera DFK 23G445 (Germany) turned along with the PTU to extend the field of view and generated 640 × 480 pixel video with 60 frames per second. The fixed-wing UAV Pioneer had a wingspan of 2.3 m and a total mass of 14 kg with petrolic propulsion. The simulation followed the same configuration as the outdoor environment, including the UAV model, PTU, and camera parameters. In addition, the real UAV autopilot PX4 was also introduced to establish a hardware-in-loop simulated system and to realize a more realistic flight.

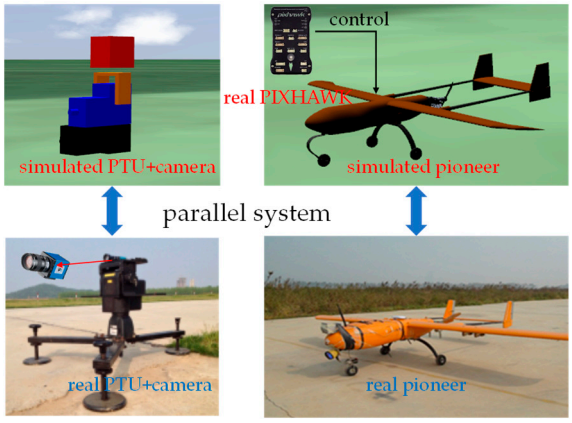

Figure 3. The parallel system contains the outdoor and Gazebo-based simulated environments. It is composed of the guidance system and fixed-wing UAV, and the outdoor and simulated environments have the same configuration.

Online experiments were conducted both in simulated and outdoor environments to validate the performance. A complete landing comprises sloping, flaring, and taxiing, and there are different requirements for pose estimation accuracy in different phases. Therefore, the performances of pose estimation were analyzed for all three phases. For autonomous ROI extraction, YOLO-v4 [40] was employed to provide the ROI of the UAVs. The dataset that was used for offline anchor detection and ROI extraction training was generated from 5 simulated (1610 frames) and 2 outdoor (730 frames) landings. Experimental results show that YOLO-v4 achieved 97.2% UAV ROI detection accuracy in our landing scenes. Almost all of the misdetections occurred in the taxiing period since the background on the ground was significantly more complex than the sky.

5.1. Anchor Detection

To validate the anchor detection accuracy through the training data augmentation, three different training datasets were used to train the networks. These datasets included the real dataset (RD), the simulated dataset (SD), and the mixed dataset (MD) combining the RD and SD. We implemented and evaluated two classic anchor detection methods as a baseline for anchor detection experiments: a conventional network (only the F part shown in Figure 2) and KeyPose [41]. KeyPose localizes the anchor by predicting heatmaps. As shown in Table 1, three methods were trained using the above three datasets, and nine networks with different parameters were obtained accordingly. The anchor detection error e is defined as:

$$e = \frac{\|(u, v) - (\widetilde{u}, \widetilde{v})\|}{w} \tag{26}$$

where (u, v) and $(\widetilde{u}, \widetilde{v})$ are the detection and ground truth of the anchor locations in the image frames, respectively. As shown in Figure 4, e indicates the pixel distance between the detection and ground truth of the anchor, and w is the width of the ROI. For each anchor, a detection with $e > 5\%$ is regarded as a failure. For a complete test, assuming that the number of the total anchors is M and the failure number is m, the failure rate f is:

$$f = \frac{m}{M} \tag{27}$$

Table 1. The failure rates of anchor detection (defined as Formula (27)) using 3 different training datasets.

Test Data	Networks / Training Datasets	Conventional Network SD	Conventional Network RD	Conventional Network MD	KeyPose SD	KeyPose RD	KeyPose MD	Block-CNN SD	Block-CNN RD	Block-CNN MD
Simulation 1000 frames	*LW* (%)	2.0		1.7	1.8		1.0	0.0		0.5
	LT (%)	5.6		5.1	4.3		3.7	0.9		0.2
	FT (%)	2.3		0.2	1.4		0.6	0.0		0.2
	RT (%)	4.9		2.2	5.9		3.0	0.5		0.2
	RW (%)	5.2		1.2	2.0		1.0	1.1		0.2
	Average	4.0		**2.1**	3.1		**1.9**	0.5		**0.3**
Outdoor 1000 frames	*LW* (%)		7.0	7.4		4.5	4.0		2.5	1.9
	LT (%)		6.6	6.1		6.2	6.0		2.2	2.0
	FT (%)		5.2	5.2		3.9	4.0		2.2	1.5
	RT (%)		9.1	6.9		6.7	7.5		4.4	2.1
	RW (%)		8.0	7.7		6.0	5.8		3.3	3.9
	Average		7.2	**6.7**		5.5	**5.5**		2.9	**2.3**

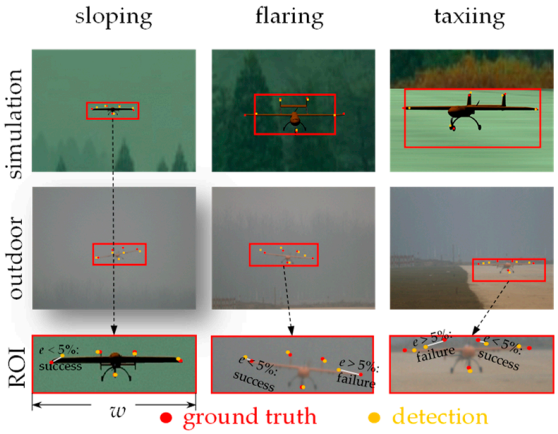

Figure 4. Detection samples in different landing phases using network *F*. Once the pixel distance between the detection and ground truth of the anchor **exceeds** $5\% \times w$ (ROI width), this detection is regarded as a failure.

Table 1 presents the detection failure rates of the five UAV anchors. Compared with the networks trained by only the real or simulated dataset, the networks trained by the mixed dataset improved detection accuracy for the conventional network (from 4.0% to 2.1% and 7.2% to 6.7%), KeyPose (from 3.1% to 1.9% and 5.5% to 5.5%), and the Block-CNN (from 0.5% to 0.3% and 2.9% to 2.3%). Using the training dataset MD, the Block-CNN realized detection precision rates of 99.7% and 97.7% for the simulated and outdoor tests. In addition, under the premise of using the same training dataset, the Block-CNN achieved 3.5% and 2.5% precision improvement compared with the two benchmarks, respectively.

Figure 4 also displays several detection samples in different landing phases using network F. According to the images captured in the simulated and outdoor environments, the UAV showed the more distinguishable features in the simulated cases. This indicates

that the dataset acquired from the simulation was more efficient than that of the outdoor environment. This is because the experimental conditions in the simulated environment were more controllable. Therefore, generating simulated data not only expanded the dataset but also improved the dataset quality, hence contributing to the improvement of the anchor detection accuracy.

In summary, the experimental results suggest that training dataset augmentation contributes to anchor detection improvement. Furthermore, compared with the two baselines, the proposed Block-CNN achieves significant improvement in the accuracy of anchor detection in UAV landing scenes.

5.2. Pose Estimation

5.2.1. Simulations

Here, we evaluate the pose estimation performance of the proposed algorithm and make comparisons with the PNP solution in the simulated environment. Three methods were tested on the simulated guidance system including the monocular PNP (MP), monocular EKF (ME), and stereo EKF (SE) proposed in this paper. Since MP and ME only need monocular vision, we considered the average results of the two cameras as their final results.

We present the three simulated landing scenes (S_1, S_2, S_3) that were designed to validate the performance of the proposed algorithm. For each landing scene, three different landing trajectories were designed, which means that a total of 9 simulated experiments were conducted and discussed. The first landing was completed without wind disturbance. To simulate the crosswind in the outdoor environment, a continuous crosswind was created during the second and third landings. Additionally, in the third landing, a going-around process was also simulated, which was common during the actual landing. The estimated pose error in sloping, flaring, and taxiing phases is shown in Figure 5. Each group of error graphs represents the average error of three flight experiments in the same scene. Since the UAV gradually approached the guidance system, the position estimation error of the ME and MP was gradually reduced from the sloping to the taxiing phase. In other words, the ME and MP are sensitive to the distance between the UAV and the camera. On the contrary, the distance almost does not affect the positioning error of the proposed SE; a remarkable positioning error of the MP exceeding 20 m (S_2 at the *X*- and *Y*-axis) which is likely to result in the deviation from the runway. In the flaring period, the estimation of the height above the ground (*Z*-axis) is also important. Except for the SE, the other methods had significant errors along the *Z*-axis.

The attitude estimation error showed greater volatility than that of the positioning error. The biggest pitch estimation error reached 10° in the sloping phase of S_2. It probably caused the UAV to descend too fast to successfully land. A high initial yaw estimation error was also seen in S_2. However, the errors of SE and ME gradually converged before entering the flaring period, whereas the error of MP was deemed volatile.

Figure 6 displays the RMSE for the three landing scenes (each scene contains three flights with different trajectories). For the conventional method MP, the positioning RMSE reached 18.41 m at the *X*-axis and 3.67 m at the *Z*-axis, showing an unacceptable level of accuracy for the landing process. In addition, the yaw RMSEs exceeding 6 degrees in the simulations S_2 and S_3, cannot support a successful landing. By comparison, ME achieved accuracy improvement for both the position and attitude estimation. Furthermore, a significantly remarkable pose RMSE reduction was achieved by the SE. The RMSEs at the X- and *Y*-axes did not exceed 1.0 m, which ensured that the UAV was completely within the runway range. More importantly, the RMSE at the *Z*-axis did not exceed 0.1 m. It also laid a key foundation for a successful landing.

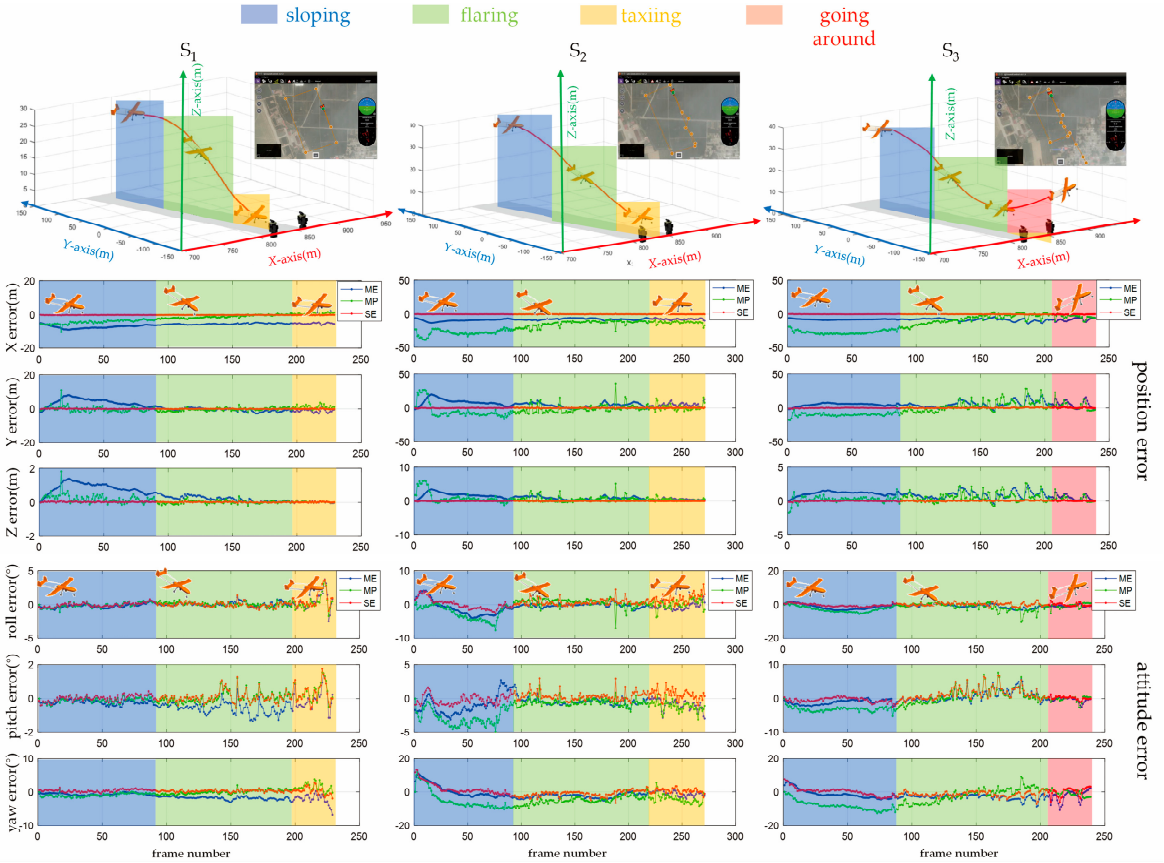

Figure 5. The landing trajectories in the simulated environment and the pose estimation error for the three simulated landing scenes. Each landing scene contains three different landing trajectories for simulated validation.

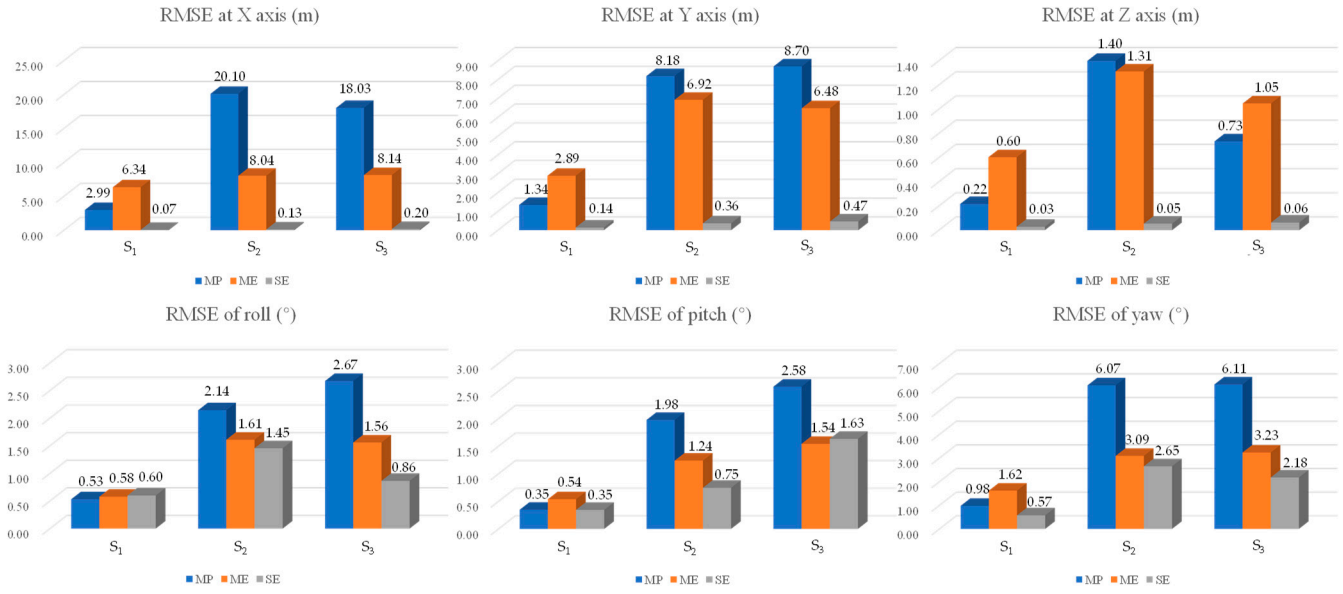

Figure 6. The RMSE of pose estimation for the three simulated experiments.

In addition to the position estimation, the estimation of yaw performed with lower accuracy compared with roll and pitch, but the yaw RMSE of SE still did not exceed 2.01°. In terms of the real-time capability, the final pose estimation fps was almost the same as the anchor detection module (>30 fps), since the EKF estimator consumed much less time

than the networks. The anchors' configuration is critical for improving pose estimation accuracy, such as their spatial distribution and number. Generally speaking, the more detected anchors, the more information the anchors provide, and the lower the sensitivity to the anchor detection error. Therefore, the maximum possible number of anchors should be configured. In our algorithm, 5 anchors were configured for the UAV.

To explore the influence of the anchors' configuration on position and attitude estimation accuracy, another two configurations were employed to compare the pose estimation accuracy. As shown in Figure 7, 3 anchors and 8 anchors were configured and the results in simulations S_2 and S_3 are listed in Table 2. According to the RMSE of the position and attitude, it is obvious that the configuration with 3 anchors performed with a drastically lower accuracy in both position and attitude estimation compared with the case with 5 anchors. It is also impossible to realize a safe auto-landing with the position RMSE of 5.03 m and 3.38 m at the *Z*-axis. In other words, the 3-anchor configuration is infeasible. For the 8-anchor configuration, the new 3-anchor addition did not introduce a remarkable enhancement to the estimation accuracy. In summary, for the UAV landing application, the experimental results indicate the superiority of the 5-anchor configuration employed in our algorithm.

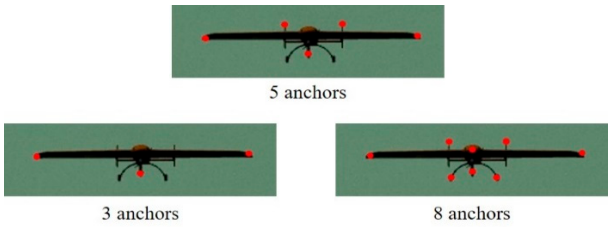

Figure 7. Another two different anchor configurations. Five-anchor configurations are employed in our algorithm.

Table 2. The RMSE in the SE for three-, five-, and eight-anchor cases.

Index	Anchor Number	Position RMSE (m)			Attitude RMSE (°)		
		X	Y	Z	Roll	Pitch	Yaw
S_2	3	3.54	16.99	5.03	2.16	15.58	2.11
	5	**0.14**	0.64	**0.09**	**0.84**	**0.79**	**2.01**
	8	0.27	**0.53**	0.10	1.21	0.79	2.23
S_3	3	5.83	20.15	3.38	1.92	18.62	2.66
	5	**0.19**	**0.71**	0.08	**0.69**	1.57	**1.64**
	8	0.22	0.73	**0.05**	0.82	**1.37**	2.00

5.2.2. Outdoor Evaluation

We implemented and evaluated KeyPose combined with classical triangulation (KC) [41] and object triangulation (KO) [41], respectively, as the baseline of the 6D pose estimation. The ground truth of the UAV pose was obtained by the onboard synchronous D-GPS and inertial navigation system (INS).

For auto-landing, the estimation accuracy at the *Z*-axis is the most important in terms of position estimation, especially in the flaring period. Once the UAV is taxied on the runway, which is usually detected by the landing detector, the UAV controller would not consider the location at the *Z*-axis. In our outdoor experiments, the runway width was about 10 m. This means that the maximum error at the *X*-axis in the flaring and taxiing phases should not exceed 5 m.

Three outdoor experiments were conducted for performance evaluation. Since the computer platform on which the algorithms run was the same as the one used in simulation, the real-time capability in simulation (30 fps) was also valid for outdoor evaluation. Figure 8 illustrates the position and attitude estimation results of one of the three outdoor experiments and the pose estimation error. For a successful landing, smooth temporal

positioning is a prerequisite. According to the position estimation results, the error curves of KC and KO showed more fluctuations compared with the proposed method SE. For instance, the maximum errors of KC and KO even reached 132 m at the *Y*-axis and 14 m at the *X*-axis. This can easily deviate the UAV from the 10 m width runway. On the contrary, the maximum error of SE at the *X*-axis did not exceed 5 m, which was necessary to ensure that the aircraft was always within the range of the runway in flaring and taxiing phases. For the algorithms KC and KO, there was a higher error at the *Y*-axis than that of the other axes. This is because the *Y*-axis has a high degree of coincidence with the direction of the cameras' optical axis. This makes the positioning at the *Y*-axis more sensitive to measurement errors. The details have been discussed in our previous work [6]. For the attitude estimation, the results of the three methods showed comparable temporal fluctuation. The images and detected anchors from the two cameras are also shown at points A, B, and C. The red and yellow anchors are ground truth and detection results, respectively. It is seen that the anchor *RW* was out of the field of view in the left camera at point C. This caused a remarkable *Z*-axis positioning error.

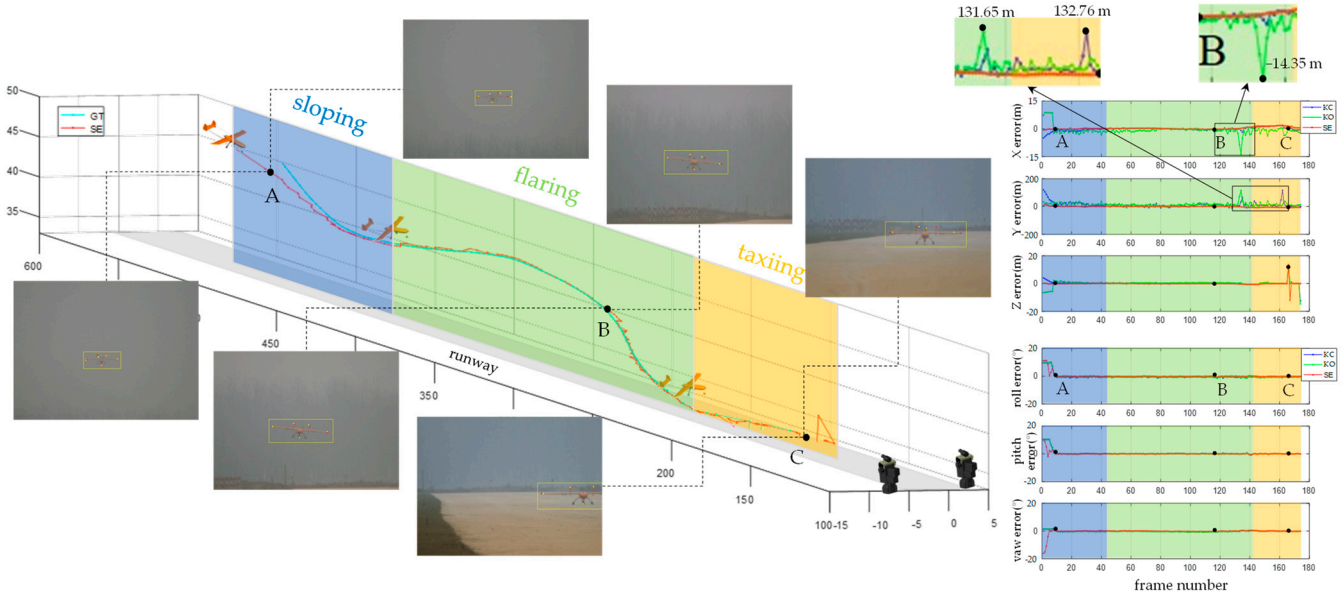

Figure 8. The trajectory estimation results of one of the outdoor experiments and the pose estimation error by KC, KO, and SE. Estimated trajectories in sloping, flaring, and taxiing are depicted. In addition, several exemplary images captured from cameras during auto-landing and the anchor-detection results are also shown.

Table 3 shows the RMSE of the three outdoor experiments in the sloping, flaring, and taxiing, respectively. The RMSE in the sloping phase is higher than that of the other two landing phases. This is because the UAV was the farthest away from the cameras in the sloping phase, and our previous work [6] has demonstrated that for a given detection error, the further the UAV is from the cameras, the greater the location error. In addition, the roll and yaw RMSEs of SE in the sloping phase reached 21.0°, 19.3°, and 17.8° and 11.5°, 10.3°, and 11.8°, respectively, which were even higher than that of the triangulation solution (16.2°, 18.5°, and 19.9° and 7.8°, 7.6°, and 8.5°). The reason is that the initial UAV pose of the proposed method SE was roughly estimated, and it needed to take several steps to converge. The RMSEs of the height estimation (at the *Z*-axis) in the flaring period by the triangulation solution reached 1.1 m, 1.4 m, and 1.6 m, thus leading to a failed landing. In contrast, the SE results did not exceed 0.5 m, thus satisfying the safe landing requirement.

Table 3. The RMSE in outdoor performance evaluations.

RMSE	Solution	Sloping			Flaring			Taxiing		
		Exp 1	Exp 2	Exp 3	Exp 1	Exp 2	Exp 3	Exp 1	Exp 2	Exp 3
***X* axis (m)**	SE	**3.2**	**2.8**	**3.8**	**0.3**	**0.5**	**0.6**	**1.9**	**1.9**	2.8
	KC	4.6	3.9	4.1	2.5	1.5	1.0	3.1	3.1	**2.7**
	KO	3.8	3.8	3.8	1.0	1.7	2.1	2.3	3.8	3.4
Y axis (m)	SE	**11.9**	**10.8**	**12.9**	**3.7**	**4.9**	**2.0**	**17.8**	**12.4**	**10.0**
	KC	23.4	24.6	18.5	22.9	19.4	15.8	34.9	30.6	20.1
	KO	24.1	29.2	21.0	20.1	22.5	17.6	30.4	30.8	26.4
Z axis (m)	SE	5.4	2.9	**1.8**	**0.4**	**0.3**	**0.5**	7.5	**5.9**	**5.4**
	KC	**3.4**	3.0	2.0	1.2	1.5	1.4	6.4	8.0	7.6
	KO	**3.4**	**2.8**	2.0	1.0	1.3	1.8	**6.2**	7.3	8.7
Roll (°)	SE	21.0	19.3	**17.8**	**0.7**	**1.5**	**0.9**	**0.8**	**1.5**	**1.1**
	KC	16.3	**18.4**	19.4	4.5	2.7	2.8	2.9	2.2	1.9
	KO	**16.1**	18.5	20.4	3.9	2.2	2.9	2.9	2.2	1.8
Pitch (°)	SE	**19.5**	**22.8**	**23.9**	**0.5**	**0.8**	1.0	**1.1**	**0.9**	**1.4**
	KC	21.5	23.5	23.9	1.2	1.4	1.8	1.6	2.5	3.1
	KO	22.3	23.4	31.0	0.9	1.9	**0.9**	1.3	2.2	2.4
Yaw (°)	SE	11.5	10.3	11.8	**1.8**	**2.2**	**1.9**	2.2	**1.4**	**2.0**
	KC	8.2	7.9	**8.2**	3.1	2.8	2.4	**1.8**	3.1	2.9
	KO	**7.4**	**7.2**	8.7	2.8	2.5	2.5	**1.8**	2.2	3.5

In summary, the outdoor experimental results show that, compared with the two benchmarks KC and KO, the proposed SE achieved 19.1% and 19.9% (average 19.5%) accuracy improvement in position, and 12.3% and 13.0% (average 12.7%) accuracy improvement in attitude, respectively. The outdoor experiments also confirmed the feasibility of the proposed solution SE to provide the real-time poses of the UAV during autonomous landing.

6. Conclusions

We have introduced a UAV pose joint estimation algorithm for autonomous guidance, enabled by an N-cameras ($N \geq 1$) ground vision system. This approach involves constructing a pipeline that combines CNN-based anchor detection and anchors-driven pose estimation. To enhance anchor detection, we have introduced a Block-CNN learning-based detection algorithm, leveraging a blocking mechanism that significantly improves accuracy and robustness. Given the high cost associated with outdoor experiments, we have devised a parallel system that includes both simulated and outdoor environments, sharing the same configuration, to augment the training dataset via simulated experiments. The actual pose estimation is achieved through an EKF estimator that uses the detected anchor locations. Our simulation and experiments show that our method achieves 3.0% anchor detection precision improvement and 19.5% and 12.7% accuracy improvement of position and attitude estimation, compared with other state-of-the-art methods. Furthermore, these experiments affirm that our algorithm satisfies the stringent landing navigation requirements in terms of accuracy, real-time capability, and robustness, an essential prerequisite for continuous UAV pose estimation is maintaining the UAV within the field of view of the ground camera. Hence, accurate and stable servo tracking of pan-tilt units is crucial for successful auto-landing. However, in our outdoor experiments, we occasionally encountered issues with failed servo tracking, leading to experimental failures. To enhance the stability of our ground vision system, we will focus on addressing the servo tracking problem in our future work. Additionally, our future endeavors will encompass exploring special scenarios, including varying weather conditions and complex backgrounds.

We conducted a series of simulated and outdoor experiments, and the results show that, compared with the baselines, our method achieves 3.0% anchor detection precision improvement and 19.5% and 12.7% accuracy improvement of position and attitude estimation.

Author Contributions: Conceptualization, D.T. and L.S.; methodology, D.T.; software, J.L.; validation, D.T. and H.Z.; data curation, D.T. and J.L.; writing—original draft preparation, D.T.; writing—review and editing, X.X.; supervision, L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The auto-landing dataset built in this study is available in the repository: https://pan.baidu.com/s/16XKMmL46M1UBGSFNxXNPFQ?pwd=h039 (accessed on 13 November 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Quan, Q.; Fu, R.; Li, M.; Wei, D.; Gao, Y.; Cai, K.-Y. Practical Distributed Control for VTOL UAVs to Pass a Virtual Tube. *IEEE Trans. Intell. Veh.* **2021**, *7*, 342–353. [CrossRef]
2. Shao, S.; Chen, M.; Hou, J.; Zhao, Q. Event-Triggered-Based Discrete-Time Neural Control for a Quadrotor UAV Using Disturbance Observer. *IEEE/ASME Trans. Mechatron.* **2021**, *26*, 689–699. [CrossRef]
3. Souli, N.; Kolios, P.; Ellinas, G. Online Relative Positioning of Autonomous Vehicles using Signals of Opportunity. *IEEE Trans. Intell. Veh.* **2021**, *7*, 873–885. [CrossRef]
4. Henawy, J.; Li, Z.; Yau, W.-Y.; Seet, G. Accurate IMU Factor Using Switched Linear Systems for VIO. *IEEE Trans. Ind. Electron.* **2021**, *68*, 7199–7208. [CrossRef]
5. Herissé, B.; Hamel, T.; Mahony, R.; Russotto, F.-X. Landing a VTOL Unmanned Aerial Vehicle on a Moving Platform Using Optical Flow. *IEEE Trans. Robot.* **2012**, *28*, 77–89. [CrossRef]
6. Tang, D.; Hu, T.; Shen, L.; Zhang, D.; Kong, W.; Low, K.H. Ground Stereo Vision-Based Navigation for Autonomous Take-Off and Landing of UAVs: A Chan-Vese Model Approach. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 67. [CrossRef]
7. Yang, B.; Yang, E. A Survey on Radio Frequency based Precise Localisation Technology for UAV in GPS-denied Environment. *J. Intell. Robot. Syst.* **2021**, *103*, 38. [CrossRef]
8. Kim, J.; Woo, S.; Kim, J. Lidar-guided Autonomous Landing of an Aerial Vehicle on a Ground Vehicle. In Proceedings of the 14th International Conference on Ubiquitous Robots and Ambient Intelligence, Jeju, Republic of Korea, 28 June–1 July 2017; pp. 228–231.
9. Pavlenko, T.; Schütz, M.; Vossiek, M.; Walter, T.; Montenegro, S. Wireless Local Positioning System for Controlled UAV Landing in GNSS-Denied Environmen. In Proceedings of the 5th International Workshop on Metrology for AeroSpace, Rome, Italy, 20–22 June 2019; pp. 171–175.
10. Zhang, C.; Chen, L.; Yuan, S. ST-VIO: Visual Inertial Odometry Combined with Image Segmentation and Tracking. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8562–8570. [CrossRef]
11. Garforth, J.; Webb, B. Visual Appearance Analysis of Forest Scenes for Monocular SLAM. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 1794–1800.
12. Ho, H.; De Wagter, C.; Remes, B.; de Croon, G. Optical-Flow Based Self-Supervised Learning of Obstacle Appearance Applied to MAV Landing. *Robot. Auton. Syst.* **2018**, *100*, 78–94. [CrossRef]
13. Yuan, H.; Xiao, C.; Xiu, S.; Zhan, W.; Ye, Z.; Zhang, F.; Zhou, C.; Wen, Y.; Li, Q. A Hierarchical Vision-Based UAV Localization for an Open Landing. *Electronics* **2018**, *7*, 68. [CrossRef]
14. Wang, L.; Bai, X. Quadrotor Autonomous Approaching and Landing on a Vessel Deck. *J. Intell. Robot. Syst.* **2018**, *92*, 125–143. [CrossRef]
15. Baca, T.; Stepan, P.; Spurny, V.; Hert, D.; Penicka, R.; Saska, M.; Thomas, J.; Loianno, G.; Kumar, V. Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle. *J. Field Robot.* **2019**, *36*, 874–891. [CrossRef]
16. Lim, J.; Lee, T.; Pyo, S.; Lee, J.; Kim, J.; Lee, J. Hemispherical InfraRed (IR) Marker for Reliable Detection for Autonomous Landing on a Moving Ground Vehicle from Various Altitude Angles. *IEEE/ASME Trans. Mechatron.* **2021**, *27*, 485–492. [CrossRef]
17. Hu, B.; Mishra, S. Time-Optimal Trajectory Generation for Landing a Quadrotor Onto a Moving Platform. *IEEE/ASME Trans. Mechatron.* **2019**, *24*, 585–596. [CrossRef]
18. Lungu, M. Auto-Landing of Fixed Wing Unmanned Aerial Vehicles Using the Backstepping Control. *ISA Trans.* **2019**, *95*, 194–210. [CrossRef]
19. Abu-Jbara, K.; Sundaramorthi, G.; Claudel, C. Fusing Vision and Inertial Sensors for Robust Runway Detection and Tracking. *J. Guid. Control Dyn.* **2018**, *41*, 1929–1946. [CrossRef]
20. Meng, Y.; Wang, W.; Han, H.; Zhang, M. A Vision/Radar/INS Integrated Guidance Method for Shipboard Landing. *IEEE Trans. Ind. Electron.* **2019**, *66*, 8803–8810. [CrossRef]
21. Gui, Y. Research on Key Techniques of Airborne Vision-Based Navigation for Autonomous Landing of A UAV on A Ship Deck. Ph.D. Thesis, Aeronautical and Astronautical Science and Technology Graduate School of National University of Defense Technology, Changsha, China, 2013.

22. Santos, N.P.; Lobo, V.; Bernardino, A. Autoland Project: Fixed-Wing UAV Landing on a Fast Patrol Boat Using Computer Vision. In *OCEANS 2019 MTS/IEEE SEATTLE*; IEEE: Piscataway, NJ, USA, 2019.
23. Ma, Z.; Hu, T.; Shen, L. Stereo Vision Guiding for the Autonomous Landing of Fixed-Wing UAVs: A Saliency-Inspired Approach. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 43. [CrossRef]
24. Li, M.; Hu, T. Deep Learning Enabled Localization for UAV Autolanding. *Chin. J. Aeronaut.* **2021**, *34*, 585–600. [CrossRef]
25. Kong, W.; Zhou, D.; Zhang, Y.; Zhang, D.; Wang, X.; Zhao, B.; Yan, C.; Shen, L.; Zhang, J. A Ground-Based Optical System for Autonomous Landing of a Fixed Wing UAV. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), Chicago, IL, USA, 14–18 September 2014; pp. 4797–4804.
26. Nguyen, T.H.; Cao, M.; Qiu, Z.; Xie, L. Integrated UWB-Vision Approach for Autonomous Docking of UAVs in GPS-denied Environments. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 9603–9609.
27. Dong, X.; Gao, Y.; Guo, J.; Zuo, S.; Xiang, J.; Li, D.; Tu, Z. An Integrated UWB-IMU-Vision Framework for Autonomous Approaching and Landing of UAVs. *Aerospace* **2022**, *9*, 797. [CrossRef]
28. Liu, F.; Wei, Z.; Zhang, G. An Off-Board Vision System for Relative Attitude Measurement of Aircraft. *IEEE Trans. Ind. Electron.* **2021**, *69*, 4225–4233. [CrossRef]
29. Yang, J.; Du, T.; Liu, X.; Niu, B.; Guo, L. Method and Implementation of a Bioinspired Polarization-Based Attitude and Heading Reference System by Integration of Polarization Compass and Inertial Sensors. *IEEE Trans. Ind. Electron.* **2020**, *67*, 9802–9812. [CrossRef]
30. Li, K.; Chang, L.; Chen, Y. Common Frame Based Unscented Quaternion Estimator for Inertial-Integrated Navigation. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 2413–2423. [CrossRef]
31. Mammarella, M.; Campa, G.; Napolitano, M.R.; Fravolini, M.L.; Gu, Y.; Perhinschi, M.G. Machine Vision/GPS Integration Using EKF for the UAV Aerial Refueling Problem. *IEEE Trans. Syst. Man Cybern.* **2008**, *38*, 791–801. [CrossRef]
32. Fu, Q.; Quan, Q.; Cai, K.-Y. Robust Pose Estimation for Multirotor UAVs Using Off-Board Monocular Vision. *IEEE Trans. Ind. Electron.* **2017**, *64*, 7942–7951. [CrossRef]
33. Diogo, L.; David, P.; Hedi, T. Multi-Task Deep Learning for Real-Time 3D Human Pose Estimation and Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 2752–2764.
34. Valle, R.; Buenaposada, J.; Baumela, L. Multi-Task Head Pose Estimation in-the-Wild. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 2874–2881. [CrossRef] [PubMed]
35. Lin, W.; Anwar, A.; Li, Z.; Tong, M.; Qiu, J.; Gao, H. Recognition and Pose Estimation of Auto Parts for an Autonomous Spray Painting Robot. *IEEE Trans. Ind. Inform.* **2019**, *15*, 1709–1719. [CrossRef]
36. Wu, H.; Zhou, J. Privacy Leakage of SIFT Features via Deep Generative Model Based Image Reconstruction. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 2973–2985. [CrossRef]
37. Campos, C.; Elvira, R.; Rodriguez, J.; Montiel, J.; Tardos, J. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
38. Nanni, L.; Ghidoni, S.; Brahnam, S. Handcrafted vs. Non-Handcrafted Features for Computer Vision Classification. *Pattern Recognit.* **2017**, *71*, 158–172. [CrossRef]
39. Tang, D.; Fang, Q.; Shen, L.; Hu, T. Onboard Detection-Tracking-Localization. *IEEE/ASME Trans. Mechatron.* **2020**, *25*, 1555–1565. [CrossRef]
40. Alexey, B.; Chien-Yao, W.; Hong-Yuan, M. Yolov4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
41. Liu, X.; Jonschkowski, R.; Angelova, A.; Konolige, K. Keypose: Multi-View 3D Labeling and Keypoint Estimation for Transparent Objects. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11599–11607.