Abstract: Nowadays, quadcopters are used widely in different applications, but their flight time is limited during operation. In this paper, a precision landing method based on a Kalman filter is proposed for an autonomous indoor persistent drone system that aims to increase the flight time of quadcopters. First, a local positioning system is used for tracking performance. Second, instead of using this local positioning system during the landing phase, a multi-ranger sensor is proposed to increase the accuracy of horizontal errors. Next, based on the relative position provided by the multi-ranger sensor, a Kalman filter technique is applied to estimate the relative velocity of the system, which is then applied to control the position of the quadcopter during the landing phase. Finally, a charging state machine law is proposed to charge the battery of three quadcopters sequentially.

Keywords: wireless charging; quadcopter; unmanned aerial vehicle (UAV); precision landing

1. Introduction

Quadcopters are unmanned aerial vehicles (UAVs) that have been studied and utilized in a variety of applications owing to their advantages, such as simple structure, small size, low cost, and agility. Consequently, quadcopters are more popular than other types of UAV systems. Therefore, they have been tested and commercialized for several applications, including formation flight [1,2], transportation [3,4], fault-tolerant control [5,6], precision landing [7–10], and search and rescue [11,12]. Normally, a quadcopter is lifted by four propellers through navigation and flight control systems to operate hovering and maneuverability for both indoor and outdoor tasks. However, one of the main issues of quadcopters is their short operation time. This is because quadcopters frequently generate their total force from the propellers to achieve the desired positions, and this requires high power consumption. Therefore, there is an essential need to charge the battery of UAVs autonomously for long-time operation. One of the main problems in autonomous drone-charging systems is to accomplish a precise landing phase. This task would be useful for coastal surveillance that requires the quadcopter UAVs to return to a home position for battery charging. Moreover, marine applications could be used to transport quadcopter UAVs to desired locations and to allow them to charge periodically after finishing their operations. The main issue of this task is the limited accuracy of navigation systems. Due to the drawback of low-cost navigation systems, the research topic of landing approach of the quadcopter on a stationary target is more and more crucial for many precise applications.
Several studies focused on the development of new strategies to address autonomous recharging [13,14] with the aim of increasing the flight time and endurance during missions. In [15], researchers presented a mobile vehicle that had the capability of recharging automatically on a static station and then resuming its original mission. In [16], a persistent mission based on a contact charging method was studied for UAVs. In [17], a drone-charging station was implemented using RTK-GPS, which is a more accurate positioning system than conventional GPS. An autonomous landing scheme based on an infrared beam was proposed in [18]. However, most of the above-mentioned methods are difficult to incorporate in practical solutions because of the precise positioning system and mechanical design required.

Another solution using vision techniques was recently investigated to increase the accuracy of drone charging. In [19–21], vision-based landing techniques with robust controllers were proposed to land quadcopters in a precise manner; these controllers include a model-predictive controller, neural network controller, and backstepping controller. In [22], a landing task was proposed for an aerial vehicle using optical flow and adaptive control law. Another development of the landing method was proposed in [23] based on the orientation estimation from a camera. These studies elaborated on landing on a charging pad, but they require high computation for filtering data from vision sensor. Compared with contract-based charging, wireless charging was investigated in many low-power applications, such as smartphones, cleaning robots, mobile robots, and other electronic devices. The principle of wireless operation based on magnetic resonant induction [24–27] was applied to drone-charging systems. The goal of wireless communication is to develop a battery charging system for UAVs with a considerable improvement in reliability [28]. In [29], a drone was proposed to land on a ground vehicle for charging through such ground vehicle and a robot arm. In [30], a persistent charging strategy was proposed for drones via wireless communication. Most of the aforementioned studies demonstrated significant results through simulations or experiments, but they did not present a persistent charging scheme for a group of quadcopters. This was the motivation of this study.

This paper aims to improve the precision landing method for autonomous persistent drone-charging application without human intervention. This application means that when a vehicle is low in battery in ocean surveillance, then another vehicle can replace this vehicle to charge sequentially. In contrast to previous studies, this study proposes a low-cost landing method using a multi-ranger sensor that can measure the relative position between the drone and objects to improve horizontal errors. The advantage of this method is that it can operate in both bright and dark environments. It can also reduce the noise as opposed to techniques based on vision sensors. The contributions of this study can be summarized as follows:

- Modifies the open-source software of Crazyflie drones, and a Kalman filter is applied to estimate the velocity from the relative position provided by using a multi-ranger sensor.
- A sequential charging strategy is proposed for autonomous persistent drone-charging of three quadcopters.
- Different to vision-based control techniques in previous studies in that a single drone can land precisely; this work can allow drones to land accurately with persistent feature.

2. Methodology
2.1. UAV Platform

In this study, a Crazyflie 2.1 (as shown in Figure 1), developed by Bitcraze [31], was used for experimentation. The Crazyflie 2.1 is a nano-quadcopter that weighs 27 g and measures $92 \times 92 \times 92$ mm$^3$. It communicates with the ground station through the CrazyRadio PA [32], which is a 2.4 GHz radio USB dongle. The microcontroller for the main application of Crazyflie 2.1 is the STM32F405 with the following specifications: Cortex-M4 core, 168 MHz, 192 kB SRAM, and 1 Mb flash. The inertial measurement unit (IMU) consists
of a 3-axis gyroscope, a 3-axis accelerometer, and a high-precision pressure sensor. Bitcraze provides several sensors through a so-called deck compatible with Crazyflie 2.1. In this study, a distance sensor, UWB indoor positioning system (IPS), and wireless charging were utilized. The sensors are comprehensively described in Section 3.1.

Figure 1. Crazyflie 2.1.

2.2. Charging State Machine Strategy and Control Scheme

The experiments were conducted using three Crazyflies to verify the persistent and continuous charging system. In these experiments, it was assumed that a Crazyflie needs to hover at a specific point to perform a mission. The ground station must receive the status of the vehicles through permanent communication. The battery capacity of the Crazyflie exceeds the allowed capacity. First, the Crazyflie took off from the landing pad (state 3) and reached the mission/target point; it then hovered for a preset time of 20 s (state 7). Note that the battery capacity of the Crazyflie should be larger than the allowed capacity during the hovering condition. When the Crazyflie satisfied either the hovering time or battery capacity, it was controlled to move a short distance in position mode and wait for the next coming vehicle. When the second Crazyflie satisfied the battery condition, it could take off and reached the target point where the first Crazyflie hovered. While the second Crazyflie moved to the hovering point, the first Crazyflie was controlled to return (state 8), adjust the home position (state 9), and land (state 10) at its home position/landing pad. The retry mission was required if there was a fault during the landing phase; thus, this mission was used to adjust the landing position. The scenario was the same for the third Crazyflie. By repeating the above sequential process, the mission could be continuously performed without human intervention. The whole charging state machine strategy is shown in Figure 2.

The defined parameters are shown in Figure 3, while the control structure is presented in Figure 4. The loco positioning system \((\text{Pos}_d, \text{Pos}_{x,y}, \text{Vel}_{x,y})\) was used in this study because the tests were conducted in an indoor environment. The position controller used this loco positioning system as an absolute coordinate system. In a normal case, the loco positioning system is used to control the position. However, in the landing phase, a relative positioning system \((D_{dx,dy}, D_{x,y}, D_{x,y})\) was applied to reduce the horizontal error. The relative information was obtained by using the wall that was set up. This is described in Section 3.2.
Figure 2. Charging state machine strategy. The defined parameters are shown in Figure 3, while the control structure is presented in Figure 4. The loco positioning system $(\text{Pos}_x, \text{Pos}_y, \text{Vel}_x, \text{Vel}_y)$ was used in this study because the tests were conducted in an indoor environment. The position controller used this loco positioning system as an absolute coordinate system. In a normal case, the loco positioning system is used to control the position. However, in the landing phase, a relative positioning system $(\text{D}_x, \text{D}_y, \text{D}_dx, \text{D}_dy)$ was applied to reduce the horizontal error. The relative information was obtained by using the wall that was set up. This is described in Section 3.2.

Figure 3. Definition of the parameters for proposed system.

It should be noted that the relative position $D_{x,y}$ is measured from a multi-ranger sensor which is discussed in the next section. However, the multi-ranger sensor does not provide the relative velocity $\dot{D}_{x,y}$ for the drone system. Therefore, the value of $\dot{D}_{x,y}$ is estimated from the position of the multi-ranger sensor through Kalman filter algorithm. In actuality, the speed of the indoor positioning system $\text{Vel}_{x,y}$ is estimated in a similar way. Because the $\text{Pos}_{x,y}$ of indoor positioning is insufficient for landing task, the proposed system is used for the pair of data $D_{x,y}$ and $\dot{D}_{x,y}$ (not pair of data $D_{x,y}$ and $\text{Vel}_{x,y}$) to synchronize sensor signal and to improve the accuracy.
3. Experimental Setup

3.1. UAV Platform Description

3.1.1. Sensors

As mentioned in Section 2.1, Bitcraze provides several sensors that are compatible with Crazyflie. The sensors used in this study are as follows:

- Loco positioning system (indoor positioning system).
- Multi-ranger deck (5-way distance sensor).
- Wireless charging deck.

The loco positioning system operates in indoor environments. Several anchors were placed in a specific space. The loco positioning deck is attached to the Crazyflie to calculate the position of Crazyflie by exchanging high-frequency radio messages with the anchors. The user can select a position estimation method among the following three: TWR (two-way ranging), TDoA2 (time differential of arrival 2), and TDoA3 (time differential of arrival 3). TDoA2 was used in this study. When using the multi-ranger deck, a laser-ranging sensor can detect surrounding objects in five directions, namely, front, back, left, right, and up directions, with a maximum distance of 4 m. The wireless charging deck implements the Qi wireless charging standard (WPC v1.2 Qi) [33], which can support charging up to 1 A at 5 V.

3.1.2. System Configuration

Figure 5 shows the overall system configuration. The loco positioning deck, multi-range deck, and wireless charging deck were attached to the Crazyflie. The Crazyflie 2.1 firmware [34] is an open-source firmware for drones provided by Bitcraze. A ground control station (GCS) was set up with a Crazyradio PA to interface the state of the drone. The battery status, logging data, and transmission of commands to each drone were performed by the GCS, while most of the calculations were performed using an onboard flight controller. The landing pad can provide high-speed wireless charging for drones with a consumption of 9 W.
3.1.2. System Configuration

Figure 5 shows the overall system configuration. The loco positioning deck, multi-range deck, and wireless charging deck were attached to the Crazyflie. The Crazyflie 2.1 firmware [34] is an open-source firmware for drones provided by Bitcraze. A ground control station (GCS) was set up with a Crazyradio PA to interface the state of the drone. The battery status, logging data, and transmission of commands to each drone were performed by the GCS, while most of the calculations were performed using an onboard flight controller. The landing pad can provide high-speed wireless charging for drones with a consumption of 9 W.

Figure 6. Wireless charging setup.

3.2. Wireless Charging Setup

Figure 6 shows the wireless charging setup. Three wireless charging stations were installed according to the number of Crazyflies. This is because each drone can return to home position for charging in case of low battery. Normally, when we use the original local position system from the Crazyflie manufacturer, the average accuracy is 10 cm.

In order to overcome this issue, a multi-ranger sensor was used to improve the accuracy of the system. It was placed diagonally so that a multi-ranger sensor could measure the relative distance to the wall, which was used to increase the precision in the landing phase.
The landing pads with a wireless charger were designed according to the charging pad [35] manufactured by Bitcraze. The landing pad was designed with an incline angle so that the landing gear of the drone contacts the wireless charger accurately. It should be noted that if we apply this method for large drones under outdoor environment, then the cost for design of the charging system is expensive. Therefore, in this paper, the landing pad is designed in a small size. Figure 7 shows the CAD part design and the resulting drawing with a three-dimensional printer. The Crazyflie uses a 1-cell LiPo (lithium polymer) battery with a capacity of 240 mAh. When a voltage of 3.5 V is set as the threshold, the charging time is approximately one hour. The maximum operation time is 4 min.

Figure 7. CAD model of landing pad.

3.3. Kalman Filter for Landing Phase

As mentioned in Section 2, the position controller of the landing phase made use of the relative position provided by the multi-ranger sensor instead of the absolute position. The relative velocity was estimated using the Kalman filter algorithm [36]. Equation (1) represents the state variable, and the system model can be expressed as Equation (2):

\[
x = [p \ v]^T
\]

\[x_{k+1} = Ax_k + w_k\]  

\[z_k = Hx_k + v_k\]

where \(p\) is the position, \(v\) is the velocity, \(z_k\) is the measurement, \(w_k\) is the system noise, \(v_k\) is the measurement noise, and \(x_k\) is the state variable. \(A, H, Q, R\) denote the system model matrices and they are defined as:

\[
A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad Q = \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}, \quad R = 10
\]

where \(A\) is the system state matrix, \(H\) is the observation matrix, \(Q\) is the covariance matrix of the system noise, and \(R\) is the covariance matrix of the measurement noise. The Kalman filter algorithm is presented in Figure 8. The first step is used for prediction; it calculates the predicted state \((\hat{x}_k^-)\) and predictive error covariance \((P_k^-)\) from the system matrices \((A, Q)\). The Kalman gain \((K_k)\) is computed from the predictive error covariance \((P_k^-)\) and related system matrices \((H and R)\). In step 3, an estimate \((\hat{x}_k)\) is computed from the measurement input \((z_k)\), Kalman gain, and predicted state. The error covariance is updated in step 4 using the Kalman gain from step 2 and the predictive error covariance from step 1.
It should be noted that increasing value of the matrix $Q$ will increase Kalman gain [34]. This made the estimate closer to the measured value. If the value of matrix $Q$ is decreased, the estimate is less affected by the measurement and moves closer to the previous estimate. This makes the noise in the measurement less influential on the result, and, in turn, the estimate becomes smoother. However, the matrix $R$ affects the results in the opposite way.

4. Results
4.1. Velocity Estimation from Position

As shown in Section 2.2, to use the PID controller in Crazyflie, a feedback velocity should be obtained from the sensor. However, the multi-ranger sensor does not provide the feedback velocity to the quadcopter system. To overcome this issue, the Kalman filter algorithm is applied to estimate the velocity from the position. To validate the proposed method, an optical-flow sensor was used as a reference sensor. In actuality, there is no available sensor to measure the speed attached on the Crazyflie platform. Therefore, the optical-flow sensor which can connect to the Crazyflie platform was chosen for simple purpose. Figure 9 compares the velocity of the optical-flow sensor with the estimation velocity of the Kalman filter algorithm. In actuality, the sinusoidal signal was tested to verify the time-varying response of estimation. Note that the estimation velocity of the Kalman filter algorithm shows a similar trend to that of the optical flow sensor. The user can tune the matrices $Q$ and $R$ to achieve a better performance.
4.2. Trajectory Tracking and Landing Accuracy Performance

In this section, it is assumed that the Crazyflie drones operate under no windy conditions. Because Crazyflie drones are lightweight, they are difficult to land accurately under windy conditions. The further research will consider this limitation with bigger drones. Moreover, before flight test, the rangefinder sensor of each drone can be set up in a way such that it can measure the distance from the wall in x and y directions.

4.2.1. Trajectory Tracking Performance

The trajectory tracking performance of the position controller in x and y directions using the IPS is presented in Figure 10. Figure 10 shows the overall flight performance until just before landing, and shows the setpoint and measured position values. Note that the feedback positions can track the desired positions well. The integral absolute error (IAE) in Equation (4) was used as a performance index that can quantitatively evaluate the control system.

$$ IAE = \int_{t_1}^{t_2} e(t) dt $$  \hspace{1cm} (4)

The IAE refers to the integral area of the difference between the set value and the response. The IAE of each axis is shown in Table 1.

**Table 1.** The IAE of full flight (x axis, y axis).

| Axis  | IAE Value |
|-------|-----------|
| x axis| 2.7930    |
| y axis| 2.2099    |

4.2.2. Landing Accuracy Performance

To show the operation of the proposed algorithm, the performance of the position controller using IPS and the performance of the position controller using a multi-ranger sensor (proposed system) are compared in both x and y directions. In Figures 11–14, state 8 means “come back to the landing pad” and state 9 means “determining the landing point”, as mentioned in Figure 2. Figures 11 and 12 show the landing performance using IPS from $t = 47s$ to $t = 50.2s$. When the state was switched from 8 to 9, the landing process took
place. Note that the accuracy in $x$ and $y$ directions is approximately 10 cm, which was not enough for a successful landing during the flight tests.

Figures 13 and 14 show the landing performance using a multi-ranger sensor from $t = 45s$ to $t = 48s$. When the state was switched from 8 to 9, the landing process took place. Note that the accuracy in $x$ and $y$ directions is approximately 2 cm, which was enough for a successful landing during the flight tests.

![Figure 10](image1.png)

**Figure 10.** (a) Trajectory tracking performance in $x$—direction; (b) trajectory tracking performance in $y$—direction.

![Figure 11](image2.png)

**Figure 11.** Landing performance using IPS in $x$—direction.
Figures 12 and 14 show the landing performance using a multi-ranger sensor from $t_s = 45$ to $t_s = 48$. When the state was switched from 8 to 9, the landing process took place. Note that the accuracy in $x$ and $y$ directions is approximately 2 cm, which was enough for a successful landing during the flight tests.

In order to show the experimental results more visually, they were summarized as landing errors. The error of each direction is defined as:

$$\varepsilon_x = x - c_x$$

$$\varepsilon_y = y - c_y$$

$$\varepsilon_{xy} = -\varepsilon_x + \varepsilon_y$$

where $x$, $y$ are the errors in $x$ and $y$ directions, respectively.

It should be noted that the average error from $x$ or $y$ directions is less than 2 cm. This is better than that of the existing open-source software (10 cm). Figure 15 shows the landing zone of each test. Note that the accuracy is high, and the drone achieved a successful landing during the flight tests.

Note that the accuracies of precision-landing-based fuzzy algorithm in [36] and PD control in [37] are 19.43 cm and 20 cm, respectively, but the proposed method in this article is better, with the accuracy of 2 cm.
The experiment was conducted and repeated three times. This means that the flight test of each drone was repeated three times. Therefore, experimental results for a total of nine landings were obtained. Table 2 shows the landing results of the flight test. The landing results were derived using the IAE. The IAE of each direction is defined as

$$IAE_{x,y} = \int_{t_u}^{t_v} e(t)\,dt$$

(5)

where $IAE_x$ and $IAE_y$ are the IAE values in $x$—direction and $y$—direction. IAE value was calculated from the start of state 9 to the end of state 9.

| Error     | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | Average |
|-----------|------|------|------|------|------|------|------|------|------|---------|
| $IAE_x$   | 0.1603 | 0.0704 | 0.1130 | 0.0972 | 0.1713 | 0.2658 | 0.0454 | 0.8405 | 0.1489 | 0.2125  |
| $IAE_y$   | 0.0492 | 0.1472 | 0.1796 | 0.0564 | 0.1473 | 0.5881 | 0.0833 | 0.4940 | 0.3020 | 0.2275  |

In order to show the experimental results more visually, they were summarized as landing errors. The error of each direction is defined as

$$e_x = |x_0 - x_c|$$

(6)

$$e_y = |y_0 - y_c|$$

(7)

$$e_{xy} = \sqrt{(x_0 - x_c)^2 + (y_0 - y_c)^2}$$

(8)

where $e_x$, $e_y$, $e_{xy}$ are the errors in $x$—direction, $y$—direction, and in both $x$ and $y$ directions.

It should be noted that the average error from $x$ or $y$ directions is less than 2 cm. This is better than that of the existing open-source software (10 cm). Figure 15 shows the landing zone of each test. Note that the accuracy is high, and the drone achieved a successful landing during the flight tests.

Note that the accuracies of precision-landing-based fuzzy algorithm in [36] and PD control in [37] are 19.43 cm and 20 cm, respectively, but the proposed method in this article is better, with the accuracy of 2 cm.

It should be mentioned that the hardware for implementation in this article is lightweight and different to the methods in [36,37]. It is more fair if we can compare with the same open-source structure, but this is time-consuming and is a limitation of this paper. Future work will be implemented on a bigger drone and the same open-source code under outdoor conditions for comparison, which will make the drone more reliable.

**Remark 1.** Different to current studies, the proposed algorithm does not use the vision system. This is low-cost and can help the drone to perform in both bright and dark environments. Moreover, the proposed precision landing technique is the first used for autonomous persistent drone-charging systems.

**Remark 2.** The proposed system using multi-ranger sensor is applied to indoor persistent drone-charging application. The application may fail to achieve if the original hardware is used. Therefore, the proposed system aims to improve the hardware issue using low-cost solutions and aims to introduce new software and application for nano drone systems.

**Remark 3.** The proposed control can be replaced with a robust controller, such as sliding mode control [38], for disturbance rejection. This is a limitation of this paper because the aim of this work is to propose a system which can improve upon the current open-source code from Crazyflie firmware. The future work will focus more on this issue.
Remark 4. In this manuscript, the PID controller is used instead of a nonlinear controller because this is the first modification from open-source firmware. The PID controller was designed by trial-and-error method. In future work, the nonlinear controller will be applied to handle the disturbance and uncertainties.

The complete work can be found in the following link: https://www.youtube.com/watch?v=qoLPiGrLJcc&feature=youtu.be (accessed on 1 August 2022).

Figure 15. Landing results in 2D (landing error).

5. Conclusions

In this study, a precision landing method based on a Kalman filter was investigated for an autonomous persistent drone system. A multi-ranger sensor was used during the landing stage instead of an IPS. The estimation velocity was computed from the relative position of the multi-ranger sensor. A charging state machine strategy is proposed for battery charging in sequence. The proposed method was validated through a flight test on the Crazyflie quadcopter platform. The results show that the estimation velocity can track the optical-flow sensor, which was considered as the reference sensor for comparison. The accuracy was improved significantly for landing performance when compared with existing open-source firmware. Future work should address the health management of batteries for drone systems to increase the reliability of the proposed system. Moreover, the future work will also tackle the outdoor performance under disturbances with bigger drones.

Author Contributions: Conceptualization, N.P.N. and S.K.H.; methodology, N.P.N., N.X.-M., L.N.N.T.H. and B.H.L.; software, N.P.N., N.X.-M., L.N.N.T.H., S.T.L., H.S.J. and B.H.L.; validation, N.P.N., S.T.L., H.S.J. and B.H.L.; formal analysis, N.P.N., B.H.L. and S.K.H.; investigation, N.P.N. and S.K.H.; resources, N.P.N.; data curation, N.P.N. and S.K.H.; writing—original draft preparation,
N.P.N.; writing—review and editing, N.X.-M., N.P.N., S.T.L., H.S.J. and B.H.L.; visualization, N.P.N., S.T.L., H.S.J. and B.H.L.; supervision, S.K.H.; project administration, S.K.H.; funding acquisition, S.K.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2022-2018-0-01423) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). This research also was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2020R1A6A1A03038540).

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

**References**

1. Zhao, W.; Go, T.H. Quadcopter Formation Flight Control Combining MPC and Robust Feedback Linearization. *J. Frankl. Inst.* 2014, 351, 1335–1355. [CrossRef]
2. Mahmood, A.; Kim, Y. Decentralized Formation Flight Control of Quadcopters Using Robust Feedback Linearization. *J. Frankl. Inst.* 2017, 354, 852–871. [CrossRef]
3. Paredes, J.A.; Acevedo, J.; Mogrovejo, H.; Villalta, J.; Furukawa, R. Quadcopter Design for Medicine Transportation in the Peruvian Amazon Rainforest. In 2016 IEEE XXIII International Congress on Electronics, Electrical Engineering and Computing (INTERCON); IEEE: Piscataway, NJ, USA, 2017. [CrossRef]
4. Nguyen, N.P.; Mung, N.X.; Ha, L.N.N.T.; Huynh, T.T.; Hong, S.K. Finite-Time Attitude Fault Tolerant Control of Quadcopter System via Neural Networks. *Mathematics* 2020, 8, 1541. [CrossRef]
5. Nguyen, N.P.; Hong, S.K. Active Fault-Tolerant Control of a Quadcopter against Time-Varying Actuator Faults and Saturations Using Sliding Mode Backstepping Approach. *Appl. Sci.* 2019, 9, 4010. [CrossRef]
6. Haque, M.R.; Muhammad, M.; Swarnaker, D.; Arifuzzaman, M. Autonomous Quadcopter for Product Home Delivery. In Proceedings of the 2014 International Conference on Electrical Engineering and Information&Communication Technology, Dhaka, Bangladesh, 10–12 April 2014; pp. 2–6. [CrossRef]
7. Xuan-Mung, N.; Hong, S.K.; Nguyen, N.P.; Ha, L.N.N.T.; Le, T.L. Autonomous Quadcopter Precision Landing onto a Heaving Platform: New Method and Experiment. *IEEE Access* 2020, 8, 167192–167202. [CrossRef]
8. Shakernia, O.; Ma, Y.; Koo, T.J.; Sastry, S. Landing an Unmanned Air Vehicle: Vision Based Motion Estimation and Nonlinear Control. *Asian J. Control* 2008, 1, 128–145. [CrossRef]
9. Liu, F.; Shan, J.; Xiong, B.; Fang, Z. A Real-Time and Multi-Sensor-Based Landing Area Recognition System for UAVs. *Drones* 2022, 6, 118. [CrossRef]
10. Safadinho, D.; Ramos, J.; Ribeiro, R.; Filipe, V.; Barroso, J.; Pereira, A. UAV Landing Using Computer Vision Techniques for Human Detection. *Sensors* 2020, 20, 613. [CrossRef]
11. Ononiwu, G.; Okoye, A.; Onojo, J.; Onuekwusi, N. Design and Implementation of a Real Time Wireless Quadcopter for Rescue Operations. *Am. J. Eng. Res.* 2016, 5, 130–138.
12. Eid, S.E.; Sham Dol, S. Design and Development of Lightweight-High Endurance Unmanned Aerial Vehicle for Offshore Search and Rescue Operation. In Proceedings of the 2019 Advances in Science and Engineering Technology International Conference, Dubai, United Arab Emirates, 26 March–10 April 2019. [CrossRef]
13. Chen, J.; Li, S.; Chen, S.; He, S.; Shi, Z. Q-Charge: A Quadcopter-Based Wireless Charging Platform for Large-Scale Sensing Applications. *IEEE Netw.* 2017, 31, 56–61. [CrossRef]
14. Rohan, A.; Rabah, M.; Asghar, F.; Talha, M.; Kim, S.H. Advanced Drone Battery Charging System. *J. Electr. Eng. Technol.* 2019, 14, 1395–1405. [CrossRef]
15. Silverman, M.; Jung, B. Staying Alive Longer: Autonomous Robot Recharging Put to the Test. *USC Cent. Robot. Embed. Syst. Tech. Rep.* 2003, 3, CRES-03-015.
16. Valenti, M.; Dale, D.; How, J.P.; De Farias, D.P.; Vian, J. Mission Health Management for 24/7 Persistent Surveillance Operations. *Collect. Tech. Pap.—AIAA Guid. Navig. Control Conf.* 2007, 2, 1767–1784. [CrossRef]
17. Tansuriyong, S.; Kyan, M.; Numata, K.; Taira, S.; Anezaki, T. Verification Experiment for Drone Charging Station Using RTK-GPS. In 2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIIBMS); IEEE: Piscataway, NJ, USA, 2017; pp. 229–232.
18. Rabah, M.; Rohan, A.; Mohamed, S.A.S.; Kim, S.H. Autonomous Moving Target-Tracking for a UAV Quadcopter Based on Fuzzy-PI. *IEEE Access* 2019, 7, 38407–38419. [CrossRef]
19. Tzoumanikas, D.; Li, W.; Grimm, M.; Zhang, K.; Kovac, M.; Leutenegger, S. Fully Autonomous Micro Air Vehicle Flight and Landing on a Moving Target Using Visual–Inertial Estimation and Model-Predictive Control. *J. Field Robot.* 2019, 36, 49–77. [CrossRef]

20. Almeshal, A.M.; Alenezi, M.R. A Vision-Based Neural Network Controller for the Autonomous Landing of a Quadrotor on Moving Targets. *Robotics* 2018, 7, 71. [CrossRef]

21. Qi, Y.; Jiang, J.; Wu, J.; Wang, J.; Wang, C.; Shan, J. Autonomous Landing Solution of Low-Cost Quadrotor on a Moving Platform. *Robot. Auton. Syst.* 2019, 119, 64–76. [CrossRef]

22. Ho, H.W.; De Croon, G.C.H.E.; Van Kampen, E.; Chu, Q.P.; Mulder, M. Adaptive Gain Control Strategy for Constant Optical Flow Divergence Landing. *IEEE Trans. Robot.* 2018, 34, 508–516. [CrossRef]

23. Borowczyk, A.; Nguyen, D.T.; Van Nguyen, A.P.; Nguyen, D.Q.; Saussié, D.; Le Ny, J. Autonomous Landing of a Quadcopter on a High-Speed Ground Vehicle. *J. Guid. Control. Dyn.* 2017, 40, 2373–2380. [CrossRef]

24. Kurs, A.; Karalis, A.; Moffatt, R.; Joannopoulos, J.D.; Fisher, P.; Soljačić, M. Wireless Power Transfer via Strongly Coupled Magnetic Resonances. *Science* 2007, 317, 83–86. [CrossRef]

25. Campi, T.; Cruciani, S.; Maradei, F.; Feliziani, M. Near-Field Reduction in a Wireless Power Transfer System Using LCC Compensation. *IEEE Trans. Electromagn. Compat.* 2017, 59, 686–694. [CrossRef]

26. Campi, T.; Cruciani, S.; De Santis, V.; Feliziani, M. EMF Safety and Thermal Aspects in a Pacemaker Equipped with a Wireless Power Transfer System Working at Low Frequency. *IEEE Trans. Microw. Theory Tech.* 2016, 64, 375–382. [CrossRef]

27. Feliziani, M.; Campi, T.; Cruciani, S.; Maradei, F.; Grasselli, U.; Macellari, M.; Schirone, L. Robust LCC Compensation in Wireless Power Transfer with Variable Coupling Factor Due to Coil Misalignment. In Proceedings of the 2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC), Rome, Italy, 10–13 June 2015; pp. 1181–1186. [CrossRef]

28. Campi, T.; Cruciani, S.; Feliziani, M.; Maradei, F. High Efficiency and Lightweight Wireless Charging System for Drone Batteries. In Proceedings of the 2017 AEIT International Annual Conference, Cagliari, Italy, 20–22 September 2017; pp. 1–6. [CrossRef]

29. Quaglia, G.; Visconte, C.; Scimmi, L.S.; Melchiorre, M.; Cavallone, P.; Pastorelli, S. Design of a UGV Powered by Solar Energy for Precision Agriculture. *Robotics* 2020, 9, 13. [CrossRef]

30. Williams, A.; Yakimenko, O. Persistent Mobile Aerial Surveillance Platform Using Intelligent Battery Health Management and Drone Swapping. In Proceedings of the 2018 4th International Conference on Control, Automation and Robotics (ICCAR), Auckland, New Zealand, 20–23 April 2018; pp. 237–246. [CrossRef]

31. Bitcraze. Available online: https://www.bitcraze.io/products/crazyflie-2-1/ (accessed on 17 December 2020).

32. CrazyRadio PA. Available online: https://www.bitcraze.io/products/crazyradio-pa/ (accessed on 17 December 2020).

33. Qi Wireless Charging Standard. Available online: https://www.bitcraze.io/products/qi-1_2-charger-deck/ (accessed on 17 December 2020).

34. Crazyflie 2.1 Firmware. Available online: https://github.com/bitcraze/crazyflie-firmware (accessed on 17 December 2020).

35. Phil, K.; Lynn, H. *Kalman Filter for Beginners: With Matlab Examples*; Createspace: Scotts Valley, CA, USA, 2011.

36. Muhammad, T.; Fugqan, A.; Ali, R.; Mohammed, R.; Sung Ho, K. Fuzzy logic-based robust and autonomous safe landing for UAV Quadcopter. *Arab. J. Sci. Eng.* 2019, 44, 2627–2639. [CrossRef]

37. Mohammad, R.H.; Abdyl Rahman, E.B. Automatic UAV Wireless charging over solar vehicle to enable frequent flight missions. In Proceedings of the 2019 3rd International Conference on Automation, Control and Robots, Prague, Czech Republic, 11–14 October 2019.

38. Can, A.; Efstatihedae, H.; Montazeri, A. Design of a Chattering-Free Sliding Mode Control System for 407 Robust Position Control of a Quadrotor. In Proceedings of the 2020 International Conference Nonlinearity, Information and Robotics (NIIR), Innopolis, Russia, 3–6 December 2020.