remote sensing

Review

A Review on IoT Deep Learning UAV Systems for Autonomous Obstacle Detection and Collision Avoidance

Paula Fraga-Lamas [1,2,*], Lucía Ramos [2,3], Víctor Mondéjar-Guerra [2,3] and Tiago M. Fernández-Caramés [1,2]

[1] Department of Computer Engineering, Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain
[2] Centro de investigación CITIC, Universidade da Coruña, 15071 A Coruña, Spain
[3] Department of Computer Science, Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain
* Correspondence: paula.fraga@udc.es; Tel.: +34-981167000 (ext. 6051)

Received:19 August 2019; Accepted: 10 September 2019 ; Published: 14 September 2019

Abstract: Advances in Unmanned Aerial Vehicles (UAVs), also known as drones, offer unprecedented opportunities to boost a wide array of large-scale Internet of Things (IoT) applications. Nevertheless, UAV platforms still face important limitations mainly related to autonomy and weight that impact their remote sensing capabilities when capturing and processing the data required for developing autonomous and robust real-time obstacle detection and avoidance systems. In this regard, Deep Learning (DL) techniques have arisen as a promising alternative for improving real-time obstacle detection and collision avoidance for highly autonomous UAVs. This article reviews the most recent developments on DL Unmanned Aerial Systems (UASs) and provides a detailed explanation on the main DL techniques. Moreover, the latest DL-UAV communication architectures are studied and their most common hardware is analyzed. Furthermore, this article enumerates the most relevant open challenges for current DL-UAV solutions, thus allowing future researchers to define a roadmap for devising the new generation affordable autonomous DL-UAV IoT solutions.

Keywords: UAV; drone; autonomous UAV; UAS; remote sensing; deep learning; image processing; large-scale datasets; collision avoidance; obstacle detection

1. Introduction

The Internet of Things (IoT) is expected to connect to the Internet more than 75 billion devices in 2025 [1]. Such devices are used in a wide variety of fields, like agriculture [2], industry [3–5], or environmental monitoring [6]. However, IoT still faces some challenges, especially regarding security [7,8], which result in a slowdown for its widespread adoption.

Unmanned Aerial Vehicles (UAVs) have an enormous potential for enabling novel IoT applications thanks to their low maintenance cost, high mobility and high maneuverability [9]. Due to such characteristics, UAVs have been really useful in a number of fields and applications, like remote sensing, real-time monitoring, disaster management, border and crowd surveillance, military applications, delivery of goods, or precision agriculture [10,11]. The use of UAVs has been also suggested for providing services in a number of industrial applications, like critical infrastructure inspections [12–16], sensor monitoring [17,18], automatic cargo transport [19,20], or logistics optimization of UAV swarms [21]. In the case of UAVs that operate in industrial environments, they require certain characteristics that may differ substantially from other applications [22].

For instance, UAV maneuvering in such environments needs to consider the presence of workers, mobile vehicles, robots, and heavy-duty tools in order to avoid collisions.

Most of the previously mentioned IoT applications require operating without pilot intervention, so there is a growing interest in the development of techniques that enable UAV autonomous flights. Among the different computational techniques that can be used to detect obstacles and to avoid collisions, Deep Learning (DL) techniques have arisen as a promising alternative that when applied to UAVs, derive into the concept of Deep Learning-Unmanned Aerial Vehicles (DL-UAVs). The main advantage of the application of DL to UAVs is the ability to recognize complex patterns from the raw input data captured by the sensors incorporated in the UAV, learning proper hierarchical representations of the underlying information at different levels.

This article analyzes the latest developments on DL-UAV systems and includes the following main contributions.

- The most relevant DL techniques for autonomous collision avoidance are reviewed, as well as their application to UAV systems.
- The latest DL datasets that can be used for testing collision avoidance techniques on DL-UAV systems are described.
- The hardware and communications architecture of the most recent DL-UAVs are thoroughly detailed in order to allow future researchers to design their own systems.
- The main open challenges and current technical limitations are enumerated.

It must be noted that in this article the term UAV system is used, as it has been commonly used in the literature for years to name unmanned aerial systems. However, most reputable international organizations (e.g., EUROCONTROL, EASA, FAA, and DoD) have adopted Unmanned Aerial System (UAS) as the correct official term for a system that consists of a ground control station, communication transceivers and an aircraft.

The rest of this article is structured as follows. Section 2 introduces the related work. Section 3 analyzes the use of DL techniques and datasets for autonomous collision avoidance, emphasizing their application to UAVs. Section 4 reviews the hardware of the most relevant DL-UAV developments and their communications architecture, as well as their main subsystems. Finally, Section 5 enumerates currently open challenges, and Section 6 is devoted to the conclusions.

2. Related Work

Autonomy in a UAV can be defined as the capability for being self-guided without human intervention [23]. According to the UAV autonomy levels indicated in the work by the authors of [23], the most basic level requires the UAV to be guided with a remote control, whereas the maximum level involves performing a complex mission with human level decision-making without external intervention. In the case of this review, intermediate autonomy level approaches are analyzed, in which the UAV should be able to perform path-planning and detect existing obstacles in order to avoid collisions during navigation.

Depending on the degree of autonomous navigation and on the UAV application, certain problems may arise. For example, in the case of following a fixed path, if the UAV deviates from it to avoid a collision, it would be desirable to know its position during all the navigation to return to such a path. In an outdoor environment, this issue can be solved in a relatively easy way just by making use of a Global Positioning System (GPS), but in indoor scenarios visual odometry techniques should be added [24]. In addition, if the existing obstacles move throughout the scenario, object detection algorithms should be incorporated to the system in order to estimate their next position, thus anticipating potential collisions.

Collision avoidance classical approaches make use of techniques like Simultaneous Localization and Mapping (SLAM) [25,26] and Structure from Motion (SfM) [27] to generate or update a map that represents the visual geometry of the environment, which allows for inferring obstacles and traversable spaces. These techniques use data captured by sensors like RGB-D cameras, Light Detection and Ranging (LIDAR) sensors, or Sound Navigation and Ranging (SONAR) sensors, either individually [28] or by fusing several of them [29]. However, it must be considered that the mentioned sensors can be expensive and still present some limitations (e.g., a LIDAR can suffer from mirror reflections, whereas a GPS does not work properly indoors). Alternatively, computer vision methods can be used, which involve tasks such as optical flow or depth estimation from the data captured by monocular and stereo cameras [30,31]. Such computer vision techniques require using software algorithms and a single camera, which is a considerably cheaper option than the previously mentioned sensors. Nonetheless, further research is needed regarding this latter alternative, as it requires more computational resources.

In the literature, there are some surveys and reviews of methods for obstacle detection and collision avoidance in different application domains. For instance, the authors of [32] provide a systematic review of sensors and vision-based techniques for vehicle detection and tracking for collision avoidance systems on the specific context of on-road driving. Another interesting review can be found in [33], where the authors focus on space robotics and identify the major challenges in collision-free trajectory planning for manipulators mounted on large orbital structures, small satellites, and space robots that navigate in proximity to large orbital structures. Other authors have reviewed the main techniques for improving the autonomy of unnamed surface vehicles with regard to navigation, guidance, control and motion planning with respect to the international regulations for avoiding collisions at sea [34].

In the specific context of UAV navigation, the authors of [31] reviewed vision-based techniques for positioning and mapping, obstacle avoidance, and path planning. Regarding the use of DL techniques for robotic solutions, in the work by the authors of [35], a survey is presented on the use of DL techniques for robotic perception, robotic control and exploration, and robotic navigation and autonomous driving.

Regarding the application of UAVs with attached sensors (i.e., UASs) to remote sensing applications, note that the latest advances have allowed for monitoring environmental processes and changes produced at spatial and temporal scales that would be difficult or impossible to detect by means of conventional remote sensing platforms [36]. Moreover, the characteristics of small UAS and their impact in the context of remote sensing models have been studied in the literature [37], identifying novel remote sensing capabilities as well as the challenges that this type of platforms entails. In particular, the affordability and potential for ubiquitous operation of small UASs enable advances in the type and quality of information that can be collected, and therefore in the applications that can be addressed by this type of remote sensing systems [37]. The main capabilities that small UASs can provide to remote sensing are dependent on their operation without direct human supervision, involving autonomous deployment, data collection, landing and data transfer [37]. In this regard, it is worth mentioning the work detailed in the work by the authors of [36], which provides a review on the recent progress made in remote sensing for small UASs in different fields that involve photogrammetry, multispectral and hyperspectral imaging, and synthetic aperture radars and LIDARs. Additionally, in [38], the authors present a review on remote sensing tasks that involve data acquired by UAVs. Such a review focuses on solutions that specifically address issues derived from the specific nature of the collected data (e.g., ultra-high resolution, geometric, and spectral data, or the fusion of data from multisensor acquisition). In this regard, it is worth mentioning that remote sensing systems that use data acquired by UAVs may involve four types of resolutions: spatial, spectral, radiometric, and temporal [37]. In the case of DL-UAV systems, in this article, only spatial resolution was considered.

There are only a few recent articles on the use of DL methods for UAVs. For instance, in the work by the authors of [39], the current literature is systematically reviewed according to a taxonomy that includes different abstraction levels related to perception, guidance, navigation, and control. Other authors focused only on solving specific problems (e.g., counting cars in UAV images [40]) or on improving specific algorithms [41,42]. Nevertheless, most of the current literature cannot be considered IoT-enabled, and, to the best of our knowledge, there are no articles that review the whole

DL-UAV architecture while focusing on the problem of autonomous obstacle detection and collision avoidance.

3. Deep Learning in the Context of Autonomous Collision Avoidance

During the last decade, DL has demonstrated to be an excellent technique in the area of artificial intelligence, solving different problems, and even surpassing humans in some cases [43,44]. In addition, DL yields good results in diverse areas like image recognition [45], medical imaging [46], or speech recognition [47].

Classic Machine Learning (ML) approaches require designing feature extraction methods in order to generate a descriptor (i.e., a feature vector that emphasizes the pattern to detect and that is usually more compact than the raw data). The descriptor feeds a learning algorithm that performs a specific task, such as a classification or a regression of the input data. In contrast to such an approach, DL methods are able to perform both tasks (the representation of the data and the classification) at the same time just by feeding the network with raw data [43]. Moreover, DL techniques excel when learning data representations from raw sensor data acquired by robotic solutions in real environments [39]. In the specific context of autonomous collision avoidance, DL techniques have shown their effectiveness when solving a wide variety of robotic tasks in areas like perception, planning, positioning, and control [35], which involve learning complex behaviors.

Among the DL techniques for collision avoidance there are end-to-end approaches that map directly the raw sensor data captured by the robotic system into a set of possible actions [48–50]. Such approaches are based on learning the behavior of an expert pilot at a given scenario in a real-world environment. There are other DL techniques whose execution requires several stages that involve intermediate representations (e.g., depth maps) that estimate the distance to potential obstacles, the UAV pose or its odometry in order to recalculate the path to reach a goal position. These DL approaches usually include a module for situational awareness that generates a set of feature maps related to the state of the robotic system and its surroundings, and then such computed feature maps feed up a second module for the decision-making process. Therefore, the combination of the mentioned two modules make up a complex network that takes raw sensor data as input and generates the motion control commands for the robotic system [48,51–55].

Regarding the learning paradigm, Reinforcement Learning (RL) algorithms [56] have been widely used in robotic systems. The essential idea in deep RL methods is that an agent extracts feedback from the interaction with real or simulated environments. Thus, given a specific state and based on previous experience, the agent can infer which action maximizes a predefined goal. Several approaches use RL methods in order to learn effective collision avoidance policies that require experience on successful trajectories as well as on undesirable events like collisions [51,53,54,57–59]. This use of simulated environments allows for collecting a large amount of data in an easy way. Specifically, AirSim [60] enables navigating freely with a car or a drone through different virtual scenarios, providing the raw information of different sensors (e.g., RGB cameras and depth cameras), which are synchronized with the six Degrees of Freedom (DoF) of the vehicle and the segmentation maps. Other approaches use supervised learning algorithms that are based on learning through examples. Such examples consist of a representation of the environment provided by the raw sensor data and the actions taken by an expert in the same conditions. The learning process returns the policy that best imitates the action of the experts according to the given examples [49,50,52,55]. A limiting factor of the previous supervised approaches is the need for large amounts of data to train models with generalization capabilities in different real-world environments. This necessity involves human experts during the collection and annotation of the data. In order to minimize human effort, self-supervised algorithms automate the collection and annotation processes to generate large scale datasets [48,49].

In addition to the mentioned learning techniques, transfer learning [61] allows for starting from a model previously trained with related data. Such a model is further trained for a specific application domain, thus reducing the total amount of data needed for training. For example, the policies learned from data collected by cars and bicycles can be applied to a UAV for autonomous flights [62]. Such navigation policies are generic, despite having been trained from the viewpoint of urban vehicles. Transfer learning from models trained on virtual worlds, but applied to real data, have also been studied in the literature [59,63–65].

There are different DL architectures. On the one hand, a Convolutional Neural Network (CNN) is a type of hierarchical architecture for feature extraction that allows for generating a set of high-dimensional feature maps from the raw sensor data acquired for the robotic system. Thus, CNN-based models [45] are widely used in the context of collision avoidance, especially in the stages related to environment perception or to depth estimation [52]. On the other hand, in the context of continuous autonomous flights, the use of temporary information becomes relevant. Considering that the information extracted from the sensors at each observation is related to a partial representation of the environment surrounding the robotic system, the possibility of storing and tracking all the relevant information obtained in the past allows for getting a more complete representation of the environment. For such a purpose, the use of Recurrent Neural Network (RNN) architectures [66] provides the ability to learn temporal dependencies from the information of an arbitrarily long sequence of observations. Therefore, there are RNN-based approaches that allow for keeping track of past observations such as the depth maps obtained from RGB images [53], poses, or recently avoided obstacles. In addition, the incorporation of Temporal Attention [67] allows for evaluating the information given by each observation of a sequence, and it can be used by weighting each of the recent observations based on its importance in decision-making. Such a technique increases training speed and provides a better generalization over the training dataset. A type of RNN is Long Short-Term Memory (LSTM) [68], which is capable of learning long-term dependencies with the ability of removing old information or by adding new information at any point, thus allowing the use of all the relevant past information.

3.1. On the Application of DL to UAVs

Table 1 summarizes the main characteristics of the approaches that enable UAVs for autonomous flight through the use of DL techniques. All the methods mentioned in the Table feed DL networks with RGB images whose resolution ranges from 360 p to 720 p.

Depending on their final goal, it can be distinguished among three types of methods. Some methods learn to navigate from raw images, indicating the next movement of the UAV, thus being able to navigate avoiding obstacles [49,59,62,69]. Other authors proposed specific solutions for certain applications that involve following a trail or a gate [55,70]. However, such methods do not include a collision avoidance module. There are also other methods that generate an obstacle map, which represents the probability of collision on the scene or the distance to nearby obstacles [53].

It is important to note that none of the compared systems make use of depth maps extracted directly from stereo or depth cameras. However, some try to estimate depth maps from just one single RGB camera [53]. The combination of depth maps provided by a hardware sensor with RGB images that feed a DL architecture is an interesting approach that has not been described so far in the literature.

Regarding the application scenario, the most relevant solutions of Table 1 are focused on unstructured outdoor environments such as forests, which enable applications like search and rescue, wilderness monitoring, target tracking, exploration, or environmental mapping. For instance, in [55], the authors describe a Micro Aerial Vehicle (MAV) for autonomous low-flying that follows a artificial trail. This approach is based on a Deep Neural Network (DNN) architecture for trail detection that uses transfer learning to estimate the view orientation and the lateral offset of the MAV with respect to the center of the trail. Another proposal in forest environments is presented in the work by the authors of [71], which uses a pretrained AlexNet [45] for tree detection and for the prediction of the direction for collision avoidance. Other outdoor UAV solutions are trained for urban spaces, such as CNN-based models that detect all the potential obstacles [72] or that control a UAV through the streets of a city environment [62].

Table 1. Comparison of relevant Deep Learning-Unmanned Aerial Vehicles (DL-UAV) collision avoidance techniques.

Reference	Architecture	Goal and Details	Scenario
[55]	CNN	Two deep networks: one for detection of the rail center and another for obstacle detection	Outdoors
[70]	CNN	The network detects the gate center. Then, an external guidance algorithm is applied.	Indoors
[69]	CNN	The network returns the next flight command. The commands are learned as a classification. The dataset contains several sample flights performed by a pilot.	Indoors
[49]	CNN	The network returns the next flight command. The commands are learned as a classification. Self-supervised data for training.	Indoors
[50]	CNN	The network returns the next flight command. The commands are learned as a classification. The dataset contains several sample flights annotated manually.	Indoors
[62]	CNN	The network returns the next steering angle and the probability of collision. The steering angle is learned by a regression.	Outdoors
[59]	CNN + RL	The CNN returns a map that represents the action space for the RL method. The method is trained on a virtual environment.	Indoors
[53]	cGAN + LSTM + RL	The first cGAN estimates a depth map from the RGB image. Then, these maps feed an RL with LSTM to return the flight command.	Indoors
[71]	CNN	The network returns the distance to tree obstacles. The distance is learned as three classes.	Outdoors
[72]	CNN	Object detection.	Outdoors
[73]	CNN	The network computes feature extraction for learning safe trajectories.	Outdoors
[48]	Two stream CNN	The network makes use of two consecutive RGB images and returns the distance to obstacles in three directions.	Indoors

Approaches for indoor scenarios cover applications of surveillance, transportation, goods delivery, inspection tasks or diverse tasks in manufacturing environments. For instance, Kouris et al. [48] proposed an approach that maps each input RGB image to a flight command. Such a solution is composed of a two-stream CNN for UAV perception that takes as inputs two RGB consecutive frames in order to extract spatio-temporal features for predicting the distance to the potential obstacles. The CNN is trained with a custom dataset of UAV real-flight indoor trajectories with distance labels indicating the closest obstacle. The collection and annotation of the dataset is performed in a self-supervised manner by means of a UAV equipped with three pairs of ultrasonic and infrared distance sensors that enable data automatic annotation. Then, a local motion planning policy translates the distance predictions to a single control command that modulates the yaw and forward linear velocity to guide the UAV towards the direction for which the longest navigable space is predicted within the current FoV. Another custom dataset is created to train the end-to-end approach proposed by Gandhi et al. [49]. Such a work, in contrast to other datasets related to collision avoidance, uses a UAV crash dataset in order to collect the different ways in which a UAV can crash. This negative flying data is used in conjunction with the equivalent positive data sampled from the same trajectories to learn a robust policy for UAV navigation. This CNN-based approach uses, in particular, an AlexNet architecture [45] that is initialized with pretrained weights. Self-supervised learning is used to predict the probability to move in a certain direction. Then, the decision to turn the UAV to the left or to the right while moving forward is taken according to the confidence of the CNN predictions. Another custom dataset was created to train the system proposed in the work by the authors of [50], which focuses on enabling autonomous navigation in indoor corridors by means of the DenseNet-161 architecture [74] with pretrained weights.

The dataset is composed by images at several positions from corridors with different lengths captured by means of a front-facing camera attached to a quadcopter and the corresponding ground-truth values in terms of in-flight commands. The model is trained over the custom dataset using supervised learning to predict the probability of different classes that decide whether to move forward, shift left/right, or stop. Sadeghi et al. [59] explore the possibility of performing collision-free indoor flights in the real world by only training with data generated in a simulated environment. Such an approach combines CNN networks to process the raw input data with RL without requiring any human demonstration data or real images during training. Singla et al. [53] proposed a two-way method for UAV navigation that avoids stationary and mobile obstacles. The proposed method involves intermediate depth estimations and a RNN-base method with RL that incorporates Temporal Attention [67] to integrate the relevant data gathered over time.

Among the most popular e-sports, drone racing has recently gained a lot of popularity, giving rise to the call for challenges to promote the development of this field, such as the autonomous drone racing competitions celebrated during IROS conference. These competitions motivated the development of UAV systems that make use of DL techniques. For example, in [70], the authors consider the nature of the competitions, which involve moving as fast as possible through a series of gates. Thus, the researchers devised a real-time gate detection network that is complemented with a guidance algorithm.

3.2. Datasets

The previously mentioned DL techniques provide effective solutions that extract hierarchical abstractions from large amounts of data through the training stages. There are several publicly available large-scale datasets that include labeled data acquired from real or simulated environments in challenging conditions:

- KITTI online benchmark [75] is a widely-used outdoor dataset that contains stereo gray and color video, 3D LIDAR, inertial, and GPS navigation data for depth estimation, depth completion or odometry estimation.
- IDSIA forest trail dataset [76] includes footage of hikes recorded by three cameras to predict view orientation (left/center/right) throughout a large variety of trails.
- Udacity [62] provides over 70,000 images of car driving distributed over six experiments, which include timestamped images from three cameras (left, central, right) and the data from different sensors (e.g., an IMU, GPS, gear, brake, throttle, steering angles, and speed).
- UAV crash dataset [49] focuses on representing the different ways in which a UAV can crash. It contains 11,500 trajectories of collisions in 20 diverse indoor environments collected over 40 flying hours.
- Oxford RobotCar Dataset [77] provides data of six cameras, LIDAR, GPS, and inertial navigation captured from 100 repetitions of a consistent route over a period of over a year. This dataset captures many different combinations of weather, traffic, and pedestrians, along with longer term changes such as construction and roadworks.
- SYNTHIA [78] is a synthetic dataset in the context of driving scenarios that consists of a collection of photo-realistic frames rendered from a virtual city including diverse scenes, variety of dynamic objects like cars, pedestrians and cyclists, different lighting conditions and weather. The sensors involved in the dataset include eight RGB cameras that conform a binocular 360^{o} camera and eight depth sensors.
- AirSIM [60] is a simulation environment that allows for simulating UAV trajectories in different scenarios and conditions that produce raw images, depth maps and the data that represent the UAV trajectories.

In addition to the data specifically gathered to generate a training dataset, techniques of data augmentation are frequently used to obtain more representative data that cover different conditions in order to train models with high generalization capabilities. Methods for data augmentation usually include left–right flips, random horizontal flips, random crops, or variations in scale, contrast, brightness, saturation, sharpness, rotation, and jitter with random permutations of these transformations [55,79].

4. DL-UAV Hardware and Communications Architecture

The underlying DL-UAV hardware and its communications architecture are essential for the success of the system, as they provide the support for implementing DL techniques and advanced features. Specifically, the use of a DL approach for autonomous obstacle detection and collision avoidance imposes UAV hardware and communications design restrictions (e.g., a high computational cost) that should be carefully addressed. This section reviews the latest DL-UAV communication architectures, detailing the main components of the different subsystems and analyzing the most common hardware used for developing such subsystems.

4.1. Typical Deep Learning UAV System Architecture

Figure 1 shows the typical cloud-based communications architecture for DL-UAV systems. Such an architecture is divided into three main layers:

- The layer at the bottom is the UAV layer, which includes the aerial vehicles and their subsystems:
    - Propeller subsystem: it controls the UAV propellers through the motors that actuate on them.
    - Main control subsystem and flight control subsystem: it is responsible for coordinating the actions of the other subsystems and for collecting the data sent by them. In some cases, it delegates the maneuvering tasks on a flight controller that receives data from different sensors and then acts on the propeller subsystem.
    - Sensing and Actuation subsystem: it makes use of sensors to estimate diverse physical parameters of the environment and then actuates according to the instructions given by the on-board controllers.
    - Positioning subsystem: it is able to position the UAV outdoors (e.g., through GPS) or indoors (e.g., by using reference markers).
    - Communications subsystem: it allows for exchanging information wirelessly with remote stations.
    - Power subsystem: it provides energy and voltage regulation to the diverse electronic components of the UAV.
    - Storage subsystem: it stores the data collected by the different subsystems. For instance, it is often used for storing the video stream recorded by an on-board camera.
    - Identification subsystem: it is able to identify remotely other UAVs, objects, and obstacles. The most common identification subsystems are based on image/video processing, but wireless communications systems can also be used.
    - Deep Learning subsystem: it implements deep learning techniques to process the collected data and then determine the appropriate response from the UAV in terms of maneuvering.
- The data sent by every UAV are sent to either a remote control ground station or to a UAV pilot that makes use of a manual controller. The former case is related to autonomous guiding and collision-avoidance systems, and often involves the use of high-performance computers able to collect, process, and respond in real-time to the information gathered from the UAV. Thus, such high-performance computers run different data collection and processing services, algorithms that make use of deep learning techniques, and a subsystem that is responsible for sending control commands to the UAVs.
- At the top of the architecture is the remote service layer, which is essentially a cloud composed by multiple remote servers that store data and carry out the most computationally-intensive data processing tasks that do not require real-time responses (e.g., certain data analysis). It is also usual to provide through the cloud some kind of front-end so that remote users can manage the DL-UAV system in a user-friendly way.

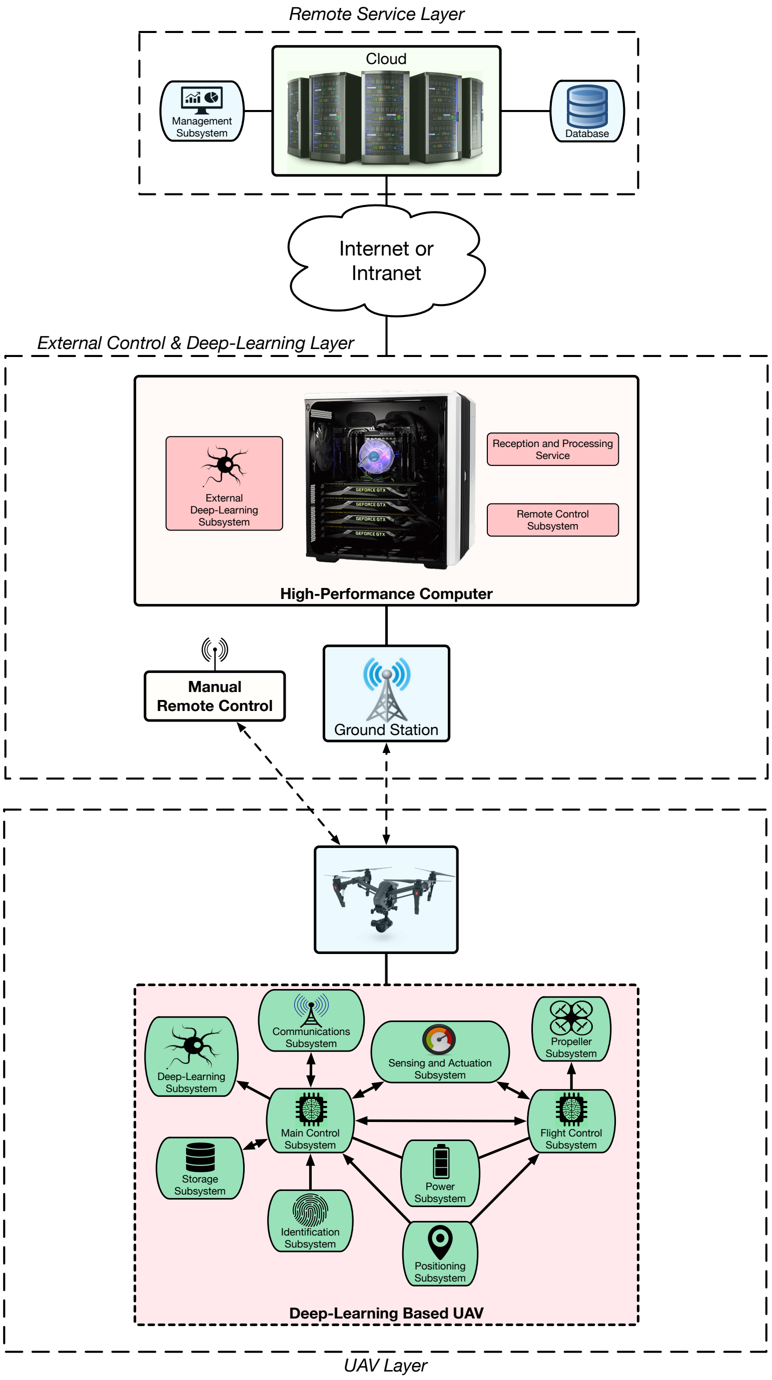

Figure 1. Typical cloud-based deep learning UAV-based communications architecture.

4.2. Advanced UAV Architectures

In the last years, UAV IoT architectures evolved towards more sophisticated architectures that essentially attempt to avoid the reliance on a remote cloud. This is due to the fact that a cloud may not scale properly, and thus it may constitute a bottleneck when a significant number of UAVs or IoT devices exchange communications with it [80]. In addition, cyberattacks may compromise the availability of the cloud servers, therefore preventing the UAV system from working properly.

Due to the previously mentioned reason, UAV architectures evolved toward IoT edge computing architectures, which allow unburdening the cloud from part of its tasks through the use of devices on the edge of a network (i.e., in the limit between the last network routing devices and the sensing/actuation devices that act on the physical world). There are basically two variants of edge computing that are commonly applied to UAV applications, depicted in Figure 2:

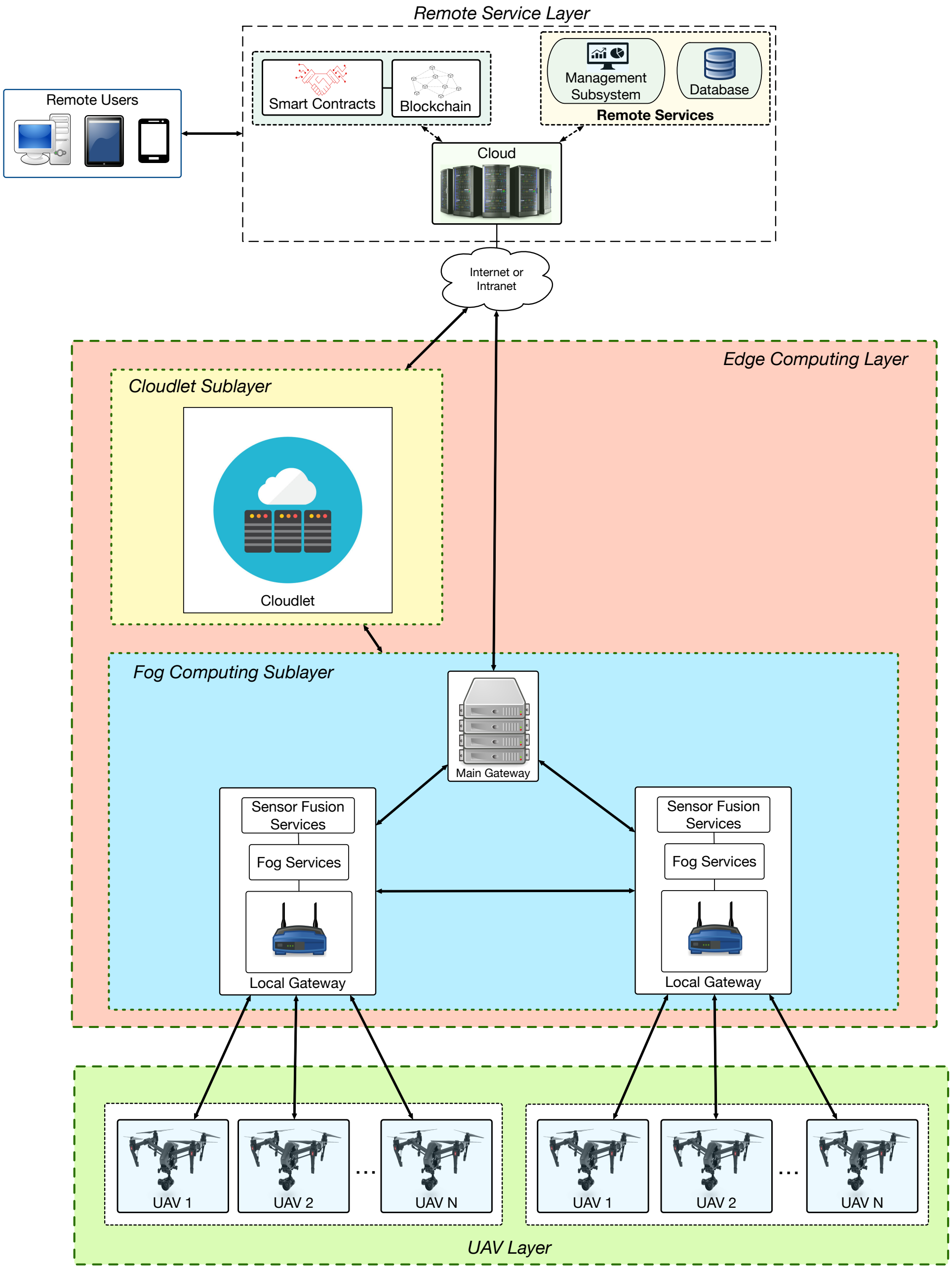

Figure 2. Advanced DL-UAV communications architecture.

- Fog Computing: It is based on the deployment of medium-performance gateways (known as fog gateways) that are placed locally (close to a UAV ground station). It is usual to make use of Single-Board Computers (SBCs), like Raspberry Pi [81], Beagle Bone [82], or Orange Pi PC [83], to implement such fog gateways. Despite its hardware, fog gateways are able to provide fast responses to the UAV, thus avoiding the communications with the cloud. It is also worth noting that fog gateways can collaborate among themselves and collect data from sensors deployed on the field, which can help to decide on the maneuver commands to be sent to the UAVs in order to detect obstacles and prevent collisions.
- Cloudlets. They are based on high-performance computers (i.e., powerful CPUs and GPUs) that carry out computationally intensive tasks that require real-time or quasi real-time responses [84].

> Like fog gateways, cloudlets are deployed close to the ground station (they may even be part of the ground station hardware), so that the response lag is clearly lower than the one that would be provided by a remote Internet cloud.

Figure 2 also includes in the Remote Service Layer the use of Distributed Ledger Technologies (DLTs) like blockchain, which have been previously applied to multiple IoT applications [85–87], including UAV systems [88]. In the field of UAV, the main advantage of blockchain and similar DLTs is that they are able to provide information redundancy, data security, and trustworthiness to the potential applications [89]. In addition, a blockchain can execute smart contracts, which translate legal terms into code that can be run autonomously [90], and therefore can automate certain tasks depending on the detection of certain events.

4.3. Hardware for Collision Avoidance and Obstacle Detection Deep Learning UAV Systems

In the literature, diverse researchers have addressed the problem of what hardware may be used by a UAV to prevent collisions and detect obstacles when harnessing the outputs of deep learning systems. Table 2 summarizes the features of the most relevant previous DL-UAV systems that deal with such a topic, including the hardware used for building their communications architecture and the provided DL-based collision-avoidance and obstacle-detection functionality.

As seen in Table 2, most UAVs are commercial platforms. In fact, the majority are manufactured by Parrot due to their easy-to-use developer APIs [91], which allow for monitoring and controlling remotely the UAVs. There are also in the list a couple of self-assembled drones, which need an additional effort from the researchers during the assembly and validation of the UAV, but that enable customizing it with specific sensors, powerful controllers and extended batteries that increase the type and number of tests that can be carried out.

The following subsections analyze the characteristics of the subsystems of the UAVs listed in Table 2 and detail some of the most popular alternatives for implementing each subsystem.

Table 2. Hardware for collision avoidance and obstacle detection deep learning UAV systems.

Reference	UAV Platform	Weight	Max. Payload	Flight Time	Control Subsystem	Sensors	Comm. Interfaces	Positioning Subsystem	Power Subsystem	Storage Subsystem	Propeller Subsystem	Identification Subsystem	DL Subsystem Technique	Onboard/ External DL Processing	Training Hardware
[92]	Crazyflie (assumed version: 2.1)	27 g	15 g	7 min	STM32F405 (Cortex-M4, 168MHz, 192 KB SRAM, 1 MB flash)	3-axis accelerometer/gyroscope (BMI088), high-precision pressure sensor (BMP388), external optical tracking system	BLE	Ultra-Wide Band distance sensor	LiPo Batteries	8 KB EEPROM, 1 MB flash	Quadrotor Micro-Aerial Vehicle (MAV)	UWB	Recurrent Neural Network, Deep Q-Learning Network	N/A	-
[93]	Parrot AR.Drone 2.0	420 g (with hull)	≈500g	12 min (up to 36 min with Power Edition)	Internal: ARM Cortex A8@1 GHz (with a Linux 2.6.32 OS), 1 GB of RAM, video DSP@800 MHz (TMS320DMC64x)	720p@30 fps video camera, 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer, pressure sensor, vertical QVGA camera @ 60 fps for measuring flight speed	Wi-Fi (IEEE 802.11 b/g/n)	Ultrasonic altimeter	LiPo Batteries	Up to 4 GB of flash	Quadrotor	Visual identification	Deep Reinforcement Learning	External (Workstation/cloud with i7 cores and a GTX1080 GPU)	-
[70]	Self-assembled	-	-	-	NVIDIA Jetson TX2, 8 GB of RAM, operating system is based on Ubuntu 16.04 and works in a ROS-Kinetic environment, Flight Control Computer: in-house board ([94])	a PX4Flow for ground speed measurement, PointGrey Firefly VGA camera@30 fps	Wi-Fi (assumed)	ZED stereo camera for odometry measurement, Terabee TeraRanger-One one-dimensional LIDAR for altitude measurement	-	32 GB of eMMC (embedded Multi Media Card)	Quadrotor	Visual identification	Deep Convolutional Neural Network	Onboard (in real-time)	Intel i7-6700 CPU 3.40 GHz, 16 G RAM, NVIDIA GTX 1080 Ti
[95]	Parrot AR.Drone 2.0 Elite Edition	420 g (with hull)	≈500 g	12 min	Internal: ARM Cortex A8@1 GHz (with a Linux 2.6.32 OS), 1 GB of RAM, video DSP@800 MHz (TMS320DMC64x)	720p@30 fps video camera, 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer, pressure sensor, vertical QVGA camera @ 60 fps for measuring flight speed	Wi-Fi (IEEE 802.11 b/g/n)	Ultrasonic altimeter	LiPo Batteries	-	Quadrotor	Visual identification	Deep Convolutional Neural Network	The UAV camera was only used for generating a dataset	Not specified CPU
[62]	Parrot Bebop 2.0	500 g	-	25 min	Dual-core processor with quad-core GPU	1080p video camera, pressure sensor, 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer	Wi-Fi (IEEE 802.11 b/g/n/ac)	GPS/GLONASS, height ultrasonic sensor,	Li-ion Batteries	8 GB Flash	Quadrotor	Visual identification	Deep Convolutional Neural Network	External (the training hardware receives images at 30 Hz from the drone through Wi-Fi)	Intel Core i7 2.6 GHz CPU
[93]	Parrot AR.Drone	420 g (with hull)	-	12 min	ARM9@468 MHz, 128 MB of RAM, Linux OS	VGA camera, vertical video camera @ 60 fps, 3-axis acceleration sensor, 2-axis gyroscope, 1-axis Yaw precision gyroscope	Wi-Fi (IEEE 802.11 b/g)	Ultrasonic altimeter	LiPo Batteries		Quadrotor	Visual identification	Deep Convolutional Neural Network	External (Jetson TX development board)	Desktop computer with an NVIDIA GTX 1080Ti

Table 2. *Cont.*

Reference	UAV Platform	Weight	Max. Payload	Flight Time	Control Subsystem	Sensors	Comm. Interfaces	Positioning Subsystem	Power Subsystem	Storage Subsystem	Propeller Subsystem	Identification Subsystem	DL Subsystem Technique	Onboard/External DL Processing	Training Hardware
[96]	DJI Matrice 100	≈2400 g	≈1200 g	19–40 min	Not indicated	-	-	-	LiPo Batteries	-	Quadrotor	Visual identification	Deep Convolutional Neural Network	External (Jetson TX development board)	Desktop computer with an NVIDIA GTX 1080Ti
[71]	Parrot Bebop 2.0	500 g	-	25 min	Dual-core processor with quad-core GPU	1080p video camera, pressure sensor, 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer	Wi-Fi (IEEE 802.11 b/g/n/ac)	GPS/GLONASS, height ultrasonic sensor,	Li-ion Batteries	8 GB Flash	Quadrotor	Visual identification	Deep Convolutional Neural Network (AlexNet)	External	Computer with a NVIDIA GeForce GTX 970M GPU
[72]	Self-assembled	-	-	-	Pixhawk as flight controller, NVIDIA TX2 for the neural network	-	-	-	-	-	-	-	Depthwise Convolutional Neural Network	Onboard (tested also in external hardware)	Nvidia GTX 1080 and Nvidia TX2
[73]	Modified 3DR ArduCopter	-	-	-	Quad-core ARM	Microstrain 3DM-GX3-25 IMU, PlayStation Eye camera facing downward for real-time pose estimation and one high-dynamic range PointGrey Chameleon camera for monocular navigation	-	-	-	-	Quadrotor	Visual identification	Deep Convolutional Neural Network	External	-
[69]	Parrot Bebop	390 g (with battery)	-	≈10 min	Parrot P7 Dual-Core	1080p video camera, accelerometer, barometer, vertical camera, gyroscope, magnetometer	Wi-Fi (IEEE 802.11 b/g/n/ac)	GPS, ultrasonic altimeter	Li-ion Batteries	8 GB Flash	Quadrotor	Visual identification	Deep Convolutional Neural Network	External (in the training hardware)	NVIDIA GeForce GTX 970M, Intel Core i5, 16 GB memory
[59]	Parrot Bebop	390 g (with battery)	-	≈10 min	Parrot P7 Dual-Core	1080p video camera, accelerometer, barometer, vertical camera, gyroscope, magnetometer	Wi-Fi (IEEE 802.11 b/g/n/ac)	GPS, ultrasonic altimeter	Li-ion Batteries	8 GB Flash	Quadrotor	Visual identification	Deep Convolutional Neural Network		-
[59]	Parrot Bebop 2.0	500 g	-	25 min	Dual-core processor with quad-core GPU	1080p video camera, pressure sensor, 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer	Wi-Fi (IEEE 802.11 b/g/n/ac)	GPS/GLONASS, height ultrasound sensor,	Li-ion Batteries	8 GB Flash	Quadrotor	Visual identification	Deep Convolutional Neural Network (AlexNet)		-

4.3.1. Platform Frame, Weight and Flight Time

The UAV platforms listed in Table 2 can be considered light, with an average weight of roughly 400 g. Nonetheless, extremely light platforms like Crazyflie have also been used for developing Micro-Aerial Vehicle (MAV) applications. In contrast, relatively heavy platforms like DJI Matrice 100, which weights ~2.4 Kg, are able to move heavier payloads (more than 1 Kg). In fact, most of the commercial drones on the list have not been designed to transport additional payloads, but some of them, like Parrot's AR.Drone 2.0, thanks to their powerful propellers, are able to carry up to 500 g.

Regarding flight time, it is related to their weight and maximum payload: the higher the weight, the larger the carried batteries, so the longer the flight time. Moreover, it is worth pointing out that some UAVs have optional batteries and modules that are able to double (e.g., in the case of DJI Matrice 100) or triple (e.g., with the AR.Drone 2.0 Power Edition) the original battery life.

4.3.2. Propeller Subsystem

UAVs are usually conceived for flying indoors and/or outdoors, what influences significantly their flight capabilities. In any case, the selected hardware must provide a good trade-off among cost, payload capacity and reliability. All the UAVs listed in Table 2 are multirotor UAVs (specifically, quadrotors), which offer good reliability and may minimize vibration during operation (this is useful for collecting stabilized images and videos from a DL-UAV), but it is also more expensive and heavier than alternatives like single-rotor UAVs. In contrast, single-rotor UAVs benefit from slower spinning speeds and less power consumption, which is key for some applications where an extended flight time or heavy payload transport are necessary.

4.3.3. Control Subsystem

The controller hardware has greater power in comparison to the traditional IoT node hardware. The majority of the hardware can be considered powerful and low-power; it is the typical hardware used for mobile device platforms. Most controllers run on ARM devices (e.g., ARM Cortex-M4, ARM Cortex A8, ARM9, or ARM A57 Complex in the case of the NVIDIA Jetson TX2), which currently provide the best trade-off for mobile computing devices in terms of power consumption and performance by including 32/64-bit architectures, fast multicore processors, dedicated GPUs and support for managing several gigabytes of RAM. In fact, some of the listed UAVs are so powerful that they actually run certain versions of the Linux operating system.

It is important to note that some UAVs distinguish between the hardware used for controlling the flight and the one that interacts with the other subsystems. For instance, it is usual in self-assembled drones to make use of a PixHawk controller [97] as flight controller, which embeds a programmable processor, an Inertial Measurement Unit (IMU), a GPS, and provides connectors for radio telemetry and control interfaces.

Both the main control subsystem and the flight controller can make use of different electronic devices. The most common are microcontrollers, Application-Specific Integrated Circuits (ASIC), and System-on-Chips (SOCs), but it is also possible to embed Field-Programmable Gate Arrays (FPGAs) or Central Processing Units (CPUs) optimized for mobile computing devices (i.e., that provide a good trade-off between performance and power consumption).

High-performance microcontrollers are probably the most commonly used devices, owing to their low power consumption, their ability to be reprogrammed easily, and the fact that they have enough processing power for carrying out the required control tasks. SoCs usually integrate a medium-to-high performance microcontroller and multiple peripherals (e.g., wireless transceivers), which makes them more appropriate for lightweight systems, but causes them to have greater energy consumption that traditional microcontrollers.

In the case of FPGAs, they are able to provide excellent performance for executing certain deterministic demanding tasks and that can be reconfigured easily with a different design;

but, unfortunately, FPGA design is usually clearly slower (and consequently more expensive) and often consumes more energy due to the requirement to power the used logic continuously.

ASICs provide even higher performance and significantly lower power consumption that FPGAs and other embedded devices thanks to being designed explicitly for very specific applications. Nonetheless, in ASIC, the development cost is very high (in the order of several millions of U.S. dollars), so their application in DL-UAV systems is limited to already programmed Commercial Off-The-Shelf (COTS) products.

4.3.4. Sensing Subsystem

Sensors are essential for UAV maneuvering. The most common are Inertial Measurement Units (IMUs), which include accelerometers, gyroscopes, and magnetometers. Photo and video cameras are also necessary for providing feedback to drone pilots and inputs to neural networks.

Perform control tasks and, at the same time, processing the large amount of data that comes from sensors and cameras is difficult. For instance, there are video cameras that can stream up to 60 frames per second (fps), but such an amount of video data may not be processed in real-time by the on-board UAV hardware, so two alternatives are often implemented: UAVs either sample frames to reduce the input frame rate or they make use of external remote hardware that carries out the video processing tasks. In practice, only a few UAVs perform the mentioned processing on-board in real-time: most UAVs delegate such a task to external systems, which are usually powerful servers that integrate high-performance graphic cards (the characteristics of some of such servers are indicated in the last two columns of Table 2). In addition, such a powerful hardware is often used during the training phase of the deep learning networks in order to save time.

In summary, it can be stated that the most commonly used sensors in DL-UAV systems are as follows.

- IMUs that embed 3-axis accelerometers, gyroscopes and magnetometers.
- High-precision pressure sensors (barometers).
- Photo and video cameras.
- 3D-camera systems (e.g., Kinect).
- Video Graphics Array (VGA)/Quarter Video Graphics Array (QVGA) cameras.
- Laser scanners (e.g., RP-LIDAR).
- Ultrasonic sensors.

The reliability and safety of the previously mentioned sensors and actuators should be ensured to guarantee the security of the DL-UAV system. There are just a few examples in the literature on this specific research area. For instance, the authors of [98] designed a fault detection and diagnosis system for a quadrotor under total failures (e.g., loss of a propeller or a motor) or with partial faults (e.g., degradation or displacement of a component). The obtained numerical results illustrate the effectiveness of the proposed method under partial and total rotor faults. Another example is presented in [99], where the authors propose a hybrid feature model and CNN-based fault diagnosis system validated through flight experiments.

4.3.5. Positioning Subsystem

Besides the sensing subsystem, the positioning subsystem is probably the most important for autonomous UAV navigation. Outdoors, most UAV systems make use of GPS/ Global'naya Navigatsionnaya Sputnikovaya Sistema (GLONASS) in combination with an altimeter based on ultrasound measurements. The most sophisticated UAVs make use of LIDARs and special cameras that measure both odometry and speed. Ultra-Wide Band (UWB) transceivers are only used in one of the UAV systems listed in Table 2 [92], although they have been previously analyzed in the literature when performing accurate (centimeter-precision) distance measurements [100].

Among the different available indoor location techniques, those based on Received Signal Strength Indicator (RSSI) or Received Signal Strength (RSS) have proved their accuracy when positioning in limited areas [101,102], but their heavily depend on characteristics of the scenario (e.g., presence of metallic objects) and on the used UAV hardware (e.g., antennas) [103]. There are also positioning techniques based on computing the Angle of Arrival (AoA) of the received signals [104], or their time of arrival (through Time of Arrival (ToA) and Time Difference of Arrival (TDoA) techniques) [105].

4.3.6. Communications Subsystem

The data to/from the different subsystems is transmitted to remote computers by using different wireless communications technologies. The most common is Wi-Fi (IEEE 802.11 standard), which operates in unlicensed bands and provides a good trade-off among hardware cost, outdoor range and speed rate. Bluetooth Low-Energy (BLE) is also often used, but it is usually aimed at providing short-range communications.

There are also many other communications technologies that can be used by DL-UAVs, like 3G/4G/5G [106,107], ZigBee (IEEE 802.15.4) [103], Long-Range Wide Area Network (LoRaWAN) [108], Ultra Wide Band (UWB) [109], IEEE 802.11ah [110], or Wi-Fi Direct [111]. Table 3 shows a comparison on the main characteristics of the latest communications technologies for DL-UAVs [4], indicating their frequency band, coverage, data rate, power consumption, and potential applications.

Table 3. Main characteristics of the most relevant communications technologies for DL-UAVs.

Technology	Frequency Band	Maximum Range	Data Rate	Power Consumption	Potential UAV-Based Applications
Bluetooth 5 [112]	2.4 GHz	<250 m	up to 2 Mbit/s	Low power and rechargeable (days to weeks)	Beacons, medical monitoring in battlefield [113]
IQRF [114]	868 MHz	hundreds of meters	100 kbit/s	Low power and long range	Internet of Things and M2M applications
Narrowband IoT (NB-IoT) [115]	LTE in-band, guard-band	<35 km	<250 kbit/s	Low power and wide area	IoT applications
LoRa, LoRaWAN [116]	2.4 GHz	< 15 km	0.25–50 kbit/s	Long battery life and range	Smart cities, M2M applications, forest fire detection [108]
SigFox [117]	868–902 MHz	50 km	100 kbit/s	Global cellular network	IoT and M2M applications
Ultrasounds	>20 kHz (2–10 MHz)	<10 m	250 kbit/s	Based on sound wave propagation	Asset positioning and location [118]
UWB/IEEE 802.15.3a	3.1 to 10.6 GHz	< 10 m	>110 Mbit/s	Low power, rechargeable (hours to days)	Fine location, short-distance streaming [109]
Weightless-P [119]	License-exempt sub-GHz	15 km	100 kbit/s	Low power	IoT applications
Wi-Fi (IEEE 802.11b/g/n/ac)	2.4–5 GHz	<150 m	up to 433 Mbit/s (one stream)	High power, rechargeable (hours)	Mobile, business and home applications
Wi-Fi HaLow/IEEE 802.11ah [120]	868–915 MHz	<1 km	100 Kbit/s per channel	Low power	IoT and long-range applications [110]
Wi-Sun/IEEE 802.15.4g [121]	<2.4 GHz	1000 m	50 kbit/s–1 Mbit/s	-	Field and home area networking, smart grid and metering
ZigBee [122]	868–915 MHz, 2.4 GHz	<100 m	20–250 kbit/s	Very low power (batteries last months to years), up to 65,536 nodes	Industrial applications [103]

The DL-UAV communication network is one of the main factors that affect energy consumption. As a result, there is ongoing research on different network topologies such as rings, stars and, specially, on ad hoc networks (i.e., on Flying Ad-hoc Networks (FANETS)). There is also research on path planning and on relay selection to optimize network efficiency. It is important to note that in some IoT applications, delay requirements of the DL-UAV data traffic can be very strict and real-time data transfer models may be required (e.g., in emergency situations). Some alternatives for UAV-based data collection frameworks are presented in the work by the authors of [123]. For instance, a model that increases the efficiency of collaborative UAV data transmission while optimizing energy consumption is described in the work by the authors of [124]. Another example of a data gathering framework for UAV-assisted Wireless Sensor Networks (WSNs) is described in the work by the authors of [125]. In such a paper, to increase the efficiency of data collection, and thus maximize the system throughput, the authors propose to use a priority-based data access scheme that considers UAV mobility to suppress redundant data transmissions between sensor nodes and the UAV. Thus, the authors classify the nodes within the UAV coverage area into different frames according to their locations, and assign them different transmission priorities. The authors also introduce a novel routing algorithm to extend the lifetime of the WSN.

4.3.7. Power Subsystem

A UAV power subsystem is responsible for powering the different electronic components. In the case of the DL-UAVs listed in Table 2, their main energy source are either Li-Po or Li-ion batteries, which are appropriate for high-power electronics (usually of up to several tens of Watts) and for powering the propeller subsystem.

However, it must be noted that the higher the battery power, the heavier the battery. This derives into the fact that current UAV battery technologies like Li-ion or Li-Polymer are a bottleneck when designing systems that require long battery life and reduced weight. Therefore, nowadays, it is necessary either to reduce on-board electronics power consumption by introducing novel technologies (e.g., Polymer Electrolyte Membrane (PEM) fuel cells [126]) or make use of additional power sources. For the latter case, it is possible to embed energy harvesting devices, which can collect ambient energy and store it into the batteries [127] or supercapacitors, which are environmentally friendly and provide an energy density similar to lead–acid batteries, fast charging/discharging speed and a long life cycle [128]. For UAVs, it seems that motion can be harvested by generating mechanical energy that can be collected through thermoelectric systems [129]. Other power sources can be collected through photovoltaic panels [130] or by rectifying electromagnetic waves [131].

Regarding UAV recharging, most systems make use of USB connectors, DC power jacks, or special dock stations where the batteries are connected. There are a few recent examples in the literature of wireless UAV recharging, as wireless rechargers are currently inefficient and require the wireless receiver to be next to the transmitter [132,133].

The impact of the battery on UAV flight time depends mainly on the aircraft and its payload and on the wind conditions, but it is commonly in the range of minutes. As a result, there is a clear need for developing systems that can endure longer flights. There are some recent studies on the topic, but none of them is explicitly focused on DL-UAV systems. For example, the authors of [134] propose an automated battery-management platform for small-scale UAVs (with a typical endurance of less than 10 min) that consists of an autonomous battery change/recharge station. The autonomous battery recharger uses a swapping mechanism with a linear sweeping motion to exchange batteries. The obtained results indicate that the system lasts at least 5 h. The researchers also evaluated the proposed recharge stations through flight validation with Markov Decision Process based planning and a learning algorithm in a 3-h mission with more than 100 swaps. However, the article points out that there are still open issues with the developed recharge station in relation to accurate landing and to non-balanced charging. Another alternative is proposed in [135], where the authors jointly manage the battery levels of UAVs with recharger stations (i.e., solar panels and batteries installed in a set of ground sites in a cellular network). A more sophisticated approach is presented in [136], where a UAV is able to transfer energy and information to several ground IoT devices at the same time, being each device equipped with a power splitter.

4.3.8. Storage Subsystem

As it can be observed in Table 2, most UAVs include several gigabytes of flash memory for storing the collected video and data. Such data are first processed by the main control subsystem in a light way and then stored in static memories such as EEPROMs or SD cards.

Nonetheless, the majority of the DL-UAV systems in Table 2 stream the collected data to a ground station, but it could be upload to a cloud server, a cloudlet or to a fog computing gateway to further process and store the data, which can be ultimately presented to the user through a graphical interface, (e.g., through a web application).

It is worth pointing out that, although local storage can be inexpensive, it is prone to technical failures (e.g., SD card failures) or external factors (e.g., UAV impacts), which can harm the storage subsystem and then loss the stored information. This fact is essential for fostering remote storage subsystems in cloud servers, fog gateways or cloudlets, which provide redundancy. However, the information stored in such systems may be affected by cyberattacks (e.g., Denial-of-Service (DoS) attacks) that affect the availability of the data, and their trustworthiness and integrity. Due to such problems, DLTs like blockchain have been proposed as an alternative for storing certain data (or a proof of such data) securely and between entities that may not trust each other [137–141].

There is recent specific research devoted to UAV deployments with limited storage. For example, the authors of [142] studied joint cooperative transmissions influenced, not only by location planning, but also by strategic content placement. In addition, in [143] the researchers analyzed a swarm of miniature UAVs that make use of distributed in-network storage, which allows them to behave as sensor nodes specialized in detecting certain features. However, none of the above studies takes the storage issues into account when dealing with DL-UAV systems.

4.3.9. Identification Subsystem

The majority of the UAVs stream images collected with a video camera to the deep learning subsystem, which process them in order to determine whether there are obstacles in the path of the drone and then performs the appropriate movement to avoid them.

However, there are other identification technologies that can be used by DL-UAVs, specifically in conjunction with location, inventory, and traceability applications. For instance, Radio-Frequency IDentification (RFID) is one of the most commonly used technologies for such applications [7]. It uses radio frequency transponders (i.e., tags) that transmit unique identifiers and at least one RFID reader that communicates with them. The most relevant advantage of RFID over other traditional identification technologies (e.g., barcodes and QR codes) is that no Line-of-Sight (LoS) is required between the tag and reader.

In the literature, there are several solutions that make use of RFID. For example, in the work by the authors of [144], an autonomous UAV is described that makes use of RFID and self-positioning/mapping techniques based on a 3D LIDAR device. Table 4 shows a comparison of the most relevant UAVs that make use of identification technologies [88].

Table 4. Comparison of features of the most relevant UAVs with identification technologies.

Type of Solution	UAV Characteristics	Identification Technology	Architecture	Application	Additional features
Academic [88]	Indoor/outdoor hexacopter designed from scratch as a trade-off between cost, modularity, payload capacity, and robustness.	RFID	Modular and scalable architecture using Wi-Fi infrastructure.	Inventory and traceability applications	Performance experiments, ability to run decentralized applications.
Academic [145]	Parrot AR Drone2.0 UAV with three attached RFID tags	Ultra High Frequency (UHF) RFID	RFID tracker and pose estimator, precise 6-DoF poses for the UAV in a 3D space with a singular value decomposition method	Localization	Experimental results demonstrate precise poses with only 0.04 m error in position and 2° error in orientation
Commercial, Hardis Group [146]	Autonomous quadcopter with a high-performance scanning system and an HD camera. Battery duration around 20 min (50 min to charge it).	Barcodes	Indoor localization technology. Automatic flight area and plan, 360°anti-collision system.	Inventory	Automatic acquisition of photo data. It includes cloud applications to manage mapping, data processing and reporting. Compatible with all WMS and ERPS and managed by a tablet app.
Commercial, Geodis, and delta drone [147]	Autonomous quadcopter equipped with four HD cameras	Barcodes	Autonomous indoor geolocation technology, it operates when the site is closed.	Plug-and-play solution to all types of Warehouse Management Systems (WMS)	Real-time data reporting (e.g., the processing of data, and its restitution in the warehouse's information system). Reading rates close to 100%.
Commercial, Drone Scan [148]	Camera and a mounted display	Barcodes	DroneScan base station communicates via a dedicated RF frequency with a range of over 100 m. Its software uploads scanned data and position to the Azure IoT cloud and to the customer systems. The imported data are used to re-build a virtual map of the warehouse so that the location of the drone can be determined.	Inventory	Live feedback both on a Windows touch screen and from audible cues as the drone scans and records customizable data. 50 times faster than manual capturing.
Academic [144]	Autonomous Micro Aerial Vehicles (MAVs), RFID reader, and two high-resolution cameras	RFID, multimodal tag detection.	Fast fully autonomous navigation and control, including avoidance of static and dynamic obstacles in indoor and outdoor environments. Robust self-localization based on an onboard LIDAR at high velocities (up to 7.8 m/s)	Inventory	-
Academic [149]	IR-based camera	QR	Computer vision techniques (region candidate detection, feature extraction, and SVM classification) for barcode detection and recognition.	Inventory management with improved path planning and reduced power consumption	Experiment performance results of 2D barcode images demonstrating a precision of 98.08% and a recall of 98.27% within a fused feature ROC curve.
Academic [150]	-	QR	It combines sub-GHz wireless technology for IoT-standardized long-range wireless communication backbone and UWB for localization. Plug-and-play capabilities.	Autonomous navigation in a warehouse mock-up, tracking of runners in sport halls.	Theoretical evaluation in terms of update rate, energy consumption, maximum communication range, localization accuracy and scalability. Open source code of the UWB hardware and MAC protocol software available online.
Academic [151]	Phantom 2 vision DJI (weight 1242 g, maximum speed 15 m/s and up to 700 m)	RFID	Drone with a Windows CE 5.0 portable PDA (AT-880) that acts as a UHF RFID reader moves around an open storage yard	Inventory	No performance experiments
Academic [152]	DraganFly radio-controlled helicopters 82×82 cm, with an average flight time of 12 min. and RFID readers attached.	RFID (EPC)	Three-dimensional graphical simulator framework has been designed using Microsoft XNA	Inventory	Coordinated distribution of the UAVs
Academic [153]	UAVs and UGVs with LIDARs	Barcodes, AR markers	Cooperative work using vision techniques. The UAC acts as a mobile scanner, whereas the UGV acts as a carrying platform and as an AR ground reference for the indoor flight.	Inventory	-

4.3.10. Deep Learning Subsystem

In Table 2 most of the analyzed systems make use of convolution neural networks. This type of neural network and other deep learning techniques, which are commonly used for avoiding collisions and detecting obstacles, were previously discussed in detail in Section 3.

5. Main Challenges and Technical Limitations

5.1. DL Challenges

Given the nature of the DL methods that extract complex hierarchical abstractions from the data used throughout the training stages, the effectiveness and capability of generalization is bounded by the quantity and quality of the used data. Therefore, most of the shortcomings of the DL models are due to the data used for the learning processes:

- A limiting factor of the supervised approaches is the large amount of data required to generate robust models with generalization capability in different real-world environments. In order to minimize annotation effort, self-supervised algorithms allow automating the collection and annotation processes to generate large-scale datasets, the results of which are bounded by the strategy for generating labeled data.
- The diversity of the datasets in terms of representative scenarios and conditions, the variety of sensors, or the balance between the different classes are also conditioning factors for learning processes.
- The trial and error nature of RL raises safety concerns and suffers from crashes operating in real-world environments [48]. An alternative is the use of synthetic data or virtual environments for generating the training datasets.
- The gap between real and virtual environments limits the applicability of the simulation policies in the physical world. The development of more realistic virtual datasets is still an open issue.
- The accuracy of the spatial and temporal alignment between different sensors in a multimodal robotic system impacts data quality.

5.2. Other Challenges

As it can be observed after reviewing the different aspects of the DL-UAV hardware and the proposed communications architecture, it is possible to highlight several additional shortcomings:

- Most current DL-UAV architectures rely on a remote cloud. This solution does not fulfill the requirements of IoT applications in terms of cost, coverage, availability, latency, power consumption, and scalability. Furthermore, a cloud may be compromised by cyberattacks. To confront this challenge, the fog/edge computing and blockchain architecture analyzed in Section 4.2 can help to comply with the strict requirements of IoT applications.
- Flight time is related directly to the drone weight and its maximum payload. Moreover, a trade-off among cost, payload capacity, and reliability should be achieved when choosing between single and multirotor UAVs (as of writing, quardrotors are the preferred solution for DL-UAVs).
- Considering the computational complexity of most DL techniques, the controller hardware must be much more powerful than traditional IoT nodes and UAV controllers, so such a hardware needs to be enhanced to provide a good trade-off between performance and power consumption. In fact, some of the analyzed DL-UAVs have been already enhanced, some of which even have the ability to run certain versions of Linux and embedding dedicated GPUs. The improvements to be performed can affect both the hardware to control the flight and the one aimed at interacting with the other UAV subsystems.
- Although some flight controllers embed specific hardware, most of the analyzed DL-UAVs share similar sensors, mainly differing in the photo and video camera features. Future

DL-UAV developers should consider complementing visual identification techniques with other identification technologies (e.g., UWB and RFID) in order to improve their accuracy.

- Regarding the ways for powering a UAV, most DL-UAVs are limited by the use of heavy Li-Po or Li-ion batteries, but future developers should analyze the use of energy harvesting mechanisms to extend UAV flight autonomy.
- Robustness against cyberattacks is a key challenge. In order to ensure a secure operation, challenges such as interference management, mobility management or cooperative data transmission have to be taken into account. For instance, in [154], the authors provide a summary of the main wireless and security challenges, and introduce different AI-based solutions for addressing them. DL can be leveraged to classify various security events and alerts. For example, in the work by the authors of [155], the ML and DL methods are reviewed for network analysis of intrusion detection. Nonetheless, there are few examples in the literature that deal with cybersecure DL-UAV systems. One such work is by the authors of [156], where an RNN-based abnormal behavior detection scheme for UAVs is presented. Another relevant work is by the authors of [157], which details a DL framework for reconstructing the missing data in remote sensing analysis. Other authors focused on studying drone detection and identification methods using DL techniques [158]. In such a paper, the proposed algorithms exploit the unique acoustic fingerprints of UAVs to detect and identify them. In addition, a comparison on the performance of different neural networks (e.g., CNN, RNN, and Convolutional Recurrent Neural Network (CRNN)) tested with different audio samples is also presented. In the case of DL-UAV systems, the integrity of the classification is of paramount importance (e.g., to avoid adversarial samples [159] or inputs that deliberately result in an incorrect output classification). Adversarial samples are created at test time and do not require any alteration of the training process. The defense against the so-called adversarial machine learning [160] focuses on hardening the training phase of the DL algorithm. Finally, it is worth pointing out that cyber-physical attacks (e.g., spoofing attacks [161] and signal jamming) and authentication issues (e.g., forging of the identities of the transmitting UAVs or sending disrupted data using their identities) remain open research topics.

6. Conclusions

This article reviewed the latest advances on the development of autonomous IoT UAV systems controlled by DL techniques. After analyzing the problems that arise when UAVs need to detect and avoid obstacles, the paper presented a thorough survey on the state-of-the-art of DL techniques for autonomous obstacle detection and collision avoidance. In addition, the most relevant datasets and DL-UAV applications were detailed together with the required hardware. In relation to such hardware, the most common and the newest DL-UAV communications architectures were described, with special emphasis on the different subsystems of the architecture and the few academic developments that have proposed DL-UAV systems. Finally, the most relevant open challenges for current DL-UAVs were enumerated, thus defining a clear roadmap for the future DL-UAV developers that will create the next generation of affordable autonomous UAV IoT solutions.

Author Contributions: P.F.-L., L.R., V.M.-G., and T.M.F.-C. contributed to the overall study design, data collection and analysis, and writing of the manuscript.

Funding: This work has been funded by the Xunta de Galicia (ED431C 2016-045, ED431C 2016-047, ED431G/01), Centro singular de investigación de Galicia accreditation 2016-2019 through the CONCORDANCE (Collision avOidaNCe fOR UAVs using Deep leArNing teChniques and optimized sEnsor design) Project (PC18/01), the Agencia Estatal de Investigación of Spain (TEC2016-75067-C4-1-R) and ERDF funds of the EU (AEI/FEDER, UE).

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

Abbreviations

The following abbreviations are used in this manuscript.

COTS	Commercial Off-The-Shelf
CNN	Convolutional Neural Network
CRNN	Convolutional Recurrent Neural Network
DL	Deep Learning
DL-UAV	Deep Learning Unmanned Aerial Vehicle
DNN	Deep Neural Network
DoF	Degrees of Freedom
IMU	Inertial Measurement Unit
IoT	Internet of Things
LIDAR	Light Detection and Ranging
LSTM	Long Short-Term Memory
ML	Machine Learning
RFID	Radio-Frequency IDentification
RL	Reinforcement Learning
RSSI	Received Signal Strength Indicator
RNN	Recurrent Neural Network
SLAM	Simultaneous Localization and Mapping
SOC	System-on-Chip
UAV	Unmanned Aerial Vehicle
UHF	Ultra High Frequency
UAS	Unmanned Aerial System
VGA	Video Graphics Array

References

1. IHS. Internet of Things (IoT) Connected Devices Installed Base Worldwide from 2015 to 2025 (In Billions). Available online: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ (accessed on 5 September 2019).
2. Pérez-Expósito, J.; Fernández-Caramés, T.M.; Fraga-Lamas, P.; Castedo, L. VineSens: An Eco-Smart Decision-Support Viticulture System. *Sensors* **2017**, *17*, 465. [CrossRef] [PubMed]
3. Hernández-Rojas, D.; Fernández-Caramés, T.M.; Fraga-Lamas, P.; Escudero, C. A Plug-and-Play Human-Centered Virtual TEDS Architecture for the Web of Things. *Sensors* **2018**, *18*, 2052. [CrossRef] [PubMed]
4. Fernández-Caramés, T.M.; Fraga-Lamas, P. A Review on Human-Centered IoT-Connected Smart Labels for the Industry 4.0. *IEEE Access* **2018**, *6*, 25939–25957. [CrossRef]
5. Garrido-Hidalgo, C.; Hortelano, D.; Roda-Sanchez, L.; Olivares, T.; Ruiz, M. C.; Lopez, V. IoT Heterogeneous Mesh Network Deployment for Human-in-the-Loop Challenges Towards a Social and Sustainable Industry 4.0. *IEEE Access* **2018**, *6*, 28417–28437. [CrossRef]
6. Blanco-Novoa, O.; Fernández-Caramés, T.M.; Fraga-Lamas, P.; Castedo, L. A Cost-Effective IoT System for Monitoring Indoor Radon Gas Concentration. *Sensors* **2018**, *18*, 2198. [CrossRef] [PubMed]
7. Fernández-Caramés, T.M.; Fraga-Lamas, P.; Suárez-Albela, M.; Castedo, L. A methodology for evaluating security in commercial RFID systems. In *Radio Frequency Identification*; Crepaldi, P.C., Pimenta, T.C. Eds.; IntechOpen: London, UK, 2017.
8. Burg, A.; Chattopadhyay, A.; Lam, K. Wireless Communication and Security Issues for Cyber-Physical Systems and the Internet of Things. *Proc. IEEE* **2018**, *106*, 38–60. [CrossRef]
9. Hayat, S.; Yanmaz, E.; Muzaffar, R. Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint. *IEEE Commun. Surv. Tutorials* **2016**, *18*, 2624–2661. [CrossRef]
10. Shakhatreh, H.; Sawalmeh, A.; Al-Fuqaha, A.I.; Dou, Z.; Almaita, E.; Khalil, I.M.; Othman, N.S.; Khreishah, A.; Guizani, M. Unmanned Aerial Vehicles: A Survey on Civil Applications and Key Research Challenges. *arXiv* **2018**, arXiv:1805.00881.

11. Hassanalian, M.; Abdelkefi, A. Classifications, applications, and design challenges of drones: A review. *Prog. Aerosp. Sci.* **2017**, *91*, 99–131. [CrossRef]
12. Lu, J.; Xu, X.; Li, X.; Li, L.; Chang, C.-C.; Feng, X.; Zhang, S. Detection of Bird's Nest in High Power Lines in the Vicinity of Remote Campus Based on Combination Features and Cascade Classifier. *IEEE Access* **2018**, *6*, 39063–39071. [CrossRef]
13. Zhou, Z.; Zhang, C.; Xu, C.; Xiong, F.; Zhang, Y.; Umer, T. Energy-Efficient Industrial Internet of UAVs for Power Line Inspection in Smart Grid. *IEEE Trans. Ind. Inform.* **2018**, *14*, 2705–2714. [CrossRef]
14. Lim, G.J.; Kim, S.; Cho, J.; Gong, Y.; Khodaei, A. Multi-UAV Pre-Positioning and Routing for Power Network Damage Assessment. *IEEE Trans. Smart Grid* **2018**, *9*, 3643–3651. [CrossRef]
15. Wang, L.; Zhang, Z. Automatic Detection of Wind Turbine Blade Surface Cracks Based on UAV-Taken Images. *IEEE Trans. Ind. Electron.* **2017**, *64*, 7293–7303. [CrossRef]
16. Peng, K.; Liu, W.; Sun, Q.; Ma, X.; Hu, M.; Wang, D.; Liu, J. Wide-Area Vehicle-Drone Cooperative Sensing: Opportunities and Approaches. *IEEE Access* **2018**, *7*, 1818–1828. [CrossRef]
17. Rossi, M.; Brunelli, D. Autonomous Gas Detection and Mapping With Unmanned Aerial Vehicles. *IEEE Trans. Instrum. Meas.* **2016**, *65*, 765–775. [CrossRef]
18. Misra, P.; Kumar, A.A.; Mohapatra, P.; Balamuralidhar, P. Aerial Drones with Location-Sensitive Ears. *IEEE Commun. Mag.* **2018**, *56*, 154–160. [CrossRef]
19. Olivares, V.; Córdova, F. Evaluation by computer simulation of the operation of a fleet of drones for transporting materials in a manufacturing plant of plastic products. In Proceedings of the 2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Santiago, Chile, 28–30 October 2015; pp. 847–853.
20. Zhao, S.; Hu, Z.; Yin, M.; Ang, K.Z.Y.; Liu, P.; Wang, F.; Dong, X.; Lin, F.; Chen, B.M.; Lee, T.H. A Robust Real-Time Vision System for Autonomous Cargo Transfer by an Unmanned Helicopter. *IEEE Trans. Ind. Electron.* **2015**, *62*, 1210–1219. [CrossRef]
21. Kuru, K.; Ansell, D.; Khan, W.; Yetgin, H. Analysis and Optimization of Unmanned Aerial Vehicle Swarms in Logistics: An Intelligent Delivery Platform. *IEEE Access* **2019**, *7*, 15804–15831. [CrossRef]
22. Li, H.; Savkin, A.V. Wireless Sensor Network Based Navigation of Micro Flying Robots in the Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3524–3533. [CrossRef]
23. Kendoul, F. Survey of Advances in Guidance, Navigation, and Control of Unmanned Rotorcraft Systems. *J. Field Robot* **2012**, *29*, 315–378. [CrossRef]
24. Mur-Artal, R.; Montiel, J.M.M.; Tardós, J.D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. Robot.* **2015**, *31*, 1417–1463. [CrossRef]
25. Durrant-Whyte, H.; Bailey, T. Simultaneous localization and mapping: Part I. *IEEE Robot. Autom. Mag.* **2006**, *13*, 99–110. [CrossRef]
26. Durrant-Whyte, H.; Bailey, T. Simultaneous localization and mapping: Part II. *IEEE Robot. Autom. Mag.* **2006**, *13*, 108–117. [CrossRef]
27. Schönberger, J.L.; Frahm, J. Structure-from-Motion Revisited. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4104–4113.
28. Gageik, N.; Benz, P.; Montenegro, S. Obstacle Detection and Collision Avoidance for a UAV with Complementary Low-Cost Sensors. *IEEE Access* **2015**, *3*, 599–609. [CrossRef]
29. Krämer, M.S.; Kuhnert, K.D.. Multi-Sensor Fusion for UAV Collision Avoidance. In Proceedings of the 2018 2Nd International Conference on Mechatronics Systems and Control Engineering, Amsterdam, The Netherlands, 21–23 February 2018; ACM: New York, NY, USA, 2018; pp. 5–12.
30. Bonin-Font, F.; Ortiz, A.; Oliver, G. Visual Navigation for Mobile Robots: A Survey. *J. Intell. Robot. Syst.* **2008**, *53*, 263. [CrossRef]
31. Lu, Y.; Xue, Z.; Xia, G.S.; Zhang, L. A survey on vision-based UAV navigation. *Geo-Spat. Inf. Sci.* **2018**, *21*, 21–32. [CrossRef]
32. Mukhtar, A.; Xia, L.; Tang, T.B. Vehicle detection techniques for collision avoidance systems: A review. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2318–2338. [CrossRef]
33. Rybus, T. Obstacle avoidance in space robotics: Review of major challenges and proposed solutions. *Prog. Aerosp. Sci.* **2018**, *101*, 31–48. [CrossRef]
34. Campbell, S.; Naeem, W.; Irwin, G. A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres. *Annu. Rev. Control.* **2012**, *36*, 267–283. [CrossRef]

35. Shabbir, J.; Anwer, T. A Survey of Deep Learning Techniques for Mobile Robot Applications. *arXiv* **2018**, arXiv:1803.07608.
36. Whitehead, K.; Hugenholtz, C.H. Remote sensing of the environment with small unmanned aircraft systems (UASs), part 1: A review of progress and challenges. *J. Unmanned Veh. Syst.* **2014**, *2*, 69–85. [CrossRef]
37. Lippitt, C.D.; Zhang, S. The impact of small unmanned airborne platforms on passive optical remote sensing: A conceptual perspective. *J. Remote. Sens.* **2018**, *39*, 4852–4868. [CrossRef]
38. Yao, H.; Qin, R.; Chen, X. Unmanned aerial vehicle for remote sensing applications—A review. *Remote Sens.* **2019**, *11*, 1443. [CrossRef]
39. Carrio, A.; Sampedro, C.; Rodriguez-Ramos, A.; Cervera, P.C. A Review of Deep Learning Methods and Applications for Unmanned Aerial Vehicles. *J. Sens.* **2017**, *2017*. [CrossRef]
40. Ammour, N.; Alhichri, H.; Bazi, Y.; Benjdira, B.; Alajlan, N.; Zuair, M. Deep Learning Approach for Car Detection in UAV Imagery. *Remote Sens.* **2017**, *9*, 312. [CrossRef]
41. Zhao, Y.; Ma, J.; Li, X.; Zhang, J. Saliency Detection and Deep Learning-Based Wildfire Identification in UAV Imagery. *Sensors* **2018**, *18*, 712. [CrossRef] [PubMed]
42. Zeggada, A.; Melgani, F.; Bazi, Y. A Deep Learning Approach to UAV Image Multilabeling. *IEEE Geosci. Remote. Sens. Lett.* **2017**, *14*, 694–698. [CrossRef]
43. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
44. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature* **2016**, *529*, 484–489. [CrossRef]
45. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *521*, 1097–1105. [CrossRef]
46. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Cham, Switzerland, 2015; pp. 234–241.
47. Deng, L.; Hinton, G.; Kingsbury, B. New types of deep neural network learning for speech recognition and related applications: An overview. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013.
48. Kouris, A.; Bouganis, C. Learning to Fly by MySelf: A Self-Supervised CNN-Based Approach for Autonomous Navigation. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018.
49. Gandhi, D.; Pinto, L.; Gupta, A. Learning to fly by crashing. *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)* **2017**, *2017*, 3948–3955.
50. Gandhi, D.; Pinto, L.; Gupta, A. Ram Prasad, P.; Sachin, V.; Shahzad, A.; Suman Kumar, C.; Pankaj Kumar, S. Deep Neural Network for Autonomous UAV Navigation in Indoor Corridor Environments. *Procedia Comput. Sci.* **2018**, *133*, 643–650.
51. Michels, J.; Saxena, A.; Ng, A.Y. High Speed Obstacle Avoidance Using Monocular Vision and Reinforcement Learning. In Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany, 7–11 August 2005; pp. 593–600.
52. Tai, L.; Li, S.; Liu, M. A deep-network solution towards model-less obstacle avoidance. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 2759–2764.
53. Singla, A.; Padakandla, S.; Bhatnagar, S. Memory-based Deep Reinforcement Learning for Obstacle Avoidance in UAV with Limited Environment Knowledge. *arXiv* **2018**, arXiv:1811.03307.
54. Xie, L.; Wang, S.; Markham, A.; Trigoni, N. Towards Monocular Vision based Obstacle Avoidance through Deep Reinforcement Learning. *arXiv* **2017**, arXiv:1706.09829.
55. Smolyanskiy, N.; Kamenev, A.; Smith, J.; Birchfield, S. Toward low-flying autonomous MAV trail navigation using deep neural networks for environmental awareness. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 4241–4247.
56. Kearns, M.; Singh, S. Finite-sample Convergence Rates for Q-learning and Indirect Algorithms. In *Proceedings of the 1998 conference on Advances in Neural Information Processing Systems II*; MIT Press: Cambridge, MA, USA, 1999; pp. 996–1002.

57. Chen, Y.F.; Liu, M.; Everett, M.; How, J.P. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3285–3292.
58. Tai, L.; Paolo, G.; Liu, M. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 31–36.
59. Sadeghi, F.; Sadeghi, F.; Levine, S. CAD2RL: Real Single-Image Flight Without a Single Real Image. *arXiv* **2016**, arXiv:1611.04201.
60. Shah, S.; Kapoor, A.; Dey, D.; Lovett, C. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. In *Field and Service Robotics*; Springer: Cham, Switzerland, 2018; pp. 621–635.
61. Pan, S.J.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [CrossRef]
62. Loquercio, A.; Maqueda, A.I.; del-Blanco, C.R.; Scaramuzza, D. DroNet: Learning to Fly by Driving. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1088–1095. [CrossRef]
63. Tobin, J.; Fong, R.H.; Ray, A.; Schneider, J.; Zaremba, W.; Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 23–30.
64. Varol, G.; Romero, J.; Martin, X.; Mahmood, N.; Black, M.J.; Laptev, I. Learning from synthetic humans. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4627–4635.
65. Atapour-Abarghouei, A., Breckon, T.P. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2800–2810.
66. Mikolov, T.; Karafiát, M.; Burget, L.; Cernocký, J.; Khudanpur, S. Recurrent neural network based language model. In Proceedings of the INTERSPEECH 2010 11th Annual Conference of the International Speech Communication AssociationMakuhari, Chiba, Japan, 26–30 September 2010; pp. 1045–1048.
67. Pei, W.; Baltrušaitis, T.; Tax, D.M.J.; Morency, L. Temporal Attention-Gated Model for Robust Sequence Classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 820–829.
68. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
69. Kim, D.K.; Chen, T. Deep Neural Network for Real-Time Autonomous Indoor Navigation. Available online: https://arxiv.org/abs/1511.04668 (accessed on 31 July 2019).
70. Jung, S.; Hwang, S.; Shin, H.; Him, D.H. Perception, Guidance, and Navigation for Indoor Autonomous Drone Racing Using Deep Learning. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2539–2544. [CrossRef]
71. Dionisio-Ortega, S.; Rojas-Perez, L.O.; Martinez-Carranza, J.; Cruz-Vega, I. A deep learning approach towards autonomous flight in forest environments. In Proceedings of the 2018 International Conference on Electronics, Communications and Computers (CONIELECOMP), Cholula, Mexico, 21–23 February 2018; pp. 139–144.
72. Chen, P.; Lee, C. UAVNet: An Efficient Obstacel Detection Model for UAV with Autonomous Flight. In Proceedings of the 2018 International Conference on Intelligent Autonomous Systems (ICoIAS), Singapore, 1–3 March 2018; pp. 217–220.
73. Khan, A.; Hebert, M. Learning safe recovery trajectories with deep neural networks for unmanned aerial vehicles. In Proceedings of the 2018 IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2018.
74. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269.
75. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
76. Giusti, A.; Guzzi, J.; CireÅŸan, D.C.; He, F.; RodrÃguez, J.P.; Fontana, F.; Faessler, M.; Forster, C.; Schmidhuber, J.; Caro, G.D.; et al. A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots. *IEEE Robot. Autom. Lett.* **2016**, *1*, 661–667. [CrossRef]

77. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 Year, 1000 km: The Oxford RobotCar Dataset. *Int. J. Robot. Res. (IJRR)* **2017**, *36*, 3–15. [CrossRef]
78. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3234–3243.
79. Li, B.; Dai, Y.; He, M. Monocular depth estimation with hierarchical fusion of dilated CNNs and soft-weighted-sum inference. *Pattern Recognit.* **2018**, *83*, 328–339. [CrossRef]
80. Suárez-Albela, M.; Fernández-Caramés, T.; Fraga-Lamas, P.; Castedo, L. A Practical Evaluation of a High-Security Energy-Efficient Gateway for IoT Fog Computing Applications. *Sensors* **2017**, *17*, 1978. [CrossRef] [PubMed]
81. Raspberry Pi Official Webpage. Available online: https://www.raspberrypi.org/ (accessed on 31 July 2019).
82. Beagle Bone Official Webpage. Available online: https://beagleboard.org/black (accessed on 31 July 2019).
83. Orange Pi Official Webpage. Available online: http://www.orangepi.org/ (accessed on 31 July 2019).
84. Fernández-Caramés, T.M.; Fraga-Lamas, P.; Suárez-Albela, M.; Vilar-Montesinos, M. A Fog Computing and Cloudlet Based Augmented Reality System for the Industry 4.0 Shipyard. *Sensors* **2018**, *18*, 1798. [CrossRef] [PubMed]
85. Fernández-Caramés, T.M.; Fraga-Lamas, P. A Review on the Use of Blockchain for the Internet of Things. *IEEE Access* **2018**, *6*, 32979–33001. [CrossRef]
86. Fernández-Caramés, T.M..; Froiz-Míguez, I.; Blanco-Novoa, O.; Fraga-Lamas, P. Enabling the Internet of Mobile Crowdsourcing Health Things: A Mobile Fog Computing, Blockchain and IoT Based Continuous Glucose Monitoring System for Diabetes Mellitus Research and Care. *Sensors* **2019**, *19*, 3319. [CrossRef] [PubMed]
87. Huang, J.; Kong, L.; Chen, G.; Wu, M.; Liu, X.; Zeng, P. Towards Secure Industrial IoT: Blockchain System With Credit-Based Consensus Mechanism. *IEEE Trans. Ind. Inf.* **2019**, *15*, 3680–3689. [CrossRef]
88. Fernández-Caramés, T.M.; Blanco-Novoa, O.; Froiz-Míguez, I.; Fraga-Lamas, P. Towards an Autonomous Industry 4.0 Warehouse: A UAV and Blockchain-Based System for Inventory and Traceability Applications in Big Data-Driven Supply Chain Management. *Sensors* **2019**, *19*, 2394. [CrossRef]
89. Zyskind, G.; Nathan, O.; Pentland, A.S. Decentralizing Privacy: Using Blockchain to Protect Personal Data. In Proceedings of the 2015 IEEE Security and Privacy Workshops, San Jose, CA, USA, 21–22 May 2015; pp. 180–184.
90. Christidis, K.; Devetsikiotis, M. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access* **2016**, *4*, 2292–2303. [CrossRef]
91. Parrot's SDK APIs. Available online: https://developer.parrot.com (accessed on 31 July 2019)
92. Broecker, B.; Tuyls, K.; Butterworth, J. Distance-Based Multi-Robot Coordination on Pocket Drones. In Proceedings of 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 6389–6394.
93. Anwar, M.A.; Raychowdhury, A. NavREn-Rl: Learning to fly in real environment via end-to-end deep reinforcement learning using monocular images. In Proceedings of the 2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Stuttgart, Germany, 20–22 November 2018.
94. Lee, D.; Lee, H.; Lee, J.; Shim, D.H. Design, implementation, and flight tests of a feedback linearization controller for multirotor UAVs. *Int. J. Aeronaut. Space Sci.* **2017**, *18*, 112–121. [CrossRef]
95. Junoh, S.; Aouf, N. Person classification leveraging Convolutional Neural Network for obstacle avoidance via Unmanned Aerial Vehicles. In Proceedings of the 2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS), Linkoping, Sweden, 3–5 October 2017; pp. 168–173.
96. Carrio, A.; Vemprala, S.; Ripoll, A.; Saripalli, S.; Campoy, P. Drone Detection Using Depth Maps. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1034–1037.
97. Pixhawk Official Webpage. Available online: http://pixhawk.org (accessed on 31 July 2019)
98. Guzmán-Rabasa, J.A.; López-Estrada, F.R.; González-Contreras, B.M.; Valencia-Palomo, G.; Chadli, M.; Pérez-Patricio, M. Actuator fault detection and isolation on a quadrotor unmanned aerial vehicle modeled as a linear parameter-varying system. *Meas. Control* **2019**. [CrossRef]
99. Guo, D.; Zhong, M.; Ji, H.; Liu, Y.; Yang, R. A hybrid feature model and deep learning based fault diagnosis for unmanned aerial vehicle sensors. *Neurocomputing* **2018**, *319*, 155–163. [CrossRef]

100. Mai, V.; Kamel, M.; Krebs, M.; Schaffner, A.; Meier, D.; Paull, L.; Siegwart, R. Local Positioning System Using UWB Range Measurements for an Unmanned Blimp. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2971–2978. [CrossRef]
101. He, S.; Chan, S.H.G. Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 466–490. [CrossRef]
102. Fraga-Lamas, P.; Fernández-Caramés, T.M.; Noceda-Davila, D.; Vilar-Montesinos, M. RSS Stabilization Techniques for a Real-Time Passive UHF RFID Pipe Monitoring System for Smart Shipyards. In Proceedings of the 2017 IEEE International Conference on RFID (IEEE RFID 2017), Phoenix, AZ, USA, 9–11 May 2017; pp. 161–166.
103. Yan, J.; Zhao, H.; Luo, X.; Chen, C.; Guan, X. RSSI-Based Heading Control for Robust Long-Range Aerial Communication in UAV Networks. *IEEE Internet Things J.* **2019**, *6*, 1675–1689. [CrossRef]
104. Wang, W.; Bai, P.; Zhou, Y.; Liang, X.; Wang, Y. Optimal Configuration Analysis of AOA Localization and Optimal Heading Angles Generation Method for UAV Swarms. *IEEE Access* **2019**, *7*, 70117–70129. [CrossRef]
105. Okello, N.; Fletcher, F.; Musicki, D.; Ristic, B. Comparison of Recursive Algorithms for Emitter Localisation using TDOA Measurements from a Pair of UAVs. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 1723–1732. [CrossRef]
106. Li, B.; Fei, Z.; Zhang, Y. UAV Communications for 5G and Beyond: Recent Advances and Future Trends. *IEEE Internet Things J.* **2019**, *6*, 2241–2263. [CrossRef]
107. Zhang, L.; Zhao, H.; Hou, S.; Zhao, Z.; Xu, H.; Wu, X.; Wu, Q.; Zhang, R. A Survey on 5G Millimeter Wave Communications for UAV-Assisted Wireless Networks. *IEEE Access* **2019**, *7*, 117460–117504. [CrossRef]
108. Hristov, G.; Raychev, J.; Kinaneva, D.; Zahariev, P. Emerging Methods for Early Detection of Forest Fires Using Unmanned Aerial Vehicles and Lorawan Sensor Networks. In Proceedings of the 2018 28th EAEEIE Annual Conference (EAEEIE), Reykjavik, Iceland, 26–28 September 2018.
109. Nijsure, Y.A.; Kaddoum, G.; Khaddaj Mallat, N.; Gagnon, G.; Gagnon, F. Cognitive Chaotic UWB-MIMO Detect-Avoid Radar for Autonomous UAV Navigation. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3121–3131. [CrossRef]
110. Ali, M.Z.; Misic, J.; Misic, V.B. Extending the Operational Range of UAV Communication Network using IEEE 802.11ah. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
111. Khan, M. A.; Hamila, R.; Kiranyaz, M. S.; Gabbouj, M. A Novel UAV-Aided Network Architecture Using Wi-Fi Direct. *IEEE Access* **2019**, *7*, 67305–67318. [CrossRef]
112. Bluetooth 5 Official Webpage. Available online: https://www.bluetooth.com/bluetooth-resources/bluetooth-5-go-faster-go-further/ (accessed on 31 July 2019).
113. Dae-Ki,C.; Chia-Wei, C.; Min-Hsieh, T.; Gerla, M. Networked medical monitoring in the battlefield. In Proceedings of the 2008 IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 16–19 November 2008; pp. 1–7.
114. IQRF Official Webpage. Available online: https://www.iqrf.org/ (accessed on 31 July 2019).
115. 3GPP NB-IoT Official Webpage. Available online: https://www.3gpp.org/news-events/1785-nb_iot_complete (accessed on 31 July 2019).
116. LoRa-Alliance. LoRaWAN What is it? In *A Technical Overview of LoRa and LoRaWAN*; White Paper; The LoRa Alliance: San Ramon, CA, USA, 2015.
117. SigFox Official Web Page. Available online: https://www.sigfox.com (accessed on 31 July 2019).
118. Valisetty, R.; Haynes, R.; Namburu, R.; Lee, M. Machine Learning for US Army UAVs Sustainment: Assessing Effect of Sensor Frequency and Placement on Damage Information in the Ultrasound Signals. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; pp. 165–172.
119. Weightless Official Web Page. Available online: http://www.weightless.org (accessed on 31 July 2019).
120. Wi-Fi Hallow Official Webpage. Available online: https://www.wi-fi.org/discover-wi-fi/wi-fi-halow (accessed on 31 July 2019).
121. Wi-Sun Alliance Official Web Page. Available online: https://www.wi-sun.org (accessed on 31 July 2019).
122. ZigBee Alliance. Available online: http://www.zigbee.org (accessed on 31 July 2019).
123. Jawhar, I.; Mohamed, N.; Al-Jaroodi, J. UAV-based data communication in wireless sensor networks: Models and strategies. In Proceedings of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA, 9–12 June 2015; pp. 687–694.

124. Wu, J.; Ma, J.; Rou, Y.; Zhao, L.; Ahmad, R. An Energy-Aware Transmission Target Selection Mechanism for UAV Networking. *IEEE Access* **2019**, *7*, 67367–67379. [CrossRef]
125. Say, S.; Inata, H.; Liu, J.; Shimamoto, S. Priority-Based Data Gathering Framework in UAV-Assisted Wireless Sensor Networks. *IEEE Sens. J.* **2016**, *16*, 5785–5794. [CrossRef]
126. Donateo, T.; Ficarella, A.; Spedicato, L.; Arista, A.; Ferraro, M. A new approach to calculating endurance in electric flight and comparing fuel cells and batteries. *Appl. Energy* **2017**, *187*, 807–819. [CrossRef]
127. Ji, B.; Li, Y.; Zhou, B.; Li, C.; Song, K.; Wen, H. Performance Analysis of UAV Relay Assisted IoT Communication Network Enhanced With Energy Harvesting. *IEEE Access* **2019**, *7*, 38738–38747. [CrossRef]
128. Gong, A.; MacNeill, R.; Verstraete, D.; Palmer, J.L. Analysis of a Fuel-Cell/Battery /Supercapacitor Hybrid Propulsion System for a UAV Using a Hardware-in-the-Loop Flight Simulator. In Proceedings of the 2018 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS), Cincinnati, OH, USA, 12–14 July 2018; pp. 1–17.
129. Fleming, J.; Ng, W.; Ghamaty, S. Thermoelectric-based power system for UAV/MAV applications. In Proceedings of the 1st UAV Conference, Portsmouth, Virginia, 20–23 May 2002; p. 3412.
130. Martinez, V.; Defaÿ, F.; Salvetat, L.; Neuhaus, K.; Bressan, M.; Alonso, C.; Boitier, V. Study of Photovoltaic Cells Implantation in a Long-Endurance Airplane Drone. In Proceedings of the 2018 7th International Conference on Renewable Energy Research and Applications (ICRERA), Paris, France, 14–17 October 2018; pp. 1299–1303.
131. Khristenko, A.V.; Konovalenko, M.O.; Rovkin, M.E.; Khlusov, V.A.; Marchenko, A.V.; Sutulin, A.A.; Malyutin, N.D. Magnitude and Spectrum of Electromagnetic Wave Scattered by Small Quadcopter in *X*-Band. *IEEE Trans. Antennas Propag.* **2018**, *66*, 1977–1984. [CrossRef]
132. Xu, J.; Zeng, Y.; Zhang, R. UAV-Enabled Wireless Power Transfer: Trajectory Design and Energy Optimization. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 5092–5106. [CrossRef]
133. Xie, L.; Xu, J.; Zhang, R. Throughput Maximization for UAV-Enabled Wireless Powered Communication Networks. *IEEE Internet Things J.* **2019**, *6*, 1690–1703. [CrossRef]
134. Ure, N.K.; Chowdhary, G.; Toksoz, T.; How, J.P.; Vavrina, M.A.; Vian, J. An Automated Battery Management System to Enable Persistent Missions With Multiple Aerial Vehicles. *IEEE/ASME Trans. Mech.* **2015**, *20*, 275–286. [CrossRef]
135. Amorosi, L.; Chiaraviglio, L.; Galán-Jiménez, J. Optimal Energy Management of UAV-Based Cellular Networks Powered by Solar Panels and Batteries: Formulation and Solutions. *IEEE Access* **2019**, *7*, 53698–53717. [CrossRef]
136. Huang, F.; Chen, J.; Wang, H.; Ding, G.; Xue, Z.; Yang, Y.; Song, F. UAV-Assisted SWIPT in Internet of Things With Power Splitting: Trajectory Design and Power Allocation. *IEEE Access* **2019**, *7*, 68260–68270. [CrossRef]
137. Belotti, M.; Božić, N.; Pujolle, G.; Secci, S. A Vademecum on Blockchain Technologies: When, Which and How. *IEEE Commun. Surv. Tutorials* **2019**, *6*, 1690–1703. [CrossRef]
138. Fraga-Lamas, P.; Fernández-Caramés, T.M. A Review on Blockchain Technologies for an Advanced and Cyber-Resilient Automotive Industry. *IEEE Access* **2019**, *7*, 17578–17598. [CrossRef]
139. Trump, B.D.; Florin, M.; Matthews, H.S.; Sicker, D.; Linkov, I. Governing the Use of Blockchain and Distributed Ledger Technologies: Not One-Size-Fits-All. *IEEE Eng. Manag. Rev.* **2018**, *46*, 56–62. [CrossRef]
140. Fernández-Caramés, T.M.; Fraga-Lamas, P. A Review on the Application of Blockchain for the Next Generation of Cybersecure Industry 4.0 Smart Factories. *IEEE Access* **2019**, *7*, 45201–45218. [CrossRef]
141. Liang, X.; Zhao, J.; Liang, X.; Zhao, J.; Shetty, S.; Liu, J.; Li, D. Integrating blockchain for data sharing and collaboration in mobile healthcare applications. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–5.
142. Dinh, P.; Nguyen, T. M.; Sharafeddine, S.; Assi, C. Joint Location and Beamforming Design for Cooperative UAVs with Limited Storage Capacity. *IEEE Trans. Commun.* **2019**. [CrossRef]
143. Gaynor, P.; Coore, D. Towards distributed wilderness search using a reliable distributed storage device built from a swarm of miniature UAVs. In Proceedings of the 2014 International Conference on Unmanned Aircraft Systems (ICUAS), Orlando, FL, USA, 27–30 May 2014; pp. 596–601.
144. Beul, M.; Droeschel, D.; Nieuwenhuisen, M.; Quenzel, J.; Houben, S.; Behnke, S. Fast Autonomous Flight in Warehouses for Inventory Applications. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3121–3128. [CrossRef]

145. Zhang, J.; Wang, X.; Yu, Z.; Lyu, Y.; Mao, S.; Periaswamy, S.C.; Patton, J.; Wang, X. Robust RFID Based 6-DoF Localization for Unmanned Aerial Vehicles. *IEEE Access* **2019**, *7*, 77348–77361. [CrossRef]
146. Hardis Group, EyeSee Official Webpage. Available online: http://www.eyesee-drone.com (accessed on 31 March 2019).
147. Geodis and Delta Drone Official Communication. Available online: www.goo.gl/gzeYV7 (accessed on 31 March 2019).
148. DroneScan Official Webpage. Available online: www.dronescan.co (accessed on 31 March 2019).
149. Cho, H.; Kim, D.; Park, J.; Roh, K.; Hwang, W. 2D Barcode Detection using Images for Drone-assisted Inventory Management. In Proceedings of the 15th International Conference on Ubiquitous Robots (UR), Honolulu, HI, USA, 26–30 June 2018.
150. Macoir, N.; Bauwens, J.; Jooris, B.; Van Herbruggen, B.; Rossey, J.; Hoebeke, J.; De Poorter, E. UWB Localization with Battery-Powered Wireless Backbone for Drone-Based Inventory Management. *Sensors* **2019**, *19*, 467. [CrossRef]
151. Bae, S.M.; Han, K.H.; Cha, C.N.; Lee, H.Y. Development of Inventory Checking System Based on UAV and RFID in Open Storage Yard. In Proceedings of the International Conference on Information Science and Security (ICISS), Pattaya, Thailand, 19–22 December 2016.
152. Ong, J.H.; Sanchez, A.; Williams, J. Multi-UAV System for Inventory Automation. In Proceedings of the 1st Annual RFID Eurasia, Istanbul, Turkey, 5–6 September 2007.
153. Harik, E.H.C.; Guérin, F.; Guinand, F.; Brethé, J.; Pelvillain, H. Towards An Autonomous Warehouse Inventory Scheme. In Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016.
154. Challita, U.; Ferdowsi, A.; Chen, M.; Saad, W. Machine Learning for Wireless Connectivity and Security of Cellular-Connected UAVs. *IEEE Wirel. Commun.* **2019**, *26*, 28–35. [CrossRef]
155. Xin, Y.; Kong, L.; Liu, Z.; Chen, Y.; Li, Y.; Zhu, H.; Gao, M.; Hou, H.; Wang, C. Machine Learning and Deep Learning Methods for Cybersecurity. *IEEE Access* **2018**, *6*, 35365–35381. [CrossRef]
156. Xiao, K.; Zhao, J.; He, Y.; Li, C.; Cheng, W. Abnormal Behavior Detection Scheme of UAV Using Recurrent Neural Networks. *IEEE Access* **2019**, *7*, 110293–110305. [CrossRef]
157. Das, M.; Ghosh, S.K. A deep-learning-based forecasting ensemble to predict missing data for remote sensing analysis. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2017**, *10*, 5228–5236. [CrossRef]
158. Al-Emadi, S.; Al-Ali, A.; Mohammad, A.; Al-Ali, A. Audio Based Drone Detection and Identification using Deep Learning. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; pp. 459–464.
159. Ball, J.E.; Anderson, D.T.; Chan, C.S.C. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *J. Appl. Remote. Sens.* **2017**, *11*, 042609. [CrossRef]
160. Papernot, N.; McDaniel, P.; Jha, S.; Fredrikson, M.; Celik, Z. B.; Swami, A. The limitations of deep learning in adversarial settings. In Proceedings of the 2016 IEEE European Symposium on Security and Privacy (EuroS&P), Saarbrucken, Germany, 21–24 March 2016; pp. 372–387.
161. Davidson, D.; Wu, H.; Jellinek, R.; Singh, V.; Ristenpart, T. Controlling UAVs with sensor input spoofing attacks. In Proceedings of the 10th USENIX Workshop on Offensive Technologies (WOOT 16), Austin, TX, USA, 8–9 August 2016.