

Article

Towards Fully Autonomous UAVs: A Survey

Taha Elmokadem and Andrey V. Savkin * 

School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney 2052, Australia; t.elmokadem@unsw.edu.au

* Correspondence: a.savkin@unsw.edu.au

Abstract: Unmanned Aerial Vehicles have undergone rapid developments in recent decades. This has made them very popular for various military and civilian applications allowing us to reach places that were previously hard to reach in addition to saving time and lives. A highly desirable direction when developing unmanned aerial vehicles is towards achieving fully autonomous missions and performing their dedicated tasks with minimum human interaction. Thus, this paper provides a survey of some of the recent developments in the field of unmanned aerial vehicles related to safe autonomous navigation, which is a very critical component in the whole system. A great part of this paper focus on advanced methods capable of producing three-dimensional avoidance maneuvers and safe trajectories. Research challenges related to unmanned aerial vehicle development are also highlighted.

Keywords: UAVs; aerial drones; flying robots; autonomous navigation; trajectory planning; obstacle avoidance; intelligent control; collaborative robots' control; aerial environmental monitoring; UAVs for smart agriculture



Citation: Elmokadem, T.; Savkin, A.V. Towards Fully Autonomous UAVs: A Survey. *Sensors* **2021**, *21*, 6223. <https://doi.org/10.3390/s21186223>

Academic Editors: Liviu C. Miclea, Ovidiu P. Stan, Vlad Muresan and Florin Pop

Received: 9 August 2021

Accepted: 9 September 2021

Published: 16 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Unmanned Aerial Vehicles (UAVs) have evolved greatly over recent decades with prevalent use in military and civilian applications such as search and rescue [1], wireless sensor networks and the Internet of Things (IoT) [2,3], remote sensing [4], surveillance and monitoring [5–7], 3D mapping [8], object grasping and aerial manipulation [9,10], underground mine exploration and tunnel inspection [11,12], etc. Challenges in developing UAVs keep increasing as the complexity of their tasks increases especially with the aim of moving towards fully autonomous operation (i.e., with minimum human interaction). Moreover, many applications require UAVs to autonomously operate in unknown and dynamic environments where they need to completely rely on onboard sensors to understand the environment they navigate in and to complete their tasks efficiently. The autonomous navigation problem can generally be defined as the vehicle's ability to reach a goal location while avoiding collisions with surroundings without human interaction. This is a very challenging problem as it is important to achieve safe navigation to avoid causing damage or injuries. Limitations on available technologies related to UAVs add more complexities to the development of autonomous navigation methods in order to ensure reliability and robustness compared with unmanned ground vehicles (UGVs) and autonomous underwater vehicles (AUVs). Examples of such are limitations on sensing capabilities, allowed payload capacity, flight time, energy consumption, communication, actuation and control effort. Developing efficient and advanced motion control methods plays a critical role in minimizing the effect of these factors. For example, adopting complex bio-inspired flying behaviors such as perching and maneuvering on surfaces can help extend mission flight time [13].

Many researchers have contributed towards addressing the navigation problem for UAVs. This overview aims at surveying the developments made in the past ten years towards achieving fully autonomous operations. Some key approaches developed earlier

than the considered time frame are also reported for the sake of completion. General definitions and research areas are also provided for new researchers interested in this field. Additionally, a list of useful open-source projects and tools is provided which may aid in quick development and deployment of new approaches related to UAVs as part of a complete autonomous stack.

This survey is dedicated to the more complex problem of three-dimensional (3D) obstacle avoidance utilizing the full maneuvering capabilities of UAVs. Given the fact that many of the existing algorithms are developed considering general 3D kinematic models, they are applicable to vehicles moving in 3D, including different UAV types and autonomous underwater vehicles (AUVs). Similarly, some of the general approaches developed for AUVs are also reported here given that they are applicable to UAVs. Planar approaches usually consider flights at a fixed altitude to simplify the obstacle avoidance problem. These approaches may fail with the increased complexity of the environments where UAVs are needed; hence, utilizing 3D avoidance maneuvers is more desirable. However, some planar approaches are also reported here where they can potentially inspire extensions to more general 3D methods.

This paper is organized as follows. A general overview of existing UAV types, classifications, autonomous navigation paradigms, and control structures is given in Section 2. Next, different motion planning and obstacle avoidance techniques are surveyed in Section 3. After that, Section 4 presents different control methods used for UAVs along with information about adopted dynamical models for different UAV types. Brief information about existing localization and mapping techniques is also provided in Section 5. Additionally, some useful open-source projects and tools for UAV development are provided in Section 7. Research challenges are then outlined in Section 8 along with some example applications where UAVs are used. Finally, concluding remarks are made in Section 9.

2. UAV Types, Autonomy and System Architectures

2.1. UAV Types

UAVs can be classified based on several factors such as size, mean takeoff weight, control configuration, autonomy level, etc. For example, classifications of UAVs based on size according to the Australian Civil Aviation Safety Authority (CASA) are:

- Micro: less than 250 g;
- Very Small: 0.25–2 kg;
- Small: 2–25 kg;
- Medium: 25–150 kg;
- Large: More than 150 kg.

Large UAVs are mainly used in tactical missions and military applications; for more detailed classifications related to military use, see [14]. Based on control configurations, UAVs can be categorized into (see Figure 1):

- single-rotor [15–18]: helicopter;
- multi-rotor [19–24]: tricopter, quadrotor, hexacopter, etc.;
- fixed-wing [25–27];
- hybrid [28–31];
- flapping wings [32–38]: Ornithopters and Entomopters.

Single-rotor aerial vehicles such as helicopters have not been utilized much as UAV platforms. Multi-rotors on the other hand have become the most popular choice in most civilian applications when it comes to maneuverability. Multi-rotors such as quadrotors, hexacopters and octocopters with fixed-pitch rotors share similar dynamical models for control. However, quadrotors are cheaper, faster, and highly maneuverable while hexacopters and octocopters can offer better flight stability, fault-tolerance, and more payload capacity. Multi-rotors with fixed-pitch rotors are underactuated systems where it is not

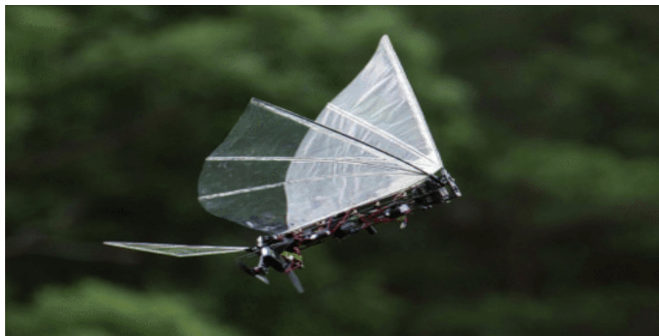
possible to completely control all degrees of freedom. There have been recent advances in developing omnidirectional tilt-rotor UAVs which are fully actuated in 6DOF [23,24,39,40].



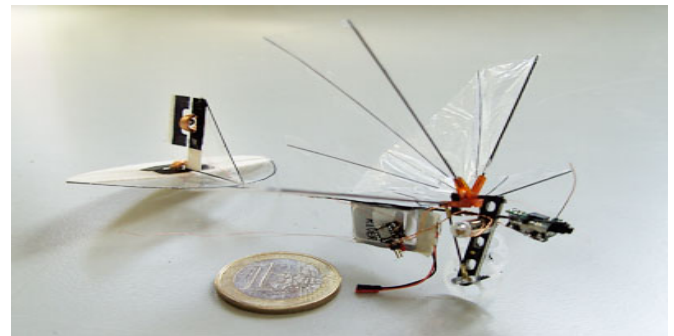
(a)



(b)



(c)



(d)

Figure 1. Different UAV types based on control configurations. (a) Multirotor (Hexacopter), (b) Fixed-Wing, (c) Ornithopter flapping-wing UAV (Robo Raven) [37], (d) Entomopter flapping-wing UAV (DelFly Micro).

Multi-rotors in general lie under the category of vertical-takeoff-and-landing (VTOL) vehicles with the ability to hover in place. On the contrary, fixed-wing UAVs are horizontal-takeoff-and-landing (HTOL) vehicles, and they cannot hover at a certain position due to nonholonomic constraints. Instead, they have to loiter around areas of interest. However, fixed-wing UAVs have advantages such as long endurance (i.e., flight time) and higher achievable speeds compared to multi-rotors. Hybrid UAVs combine both configurations of fixed wings and multiple rotors utilizing the advantages of both such as vertical takeoff and landing, hovering and long endurance flights. However, these vehicles are still under development, and more research is needed for reliable control, especially when switching between flight modes.

Another type of UAV is one with flapping wings inspired by birds (Ornithopters) and insects (Entomopters). This type is still under development due to its complex dynamics and anticipated power problems [32]. Recently, new bio-inspired hybrid unmanned vehicles have also been proposed to handle navigation in different domains such as underwater-aerial vehicles [41–43] and aerial-ground vehicles [44–48].

2.2. Autonomy Levels

Being completely able to carry out missions/tasks with minimum human interaction is an ultimate goal for unmanned aerial vehicles. Different levels of autonomy can be achieved towards that goal depending on the complexity of tasks and whether a fully autonomous solution exists or not for that specific application. These levels can be described

based on the UAV mode of operation according to the National Institute of Standards and Technology (NIST) as follows [49]:

- **Fully autonomous:** UAV can carry out a delegated task/mission without human interaction where all decisions are made onboard based on sensors observations adapting to operational and environmental changes.
- **Semi-autonomous:** A human operator is needed for high-level mission planning and for interaction during the movement when some decisions are needed that the UAV is not capable of making. The vehicle can maintain autonomous operation in between these interactions. For example, an operator can provide a list of waypoints to guide the vehicle where it can manage to move safely towards these positions with obstacle avoidance capability.
- **Teleoperated:** The remote operator relies on feedback from onboard sensors to move the vehicle either by directly sending control commands or intermediate goals with no obstacle avoidance capabilities. This mode can be used in Beyond-Line-of-Sight (BLOS) applications.
- **Remotely controlled:** A remote pilot is needed to manually control the UAV without sensors feedback which can be used in Line-of-Sight (LOS) applications.

2.3. Towards Fully Autonomous Operations

Developing a fully autonomous UAV is a very challenging and complex problem. A modular approach for both hardware and software architectural design is commonly adopted in the literature by most existing autonomous UAVs for a simpler and fault-tolerant solution.

At the hardware level, a UAV in its simplest form consists of a frame, a propulsion system and a Flight Control System (FCS). The UAV's size and propulsion system can be designed to support the needed payload and flight time as per the mission requirements. A propulsion system consists of a power source (ex. batteries, fuel cells, micro-diesels and/or micro gas turbines), motors drivers or electronic speed controllers (ESCs), motors (ex. brushless DC motors), propellers and/or control surfaces (ailerons, flaps, elevators, and rudders).

The flight control system is simply an embedded system consisting of the autopilot, avionics and other hardware directly related to flight control [14]. For example, main sensors critical to flight control include inertial measurement units (IMUs), barometers/altimeters, and GNSS (for outdoor use). Existing commercial products offer complete systems combining these sensors, which are known as Attitude Heading Reference Systems (AHRSs). More advanced solutions include onboard Kalman filtering to fuse data from all sensors to provide absolute positioning solutions; these are referred to as Inertial Navigation Systems (INSs). The next component is the computing unit (ex. a microcontroller), which is usually used to implement the autopilot logic for reliable and fault-tolerant flight control. Ideally, the computing unit must be subject to real-time constraints. That is, its response must be deterministic and within specified time constraints. In general, FCS is responsible for computing low-level control commands, estimating the vehicles states (altitude, attitude, velocity, etc.) based on sensor data, logging critical information for post-flight analysis, and interfacing with higher level components either by wired connection or through other communication links. Having a FCS is enough to allow teleoperation navigation mode where a remote operator can directly send waypoints and/or control commands. It is also possible to achieve semi-autonomous operations in simple environments where reactive control methods with low computational cost are implemented within the autopilot to provide basic collision avoidance capabilities.

For more complex tasks/missions, an onboard computer with higher processing power, namely a mission computer, is required to achieve fully autonomous operations given that a UAV with proper size and power is used. In this structure, the mission computer usually implements high-level mission and motion planning by relying on information interpreted from high-bandwidth sensory data in addition to running required

processes with expensive computational cost. It can also have its own communication link with a Ground Control Station (GCS) to stream high-bandwidth data such as images and depth point clouds.

Different kinds of sensors can be used for advanced perception and planning, depending on the mission requirements, UAV available payload and power, and environmental conditions. Examples of commonly used sensors are cameras (monocular, RGBD, thermal, hyperspectral, etc.), range sensors (LiDAR, RADAR, ultrasonic) and other task-specific sensors (ex. grippers, manipulators, sprayers, etc.). A summary of hardware and software components used with UAVs are shown in Figures 2 and 3, and an example hexacopter is presented in Figure 4 showing the system components for some use case.

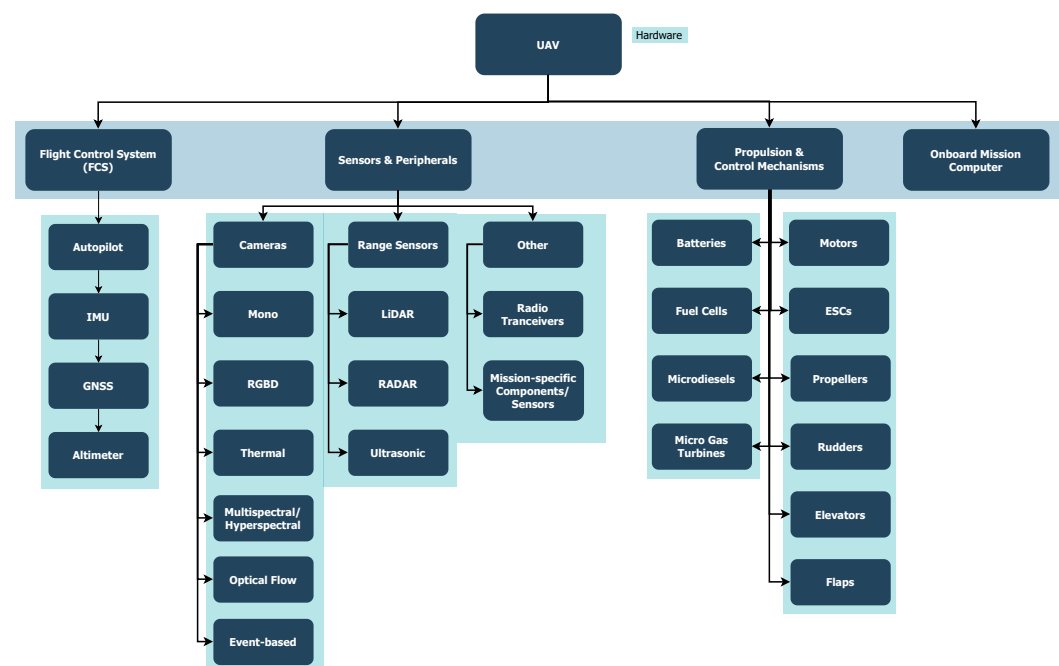


Figure 2. System architecture showing hardware components commonly used with UAVs.

The software architecture of the autonomous stack implemented on the mission computer typically consists of several processes/modules running in parallel and a messaging middleware is used to interchange messages between processes on the mission computer or with other computers on the same network (for example, in multi-UAV systems). Some of these modules are related to the mobility aspects that can ensure safe navigation which can be common among most UAV systems and other autonomous mobile robots. Other modules would implement logic that is application-specific such that the UAV can autonomously perform the delegated task. For example, in fire-fighting applications, a UAV is needed to autonomously locate and extinguish fires, which requires additional modules to be included within the autonomous stack including computer vision pipelines and an extinguisher control mechanism. In many remote sensing applications, the main task could be only collecting data either in the form of images or information from other onboard sensors to be analyzed and processed post-flight.

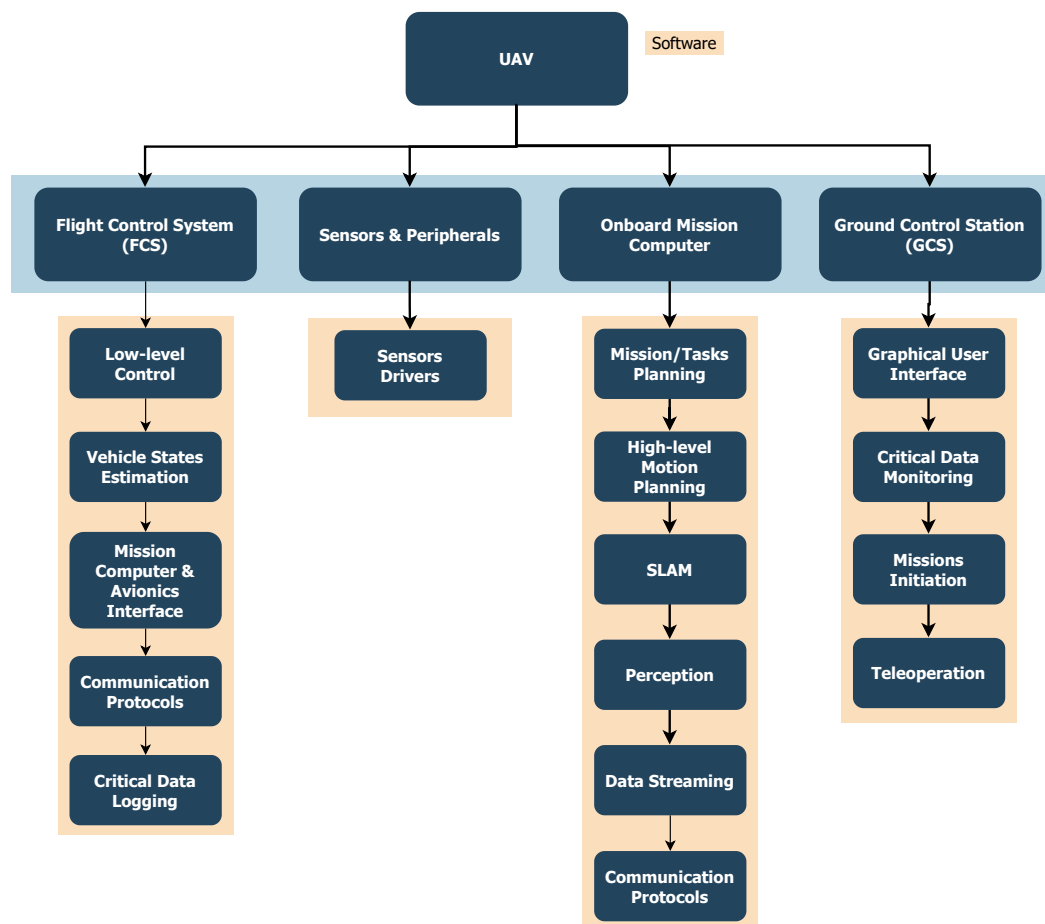


Figure 3. System architecture showing software components commonly used with UAVs.

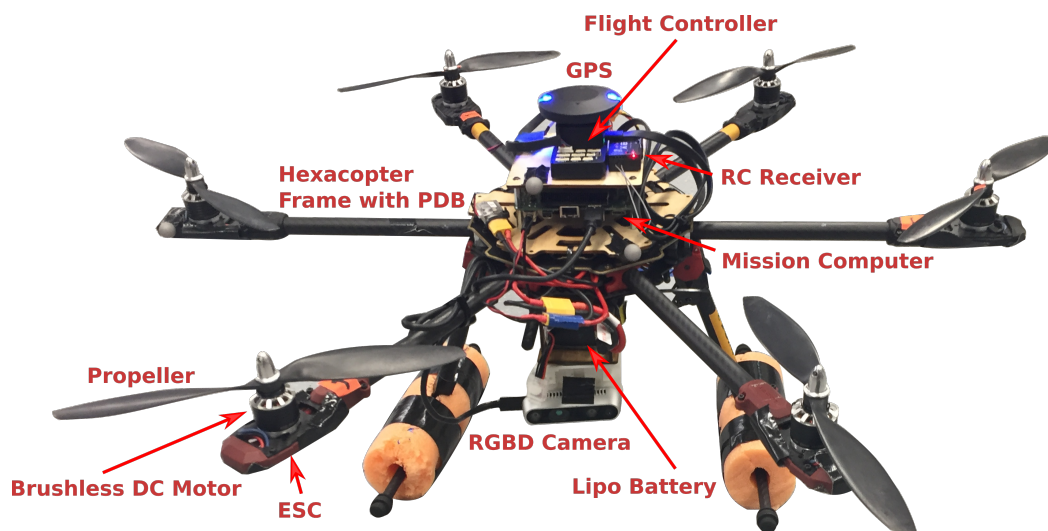


Figure 4. Example UAV setup of a hexacopter UAV type with FCS, mission computer and an RGBD sensor.

Mobility-related modules are the core components needed to ensure collision-free navigation in all applications. By considering only the mobility-related components, a popular modular structure for autonomous navigation is adopted in the literature which consists of the following modules/subsystems (Figure 5):

- Perception;
- Localization and Mapping;

- Motion Planning and Obstacle Avoidance;
- Control.

This modular approach of addressing the navigation problem offers a flexible expandable design with fault-tolerance. However, other possible designs can also be seen for less complex tasks or for vehicles with very limited resources by coupling control and planning, without the need for localization and mapping, in a reactive fashion, as will be shown in the next section.

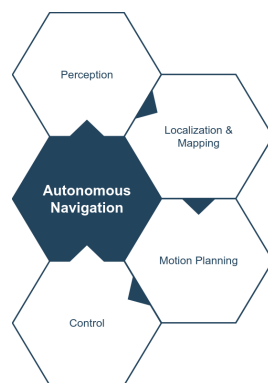


Figure 5. Modular software structure for UAV navigation stack.

3. Navigation Techniques

A crucial part of autonomous navigation is to ensure that the vehicle can move while avoiding collisions with its surroundings. This is a general problem in robotics which can be addressed by motion planning or reactive control. Generally, the motion planning problem can roughly be described as trying to find collision-free trajectories between initial and final *configurations* while satisfying some kinematic and dynamic constraints. A *configuration* in this case refers to the position and orientation of a mobile robot where a *configuration space* is the set of all possible configurations. The dimension of the configuration space equals the number of controllable degrees of freedom. For example, planning motions for quadrotors can be carried out in a space of their 3D position coordinates and heading (yaw) angle while motions for omnidirectional (fully actuated) UAVs can be planned considering all translational and rotational states (6DOF).

In a decoupled approach, the UAV control system can execute motions planned by a high-level system, namely a motion planner, where these plans need to be feasible and safe (i.e., collision-free). In other implementations, motion planning can be coupled with the control system design where reactive control laws are developed to directly generate obstacle avoidance maneuvers based on sensor measurements. Some refer to those in the literature in loose terms as obstacle/collision avoidance methods. The term collision avoidance is mostly used by the UAV research society in referring to avoiding collisions with other cooperative or noncooperative aerial vehicles (i.e., dynamic obstacles) sharing the same flight space, while the term obstacle avoidance may be used more often in indoor, industrial and urban environments where the flight space is filled with other static/dynamic obstacles. That is, high-altitude flights commonly adopt the collision avoidance terminology and low-altitude flights may use the more general obstacle avoidance term. This terminology is also adopted more often in multi-UAV systems to differentiate between methods that only consider collision avoidance among the vehicles within the system to those that also consider obstacle avoidance in obstacle-filled environments.

3.1. Navigation Paradigms

Existing navigation techniques for autonomous mobile robots in general can be classified into *deliberative* (global planning), *sensor-based* (local planning) or *hybrid* (see Figure 6). *Deliberative* approaches require a complete knowledge of the environment represented as a

map. Global path planning methods can then be used to search for safe and optimal paths. Classical path planning algorithms can be categorized into:

- Search-based methods (ex. Dijkstra, A^* , D^* , etc.);
- Potential field methods (ex. navigation function, wavefront planner, etc.);
- Geometric methods (ex. cell decomposition, generalized Voronoi diagrams, visibility graphs, etc.);
- Sampling-based methods (ex. PRM, RRT, RRT*, FMT, BIT, etc.);
- Optimization-based methods (PSO, genetic algorithms, etc.).

Many of these methods can find optimal paths if one exists at the expense of requiring full knowledge about the environment which is not suitable in unknown and dynamic environments. For more detailed information about such planning methods, the reader is referred to [50].

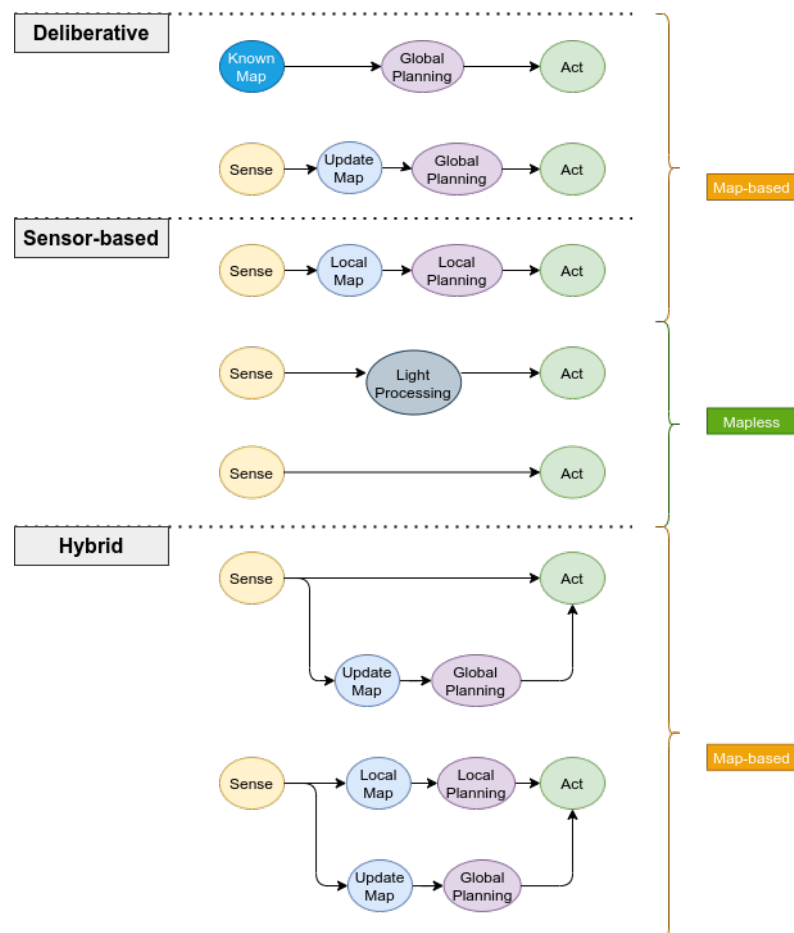


Figure 6. Different paradigms adopted for autonomous navigation from a high-level perspective.

On the other hand, *sensor-based* methods rely directly on current sensor measurements or a short history of the sensors' observations (i.e., a local map) to plan safe paths in real time. The planning horizon can typically be very short for some period ahead of time or it could be carried out at each control update cycle in a receding horizon fashion. A special class of such methods is made up of reactive approaches where sensor measurements are coupled to control actions either directly [51] or after light processing [52]. Sensor-based methods offer solutions with great computational performance which makes them favorable for navigation problems in unknown and dynamic environments. These methods do not generate global optimal solutions as they do not use complete knowledge about the environment during motion; however, it is possible to find locally optimal solutions. In practice, it is common to sacrifice optimality for computation speed, especially when

considering micro-UAVs with fast dynamics and limited computing power. Sensor-based methods are also prone to getting stuck sometimes due to the local minimum.

Hybrid approaches combine both deliberative and sensor-based methods to generate a more advanced navigation behavior benefiting from the advantages of both classes. It relies on low-latency local planning or reactive control to handle unknown and dynamic obstacles while using a high-level global planning method to guide the vehicle utilizing accumulated knowledge about the environment.

3.2. Map-Based vs. Mapless Methods

Navigation methods can alternatively be classified into *map-based* or *mapless* approaches [53,54]. This classification highlights the computational complexity including memory requirements and whether they rely on accurate localization and mapping or not.

Map-based strategies require a local (or global) map representation of the environment which can be provided before navigation starts (deliberative approaches) or it can be built during navigation based on sensor measurements (some sensor-based approaches). Safe paths can then be found using local/global planning algorithms based on either *metric* or *topological* maps. Therefore, such methods are demanding in terms of computational resources, planning time and memory requirements, which is highly dependent on the environment size and its complexity. Nevertheless, local map-based methods are very commonly used with UAVs to generate local optimal solutions for technological advances where it is possible to have mini lightweight computers with high processing power onboard.

On the contrary, *mapless* strategies (reactive methods) rely directly on sensor measurements to make motion decisions without the need for maintaining global maps and accurate localization (except when using GNSS). Hence, control actions can be directly coupled with either visual clues from image segmentation, optical flow or feature tracking in subsequent frames in vision-based methods [54] or interpreted information from range sensors and 3D point clouds such as relative distance to obstacles, gaps or bounding objects. These methods offer the best computational complexity for obstacle avoidance as control is coupled with planning through light processing of sensor data which can provide very quick reflex-like reactions to obstacles. Some of the challenges when developing purely reactive navigation methods is the possibility of getting stuck in local minimums, and a limited field of view (FOV) may affect the overall performance. Additionally, fast reactions to obstacles achieved by reactive methods come at the cost of generating nonoptimal solutions in some cases due to the fact that they do not utilize information about previously sensed obstacles.

3.3. Overall Navigation Control Structure

From a control perspective, different structures were adopted in the literature to deal with the high complexity of the navigation problem. As mentioned before, the most common structure is based on decoupling planning and control due to its simplicity in design. One can categorize the existing methods into seven different control structures, as shown in Figure 7. Structures I–III show the general decoupled approach where motion planning and control are decoupled, while structure IV is used by reactive approaches which directly couple planning and control. Structures V–VII correspond to hybrid approaches which can be a combination of structures I–IV.

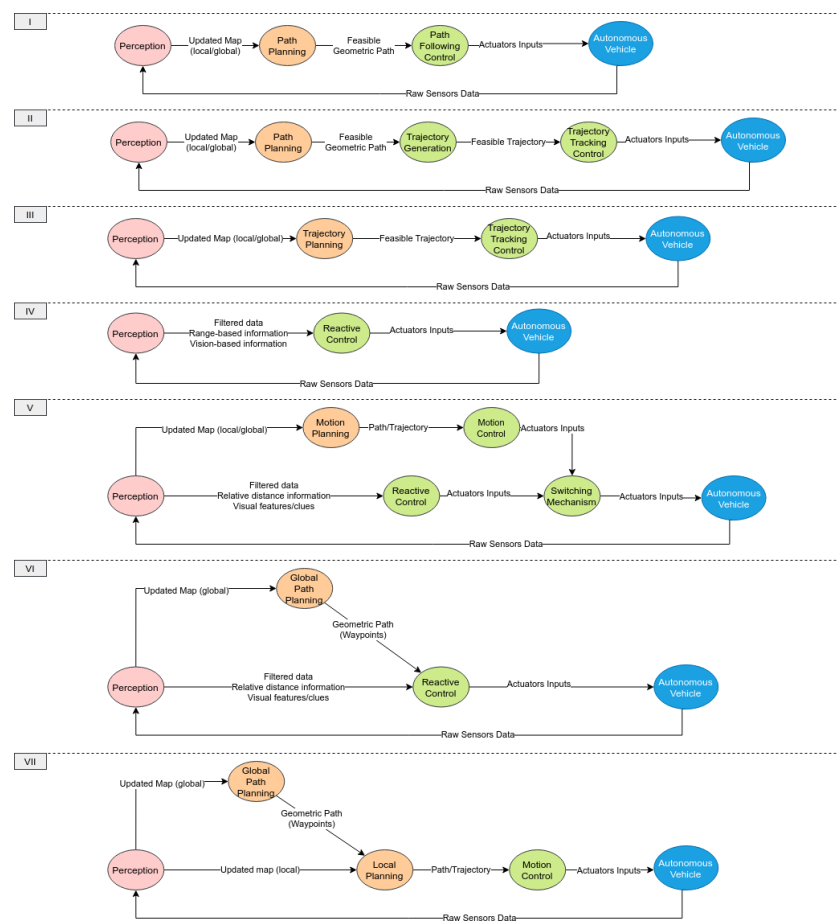


Figure 7. Different autonomous navigation control structures.

In decoupled approaches, some motion planning methods further simplify the problem by subdividing it into two stages. The first stage simply tries to find a collision-free geometric path satisfying kinematic constraints. Constraints can be considered directly in the planning algorithm, or the entire process can be further decomposed into finding a safe path first ignoring such constraints then applying path smoothing techniques to satisfy the kinematic constraints. Then, it is followed by a trajectory generation stage to obtain feasible trajectories satisfying dynamic constraints. Other approaches tackled this problem by directly planning trajectories using optimization-based methods which is a harder problem to solve.

To differentiate between different motion planning paradigms, we highlight the differences between *path planning* and *trajectory planning/generation*. *Path planning* is the process of finding a geometric collision-free path between starting and end positions without a timing law. In *trajectory planning*, a timing law is associated with the planned collision-free geometric path represented as a *trajectory* which includes information about higher derivatives (i.e., velocity, acceleration, etc.). Trajectories are mostly planned to satisfy dynamic constraints which can then be passed to a control system adopting a trajectory tracking control design. One of the common approaches for trajectory planning is by using a path planning algorithm to find an initial geometric path followed by formulating trajectory generation as an optimization problem to plan local optimal trajectories around the initial path subject to several constraints. Alternatively, some approaches adopt ideas from missile guidance, implementing path following control laws to track the geometric path given that it satisfies nonholonomic constraints whenever they exist without generating a trajectory.

In the following subsections we will survey recent works adopting local motion planning or reactive paradigms in accordance with the considered control structures.

3.4. Local Path Planning

A number of existing methods treat the problem through applying path planning algorithms locally to find feasible geometric paths assuming a general 2D/3D kinematic model. Examples of these methods include sampling-based [55–58], graph-based [59,60] and optimization-based methods [61–63]. These methods are developed at a high level considering only kinematic constraints assuming a low-level path following controller exists to execute the planned paths while satisfying the dynamic constraints similar to control structure I. They can also be combined with a trajectory generation method similar to structure II.

Adopting sampling-based methods helps address the high dimensionality problem of the 3D search space to generate collision-free paths in real time, which was considered in many works. In [56], a planning algorithm was proposed for rotary-wing UAVs. It decouples the motion planning problem into two stages, namely path planning and path smoothing, which is a common approach to simplify the problem, especially when nonholonomic constraints need to be satisfied (ex. for fixed-wing UAVs); for example, see [64,65]. A sampling-based planning algorithm, namely RRR, was adopted to search for collision-free paths followed by a path smoothing algorithm such that the smoothed path can satisfy curvature continuity and nonholonomic constraints. An analytical solution for the adopted path smoothing algorithm was also presented in [66] considering smoothing of 3D paths. An explicit path-following model predictive control (MPC) was used in [56] to ensure that the vehicle can track the planned paths, and it was formulated based on a linear model of the motion with no constraints. Another real-time path planning algorithm was suggested in [55] based on chance-constrained rapidly exploring random trees (CC-RRT) for safe navigation in 2D constrained and dynamic environments. The motion planning relies on a proposed clustering-based trajectory prediction to model and predict future behavior of dynamic obstacles. This motion prediction algorithm combines Gaussian processes (GPs) with the sampling-based algorithm RRT-Reach to cope with GP shortcomings such as the high computational cost. Another RRT variant, namely Closed-Loop RRT, was used in [57] to handle navigation in 3D dynamic environments. In [58], a sampling-based approach was adopted in an informative path planning framework where the goal is to generate safe paths that can maximize the information gathered during movement, which is important in exploring unknown environments.

Some other works formulated the 3D path planning problem as an optimal control problem, such as [61,62]. The authors of [61] formulated the optimal control problem in 2D to satisfy time and risk constraints as the 3D optimal control problem would be harder to solve. Then, a 3D path was approximated in a final stage based on a terrain height map. In contrast, the method in [62] presented a path planner based on a 3D optimal control problem formulation where a model based on artificial potential field (APF) was used. Other optimization-based methods considered parallel genetic algorithm and particle swarm optimization as in [63].

3.5. Local Trajectory Planning

A more popular approach in addressing the local planning problem for UAVs is through planning feasible trajectories to further satisfy dynamical constraints and optimality of path smoothness with respect to higher derivatives enabling high-speed and aggressive movements. Generating smooth trajectories is important for high-speed applications to avoid sudden changes in actuators' accelerations and mechanical vibration problems [67]. Therefore, it can be seen from the literature that control structures II–III are commonly used for aggressive maneuvers, whether by combining path planning and trajectory generation, as in [59,68–76], or by direct trajectory planning, as in [77–86].

Generally, many of these approaches represent the trajectories as piecewise polynomials where the polynomial coefficients are used as decision variables in the optimization problem. Some works use Bernstein and B-splines polynomial bases and utilize their properties when formulating the problem [76,78,87]. For example, Bezier curves are known

for their convex hull property, where a whole trajectory segment can be contained within a convex region by adding constraints on its Bezier control points.

A trajectory generation method for quadrotors was suggested in [68] to find minimum-snap trajectories between specified keyframes provided by a high-level planner with corridor-like constraints, representing the convex decomposition of free space. This pioneering approach was adopted in several studies such as [59,70,72,73,88]. The work [70] formulated the trajectory generation as a mixed-integer optimization problem to generate minimum-jerk polynomial trajectories constrained to convex collision-free regions with other constraints on velocity and acceleration. The authors have also proposed a way to generate the safe convex regions using Iterative Regional Inflation by Semi-definite programming (IRIS), which was initially proposed in [89].

Similarly, a real-time trajectory generation method was proposed in [59] for quadrotors, presenting another way of determining such safe convex regions. It relies on online built voxel maps and short-range planning algorithm where it uses an A^* search method to find a safe path in a discretized graph representation of the voxel map. The path is then inflated to generate a set of connected polyhedrons specifying the collision-free regions around the path, resulting in corridor-like constraints. This approach was further developed in [72] to provide a more robust and efficient solution which was implemented in [88], showing a complete system for autonomous flights of multi-rotors in GPS-denied indoors environments. A minimum-jerk trajectory is then computed similar to the approach in [73] where a convex optimization problem is formulated by confining the trajectory spline segments to be within specified flight corridors with constraints to ensure the continuity of the trajectory splines. This approach avoids the more complex nonconvex problem formulation that results when considering the trajectory planning problem with constraints corresponding to collisions with obstacles.

The works [72,73] adopt a receding horizon planning paradigm to plan trajectories over finite time intervals with safe stopping policies in case of planning failure. The works [59,73] adopt a short-range planning paradigm where a set of candidate goals within the current sensing FOV are used for trajectory planning until the global goal is reached. In contrast to expressing collision-free constraints as convex decomposition of free space, the authors of [79] suggested a different approach to efficiently handle dynamic and cluttered environments as the authors claimed that the conventional convex decomposition of free space can be conservative and may become harder in dynamic environments. This approach is based on using planes to represent the separation between the polyhedral representations of each trajectory segment. Additionally, the authors use decision variables to represent these planes within the optimization problem.

Another optimization-based method was suggested in [69] as an extension to [68] by formulating the minimum-snap trajectory generation problem as an unconstrained quadratic program (QP). This trajectory generation can be combined with a 3D kinematic planner to generate safe geometric paths where the authors have considered the RRT* planner in their implementation. Additional iterative steps are needed if the generated trajectories were found in collision where the optimization problem is repeatedly resolved using safe intermediate waypoints until a collision-free trajectory is obtained.

In contrast to optimization-based trajectory generation where dynamic constraints are considered in the optimization problem, motion primitives were considered as a simpler computationally efficient way to generate collision-free trajectories in 3D in some works such as [77,90–94]. Motion primitives offer a light-weight algebraic solution to the problem which can then be checked for dynamic constraints violation. The low-computational cost of such methods allows for high-speed and aggressive movements since it is possible to quickly search over a large number of motion primitives to achieve a certain goal [90]. Motion and sensing uncertainty were also considered in some methods at planning time such as [94].

Generally, considering dynamic constraints and constraints due to collisions with obstacles in the planning problem makes it harder to solve in real time, causing potential

convergence problems. This is known as kinodynamic planning, which is a motion planning problem in a higher dimensional space with differential and obstacle constraints [95]. Some approaches, however, have tackled this more complex problem rather than decoupling the path planning and trajectory generation such as [74,75,96]. The work [74] addressed the trajectory planning problem as a 3D Optimal Control Problem (OCP) with soft obstacle avoidance constraints on a nonconvex quadratic optimization problem. To reduce the computational burden of solving the formulated OCP, constraints based on a reduced number of obstacles, the most threatening ones, were considered. In [75], trajectory planning and control of quadrotors in constrained environments was achieved through a formulation as a minimum-time optimal control problem with several constraints on states and inputs, and it was based on the full 6DOF dynamical model. The general problem was reformulated using a change of coordinates and state-input constraints relaxation to reduce the high computational complexity of the original constrained problem.

The motion planning problem for multi-rotors among dynamic obstacles was tackled in [96] at the control level using a nonlinear model predictive controller (NMPC) based on a cost function in terms of the tracking error, input cost and input smoothness cost. Addressing path planning using a pure NMPC structure is challenging as it is computationally expensive to solve nonconvex optimization problems in real time. Therefore, [96] considered a new solver for such nonlinear nonconvex problems known as Proximal Averaged Newton for Optimal Control (PANOC) [97,98] to make the solution more appealing. There exists an open-source implementation of this solver which is OpEn (Optimization Engine) [99]. A similar approach was also considered in [100].

Formulating the 3D trajectory planning as a Quadratic Program (QP) was also considered in [71,78,82]. In [71], an optimization-based method was proposed to generate locally optimal safe trajectories for multi-rotor UAVs using high-order polynomial splines. The optimization problem was formulated to minimize costs related to higher order derivatives of the trajectory (ex. snap) and collisions with the environment. The objective function computes collision costs using a Euclidean Signed Distance Field (ESDF) function with a voxel-based 3D local map of the environment. The optimization problem was formulated as an unconstrained quadratic program (QP) so that it can be solved in real-time. The work [78] adopted a mixed-integer quadratic program formulation allowing the solver to choose the trajectory interval allocation, and the time allocation is found by a line search algorithm initialized with a heuristic computed from the previous replanning iteration. Another kinodynamic planner for quadrotors was introduced in [82] using a sampling-based method in combination with an additional optimization-based stage using a sequence of QPs to refine the smoothness and continuity of the obtained trajectory.

Recently, there has also been some growing interest in the field of perception-aware trajectory planning considering perception constraints in the planning problem. The developed methods in this area take into account perception quality to minimize state estimation uncertainty [101], which can be achieved by keeping specific objects/features in the vehicle's sensing FOV [80]. Examples of such methods can be seen in [80,101–105].

3.6. Reactive Methods

Most of the existing reactive methods are developed at a higher level considering different abstractions of UAV 2D/3D kinematic models with velocities/accelerations as control inputs. Collision avoidance can be ensured rigorously for some of these methods under certain technical assumptions [51] in contrast to other motion planning methods. For example, the design may rely on assumptions made about obstacles (shape, size, velocity profile, etc.), environment (static or dynamic) and sensing capabilities (vision-based, distance-based, FOV, range, etc.). Many of the existing reactive methods are planar, which can generally be applied to various types of mobile robots including UAVs moving at a fixed altitude; examples of such methods include [106–114]. Adopting these methods for vehicles that can navigate in 3D, such as UAVs and AUVs, becomes less efficient. Therefore, there has been a growing interest in developing 3D reactive navigation methods which will

be the main focus in this section in addition to some of the 2D vision-based approaches sufficiently suitable for UAVs in some applications.

A number of geometric-based reactive collision avoidance methods focused on non-cooperative scenarios (i.e., dynamic environments) for fixed-wing UAVs or vehicles with nonholonomic constraints adopting the idea of collision cones such as [115–120]. Many of these approaches use linear or nonlinear guidance laws to align the velocity vector (i.e., controlling heading and flight path angles) in a certain direction while keeping a constant relative distance to the obstacle to avoid collisions. The work [115] proposed two guidance laws for collision avoidance in static and dynamic environments based on collision cones where the vehicle is guided to track the surface of a safety sphere around the obstacle. Similarly, the works [116,117] adopted collision cones to safely guide fixed-wing UAVs in 3D dynamic environments. In [118], a 3D reactive navigation law was proposed based on relative kinematics between the vehicle and obstacles decoupled into horizontal and vertical planes. Obstacles were modeled as spheres, and collision cones were used for obstacle avoidance. This method was further developed in [119] where a reactive optimal approach was suggested for motion planning in dynamic environments.

A different implementation of collision cones was carried out in [120] for AUVs; however, the same idea can be applied to UAVs as well. No assumptions were made about the obstacle shape; however, obstacles were modeled as spheres for mathematical development, and it was only assumed that the collision cone to the obstacle can be interpreted from sensor measurements. This method relied on maintaining a constant avoidance angle from a nearby obstacle while ensuring a minimum relative distance is achieved. The same problem was addressed differently in [121] where a new nature-inspired 3D obstacle avoidance method for AUVs was developed based on concepts from fluid dynamics.

Another 3D reactive approach was developed in [122–124] adopting the idea of avoidance planes with more flexibility in choosing the orientation of these planes while circumnavigating around obstacles.

A different class of 3D reactive methods modified the Velocity Obstacle (VO) approach to allow navigation in dynamic environments such as [125,126]. In [125], the proposed method relied on decoupling the 3D motion to achieve constant relative bearing and elevation in both the horizontal and vertical planes simultaneously. It was assumed that the desired relative bearing and elevation with respect to the noncooperative vehicle can be estimated using onboard cameras. Additionally, the authors of [126] proposed an improvement to the Velocity Obstacle (VO) method to handle 3D static and dynamic environments.

Artificial potential field was also considered in some approaches to handle navigation in dynamic environments as in [127–129]. The approaches [127,128] developed modified APF methods for 3D nonholonomic vehicles, while the work [129] designed an APF reactive controller for quadrotors. The approach in [129] combines obstacle avoidance control law based on artificial potential field with a trajectory tracking control law using on a null-space-based scheme at the kinematic level where the obstacle avoidance input has the higher priority. A dynamic controller was then proposed to generate low-level input to ensure that velocities generated by the kinematic controller can be tracked.

The authors of [130] suggested a different 3D navigation approach for rotorcraft UAVs where an escape waypoint is determined whenever an obstacle is detected. Obstacle detection was carried out by extending a cylindrical safety volume from the UAV position along the movement direction in a 3D local map representation of the environment. The escape waypoint is determined by performing a search through a set of concentric ellipsoids around the detected obstacles by iteratively incrementing the ellipses radii until a safe escape point is found. Due to the low complexity of the algorithm, it belongs to the reactive class.

In [131], a computationally light approach was suggested through real-time deformations of a predefined 3D path based on the intersection between two 3D surfaces determined according to the free space and obstacles. Either one or both surfaces are modified in the presence of obstacles such that the intersection between the two surfaces provides a path

around the obstacle. To that end, proper functions need to be carefully chosen to represent the obstacle where the authors considered a Gaussian function whose parameters require proper tuning. A path following controller was also proposed based on multi-rotor full dynamical model where a cascaded approach for control was adopted for position and attitude. This was further implemented in [132] where a depth camera was used to detect obstacles. Another 3D reactive method adopting the idea of real-time deformable paths around dynamic obstacles was also proposed in [133].

A number of reactive methods considers vision-based structure such as [83,84,86,134,135]. In [134], a vision-based reactive approach was proposed for quadrotor MAVs based on embedded stereo vision. Obstacles are detected from stereo images-based U-V disparity maps. A short-term local map is built for planning purposes representing approximations of detected obstacles as ellipsoids. Hence, no accurate odometry is needed since no global map is built. The obstacle avoidance algorithm is mainly 2D to find the shortest path along obstacles' edges. On the other hand, the works [83,84,86] proposed 3D mapless vision-based trajectory planning methods using depth images which can be considered reactive as the planning horizon becomes very short. A different vision-based 3D reactive method was proposed in [135] based on NMPC for quadrotors navigating in dynamic environments.

Some other methods relied on 3D distance measurements (i.e., 3D pointclouds) obtained from LiDAR sensors or depth cameras such as [12,100]. The method proposed in [100] combined 3D collision avoidance with control in a nonlinear model predictive control scheme considering both dynamic and geometric constraints at the same time. It adopted a mapless approach by relying on a subspace clustering method applied to 3D point clouds obtained directly from a 3D LiDAR sensor. On the contrary, a 3D reactive approach was suggested in [12] to allow navigation in tunnel-like environments. Guidance control laws were developed to guide the UAV by directly extracting clues from 3D pointclouds to determine a progressive direction to advance through the tunnel.

Concepts from machine learning were also considered recently in some reactive methods to address obstacle avoidance problems for UAVs. However, these methods are more computationally expensive than other reactive methods, and there are still concerns related to how guaranteed a collision avoidance is as the performance relies on how good the training/learning stage is. Additionally, many of the existing approaches consider only generating motion decisions/policies in 2D without utilizing the full maneuverability of UAVs. Most of these methods are based on deep reinforcement learning [136–143] and deep neural networks [144–151].

A summary of the surveyed local motion planning methods is presented in Table 1.

Table 1. A summary of surveyed local motion planning methods for UAVs.

Refs.	Control Structure	Local Motion Planning	Model	Dynamic Environment
[55]	I/II	sampling-based path planning	2D Kinematics (nonholonomic)	✓
[56]	I/II	sampling-based path planning	3D Single-rotor Dynamics	
[57]	I/II	sampling-based path planning	3D Kinematics (nonholonomic)	✓
[58]	I/II	sampling-based path planning	3D Kinematics (holonomic)	
[60]	I/II	graph-based path planning	3D Kinematics (holonomic)	✓
[61,63]	I/II	optimization-based path planning	3D Kinematics	
[62]	I/II	optimization-based path planning	3D Quadrotor Dynamics	
[59,68,70,72,73,88]	II/III	optimization-based trajectory generation using QP with corridor-like constraints	3D Quadrotor Dynamics	

Table 1. Cont.

Refs.	Control Structure	Local Motion Planning	Model	Dynamic Environment
[78,82,93]	III	optimization-based trajectory planning using QP	3D Dynamics (acceleration/jerk input)	
[69,85]	III	optimization-based trajectory planning using unconstrained QP	3D Quadrotor Dynamics	
[71,74,75]	III	optimization-based trajectory planning with obstacles constraints	3D Quadrotor Dynamics	
[77,81,90,92,101]	III	motion primitives	3D Quadrotor Dynamics	
[94]	III	motion primitives	3D Kinematics (holonomic)	
[91]	III	motion primitives	3D Kinematics (nonholonomic)	
[79,80]	III	perception-aware trajectory planning	3D Dynamics (jerk input)	✓
[101–105]	III	perception-aware trajectory planning	3D Quadrotor Dynamics	
[96,100]	III/IV	nonconvex optimization with obstacles constraints using NMPC	3D Quadrotor Dynamics	✓
[83,84,86]	III/IV	mapless vision-based trajectory planning using depth images	3D Dynamics (jerk input)	
[115–120]	IV	Geometric-based (collision cones) reactive control	3D Kinematics	✓
[125,126]	IV	reactive control based on Velocity Obstacle (VO)	3D Kinematics	✓
[127–129]	IV	reactive control based on artificial potential field	3D Kinematics (nonholonomic)/Quadrotor Dynamics	✓
[121]	IV	nature-inspired reactive control	3D Kinematics (nonholonomic)	
[134]	IV	vision-based reactive control	2D Kinematics	
[131–133]	IV	real-time path deformation (reactive)	3D Quadrotor Dynamics	✓
[135]	IV	vision-based reactive control based on NMPC	3D Quadrotor Dynamics	✓
[136–141]	IV	deep reinforcement learning	2D Kinematics	
[142]	IV	deep reinforcement learning	2D Kinematics	✓
[143]	IV	deep reinforcement learning	3D Kinematics	
[144–148]	IV	deep neural networks	2D Kinematics	
[149,150]	IV	deep neural networks	3D Kinematics	
[151]	IV	deep neural networks	3D Kinematics	✓

4. UAV Modeling and Control

4.1. Modeling

For control design and simulation purposes, it is necessary to have a valid mathematical model that can express the UAV motion. Generally, such a model consists of two main parts which are *kinematics* and *dynamics*. *Kinematic* equations are mainly derived to represent the geometrical aspects of the motion in 3D spaces through defining translation

and rotation relationships between different coordinate frames. *Dynamics* can be obtained through the application of Newton laws for a moving rigid body to derive linear and angular momentum equations. Application of Newton laws requires an inertial reference frame \mathcal{I} to be defined. On the other hand, analyzing forces and torques acting on the vehicle needs to be carried out with respect to a coordinate frame attached to the moving vehicle (i.e., a body-fixed frame \mathcal{B}). Clearly, different UAV types would have some differences in their dynamic equations depending on the actuators' configurations and other external forces and torques acting on the vehicle. For simplicity, the origin of the body-fixed frame is commonly chosen to coincide with the vehicle's center of mass. Note that there are other coordinate frames that can be used for different purposes for navigation and control such as Earth-Centered, Geodetic and wind coordinate frames. For more details about these coordinate frames, see [14].

A rotation matrix between the inertial and body-fixed coordinate frames can be used to define the attitude/orientation of the UAV. It is also common to use other representations such as Euler angles (i.e., roll ϕ , pitch θ and yaw ψ) and quaternions $q \in \mathbb{R}^4$. Quaternions are more computationally efficient and do not have the gimbal lock problem while Euler angles are easier to understand physically and can be decoupled into separate degrees of freedom under some assumptions for simplicity.

Let the Euler angle vector be $\Phi = [\phi, \theta, \psi]^T$, and consider a quaternion vector $q = [q_1, q_2, q_3, q_4]^T$. Notice that with Euler angles, usually three rotations are applied in a specific order which can result in different forms for the rotation matrix. The following is an example considering the rotation order ZYX ,

$${}^{\mathcal{I}}_{\mathcal{B}}\mathbf{R}(\Phi) = \begin{bmatrix} c_{\theta}c_{\psi} & s_{\theta}s_{\phi}c_{\psi} - c_{\phi}s_{\psi} & c_{\phi}s_{\theta}c_{\psi} + s_{\phi}s_{\psi} \\ c_{\theta}s_{\psi} & s_{\theta}s_{\phi}s_{\psi} + c_{\phi}c_{\psi} & c_{\phi}s_{\theta}s_{\psi} - s_{\phi}c_{\psi} \\ -s_{\theta} & s_{\phi}c_{\theta} & c_{\phi}c_{\theta} \end{bmatrix} \quad (1)$$

where $c_{\alpha} := \cos \alpha$, and $s_{\alpha} := \sin \alpha$. Note that ${}^{\mathcal{I}}_{\mathcal{B}}\mathbf{R}(\Phi)$ represents the rotation from the body-fixed frame to the inertial frame. Furthermore, ${}^{\mathcal{B}}_{\mathcal{I}}\mathbf{R}(\Phi) = {}^{\mathcal{I}}_{\mathcal{B}}\mathbf{R}^T(\Phi)$.

For a velocity vector expressed in the body-fixed frame, it can be transformed to the inertial frame as follows:

$${}^{\mathcal{I}}v = {}^{\mathcal{I}}_{\mathcal{B}}\mathbf{R}(\Phi) {}^{\mathcal{B}}v \quad (2)$$

such that ${}^{\mathcal{I}}v = [x, y, z]^T$ and ${}^{\mathcal{B}}v = [u, v, w]^T$. Additionally, the angular velocity can be transformed from \mathcal{B} to \mathcal{I} as:

$$\dot{\Phi} = T(\Phi)\Omega \quad (3)$$

where

$$T(\Phi) = \begin{bmatrix} 1 & s_{\phi}t_{\theta} & c_{\phi}t_{\theta} \\ 0 & c_{\phi} & -s_{\phi} \\ 0 & s_{\phi}/c_{\theta} & c_{\phi}/c_{\theta} \end{bmatrix} \quad (4)$$

with $t_{\theta} := \tan \theta$. The gimbal lock problem can be seen clearly from $T(\Phi)$ where a singularity occurs when $\theta = \pm 90^\circ$. Such a problem does not exist when using quaternions.

Hence, the general model for a UAV is given by:

$$\dot{p} = {}^{\mathcal{I}}_{\mathcal{B}}\mathbf{R}(\Phi) {}^{\mathcal{B}}v \quad (5)$$

$${}^{\mathcal{B}}\dot{v} = \frac{F}{m} - \Omega \times {}^{\mathcal{B}}v \quad (6)$$

$${}^{\mathcal{I}}_{\mathcal{B}}\dot{R} = {}^{\mathcal{I}}_{\mathcal{B}}R\Omega \quad (7)$$

$$\dot{\Omega} = I^{-1}(M - \omega \times I\Omega) \quad (8)$$

where $p, {}^{\mathcal{I}}v \in \mathbb{R}^3$ are the position and linear velocity expressed in the inertial frame, $\Omega \in \mathbb{R}^3$ is the angular velocity defined in the body-fixed frame, $m \in \mathbb{R}^+$ is the UAV's mass, and $I \in \mathbb{R}^{3 \times 3}$ is the inertia matrix. Furthermore, $F \in \mathbb{R}^3$ and $M \in \mathbb{R}^3$ correspond to external forces and torques acting on the vehicle.

Modeling the forces and torques differ based on the UAV type, design and actuators configuration which affects the control system design. Example of these differences can be seen in the complete models for fixed-pitch multi-rotors [68,152,153], variable-pitch multi-rotors [23,40], helicopters [17], fixed-wing UAVs [154], flapping-wing UAVs [33], etc. Some researchers have further extended the UAV modeling considered in the control design to include some added systems such as cable-suspended payload [155,156].

4.2. Low-Level Control

As mentioned earlier, a common approach to handle the navigation problem is by decoupling planning from control. Thus, a low-level control can be designed independently to track the generated reference paths, trajectories, heading/flight path angles or velocity/acceleration commands. Typically, control laws are developed to minimize tracking errors by determining required input forces and body torques which can then be mapped into motor and actuator commands depending on the UAV type. State estimation is a very critical component for feedback control. Extended Kalman Filter (EKF) is a popular choice in many implementations to provide estimates for the UAV attitude, linear and angular velocities by fusing data from different sensors. Position can also be estimated by fusing information from a positioning source such as GNSS, visual odometry, external positioning system, etc.

A cascaded approach is very common in different control structures where the attitude dynamics (i.e., (7) and (8)) are decoupled to avoid considering the full nonlinear system dynamics in the control design [157]. A high-bandwidth inner loop attitude controller is used to ensure that the vehicle can accurately track reference attitude or angular velocity commands. This reduces the control problem to design an outer control loop for the translational dynamics (5) and (6) that can achieve position/velocity tracking by deciding proper laws in terms of thrust, attitude and/or angular velocities. Several control techniques were adopted in the literature, such as PID [17,158], sliding mode control [159], Lyapunov-based nonlinear control [160] and model predictive control [157,161–164].

Multi-rotors are the most popular UAV type for many civilian applications due to their simplicity in mechanical design and control. Therefore, there have been many recent developments in nonlinear control of multi-rotors enabling high-speed navigation [59,77], aggressive flights [165–167] and aerial manipulation [168–170].

Quadrotor dynamics are differentially flat, which was shown in [68] (even under drag effects [153]). Differential-flatness denotes that all system variables (i.e., states and inputs) can be written in terms of a set of flat outputs (for example, $[x, y, z, \psi]$). That is, trajectories can be planned in the space of flat outputs, and it ensures that any smooth trajectory with proper bounded derivatives can be tracked. Hence, several control methods adopted a geometric-based control design utilizing the differential-flatness property such as [68,171]. Model predictive control was also considered in [164] where additional considerations, such as blade flapping and induced drag effects modeled as external disturbances, were included in the model and control design. Including such effects in the control design was considered by several other works such as [153,157,172]. Some other control designs for fixed-pitch multi-rotor UAVs were proposed; for example, see [19,159,162,173–175] and references therein.

Variable-pitch/omni-directional multi-rotors are fully actuated vehicles where translational and rotational degrees of freedom can be decoupled; examples of control methods developed for these vehicles can be found in [23,24,40]. Thus, these vehicles can even perform more complex tasks compared to fixed-pitch multi-rotors where controlling roll and pitch is essential to achieve required translations due to being underactuated systems. Control of single-rotor UAVs (helicopters) has also been tackled in several works using a similar cascaded structure. For example, a PID-based trajectory tracking controller was designed in [17], while a robust and perfect tracking (RPT) technique was suggested in [18].

Control of fixed-wing UAVs followed a similar control structure using decoupled control loops for translational and attitude dynamics. Control designs for fixed-wing

UAVs take into consideration the models nonholonomic kinematic constraints, and many of the existing methods adopt path-following techniques based on guidance laws such as [158,160,176]. In [158], the control method adopted pure pursuit guidance and a decoupled proportional control for velocity and attitude. A similar control method was suggested in [160] based on LOS guidance algorithms and nonlinear control considering wind effects. Model predictive control was also considered in the path-following control design proposed in [161]. Alternatively, [154] presented control designs for fixed-wing UAVs based on linear pole placement and nonlinear structured multi-modal H_∞ synthesis to track a reference air speed and flight path angle. Control of other UAV types has also attracted some interest in the community developing new control methods for hybrid UAVs [48,163], flapping-wing UAVs [33,36], etc.

5. Simultaneous Localization and Mapping (SLAM)

Localization is the process of determining the vehicle's position with respect to a reference frame. This can be achieved given a certain map based on the newly obtained sensors information. On the contrary, mapping is the process of building a map representation of the environment given localization information. Thus, navigation in unknown environments requires both these processes to be carried out online simultaneously, which is known as simultaneous localization and mapping (SLAM). Development of SLAM methods is a very active field of research in robotics as the performance of map-based navigation methods relies on SLAM accuracy. This overview is not intended to provide a detailed survey of SLAM methods; however, the reader is referred to the following surveys for more details on recent developments in this area [177–179]. However, some of the recent state-of-the-art developments are briefly summarized in this section for the sake of completion.

Existing SLAM methods can be classified as either LiDAR-based or vision-based. LiDAR-based methods adopt scan matching algorithms, and they offer better accuracy (ex. see [180–184]). However, vision-based SLAM methods have become more popular for UAVs due to the lower cost and light weight of cameras compared to LiDARs. According to [178], these can be classified into feature-based [185–187], direct [188,189] or RGB-D camera-based methods [190,191]. Feature-based methods rely on detecting and extracting features from an input image to be used for localization which can be challenging in textureless environments. On the contrary, direct methods use the whole image directly, offering more robustness at the expense of increased computational cost. RGB-D camera-based methods combine both image and depth information in their formulation.

6. Summary of Recent Developments

Table 2 summarizes some of the recent contributions made towards developing fully autonomous UAVs, based on the surveyed works, in terms of control, perception, SLAM, motion planning and exploration capabilities.

Table 2. A summary of some recent developments for UAVs in control, perception, SLAM and motion planning.

References	Control	Perception	SLAM	Motion Planning	Exploration
[192]		✓			
[55,57,60–63,70–73,75,77–86,90,92–94,96,101,104,115,117–121,125–128,130,132,136–148]				✓	
[59,71,134,149–151,193]		✓		✓	
[58,194–196]				✓	✓
[17–19,48,153,154,158–164,171,173,176]	✓				
[178,180–191]			✓		
[56,68,69,74,91,102,103,105,129,131,135]	✓			✓	
[100]	✓	✓		✓	
[197]	✓	✓	✓		
[88,198,199]	✓	✓	✓	✓	
[200]	✓	✓	✓	✓	✓

7. Open-Source Projects

There have been many developments in the field of UAVs in terms of perception, control, SLAM and path planning over recent years. Implementing a complete autonomous navigation stack would require a large team with different skill sets in these areas or collaborations among research groups. Moreover, a lot of time needs to be invested in implementation and dealing with technical issues to ensure the reliability of the overall system. Open-source projects contributed by many researchers have made it possible for others in the community to focus on the development and improvement of a specific navigation component related to their research while easily integrating with other components made available by researchers, saving a lot of development time. Table 3 shows a list of some existing open-source projects and tools useful for autonomous UAV research and development.

Table 3. Open-source projects and tools for UAV development.

	Name	Description	Source
Navigation Stack	Vision-based navigation for MAVs [201]	provides an open-source system for MAVs based on vision-based sensors including control, sensor fusion, mapping, local and global planning	http://github.com/ethz-asl/voxblox http://github.com/ethz-asl/rovio http://github.com/ethz-asl/ethzasl_msf http://github.com/ethz-asl/odom_predictor http://github.com/ethz-asl/maplab http://github.com/ethz-asl/mav_control_rw
	PULP-DroNet [202]	a deep learning-powered visual navigation engine for nano-UAVs	https://github.com/pulp-platform/pulp-dronet
LiDAR-based SLAM	Google's Cartographer [181]	provides a real-time SLAM solution in 2D and 3D	https://github.com/cartographer-project/cartographer
	hdl_graph_slam [182]	a real-time 6DOF SLAM using a 3D LIDAR	https://github.com/koide3/hdl_graph_slam
	loam_velodyne [180]	Laser Odometry and Mapping	https://github.com/laboshin/loam_velodyne
	A-LOAM	Advanced implementation of LOAM	https://github.com/HKUST-Aerial-Robotics/A-LOAM
	FLOAM	a faster and optimized version of A-LOAM and LOAM	https://github.com/wh200720041/floam
Vision-based SLAM	ORB SLAM [186]	a keyframe and feature-based Monocular SLAM	https://openslam-org.github.io/orbslam.html
	ORB SLAM 2 [187]	a real-time SLAM library for Monocular, Stereo and RGB-D cameras	https://github.com/raulmur/ORB_SLAM2
	LSD-SLAM [188]	a Large-Scale Direct Monocular SLAM system	https://github.com/tum-vision/lsd_slam
	SVO Semi-direct Visual Odometry [203]	a semi-direct monocular visual SLAM	https://github.com/uzh-rpg/rpg_svo
	PTAM [185]	a monocular SLAM system	https://github.com/Oxford-PTAM/PTAM-GPL
	RTAB-Map [204,205]	RGB-D, Stereo and Lidar Graph-Based SLAM algorithm	http://introlab.github.io/rtabmap
	ElasticFusion [206]	Real-time dense visual SLAM system using RGB-D cameras	https://github.com/mp3guy/ElasticFusion
	Kintinuous [190]	Real-time dense visual SLAM system using RGB-D cameras	https://github.com/mp3guy/Kintinuous
Motion Planning	Fast-Planner [87]	a set of planning algorithms for fast flights with quadrotors in complex unknown environments	https://github.com/HKUST-Aerial-Robotics/Fast-Planner
	FUEL [207]	a hierarchical framework for Fast UAV Exploration	https://github.com/HKUST-Aerial-Robotics/FUEL
	EGO-Planner	Gradient-based Local Planner for Quadrotors	https://github.com/ZJU-FAST-Lab/ego-planner
	TopoTraj [208]	a robust planner for quadrotor trajectory replanning based on gradient-based trajectory optimization	https://github.com/HKUST-Aerial-Robotics/TopoTraj
	toppra [209]	a library for computing time-optimal trajectories subject to kinematic and dynamic constraints	https://github.com/hungpham2511/toppra
	Open Motion Planning Library	a library for sampling-based motion planning algorithms	https://ompl.kavrakilab.org/core/index.html
	AIKIDO	a C++ library for motion planning and decision making problems	https://github.com/personalrobotics/aikido
	PathPlanning	a collection of search-based and sampling-based path planners implemented in Python	https://github.com/zhm-real/PathPlanning

Table 3. Cont.

	Name	Description	Source
Control	mav_control_rw [164]	Linear and nonlinear MPC controllers for Micro Aerial Vehicles	https://github.com/ethz-asl/mav_control_rw
	rpg_mpc [102]	Perception-Aware MPC for quadrotors	https://github.com/uzh-rpg/rpg_mpc
	ACADO Toolkit	collection of algorithms for automatic control and dynamic optimization	http://acado.github.io/
	Control Toolbox	a C++ library for robotics addressing control, estimation and motion planning	https://github.com/ethz-adrl/control-toolbox
	PX4	an open-source flight control software for UAVs	https://px4.io/
	ArduPilot	an open-source flight control software for UAVs	https://ardupilot.org/
Perception	Augmented Autoencoders [210]	3D object detection pipeline from RGB images	https://github.com/DLR-RM/AugmentedAutoencoder
	MoreFusion [211]	a perception pipeline for 6D pose estimations of multi-objects	https://github.com/wkentaro/morefusion
	OpenCV	an optimized computer vision library	https://opencv.org/
	Point Cloud Library (PCL)	efficient point cloud processing C++ library	https://pointclouds.org/
	cilantro [212]	efficient point cloud processing C++ library	https://github.com/kzampog/cilantro
Simulators	Gazebo	a robot simulator	http://gazebo.sim.org/
	CoppeliaSim/V-REP	a robot simulator	https://www.coppeliarobotics.com/
	Webots	a robot simulator	https://cyberbotics.com/
	Hector Quadrotor	provides simulation tools for quadrotors (ROS-based)	http://wiki.ros.org/hector_quadrotor
	RotorS [213]	a set of tools to simulate multi-rotors in Gazebo	https://github.com/ethz-asl/rotors_simulator
General	Robot Operating System (ROS)	a middleware to facilitate building large robotic applications	https://www.ros.org/
	Ceres Solver	a C++ library for solving large optimization problems	http://ceres-solver.org/
	g2o	a C++ framework for graph-based nonlinear optimization	https://github.com/RainerKuemmerle/g2o
	NLopt	a nonlinear optimization library	https://nlopt.readthedocs.io/en/latest/
	Optimization Engine (OpEn)	a fast solver for optimization problems in robotics	https://nlopt.readthedocs.io/en/latest/
	Interior Point OPTimizer (Ipopt) [214]	a software and library for solving large-scale nonlinear optimization problems	https://github.com/coin-or/Ipopt
	SNOPT	an optimizer for large-scale nonlinear optimization problems	https://ampl.com/products/solvers/solvers-we-sell/snopt
	ifopt [215]	a light-weight C++ interface to Nonlinear Programming Solvers Ipopt and Snopt	https://github.com/ethz-adrl/ifopt

8. Research Challenges

The navigation problem for UAVs remains a very challenging one due to the wide range of tasks they are needed for. Navigating in unknown and highly dynamic environments is one of the most challenging problems, especially for micro-UAVs with limited payload capacity and onboard computation capabilities. Some of the existing 3D reactive navigation approaches were developed based on conservative assumptions about obstacles. Similarly, map-based local trajectory planning methods tend to simplify the problem by relaxing the collision avoidance constraints to make the optimization problem more tractable in real-time. Overall, more theoretically well-founded, and computationally efficient navigation solutions are needed to provide a high level of safety guarantees.

Moreover, map-based navigation approaches are highly dependent on the performance of the localization system where a lot of ongoing research is focused on that area. Due to the limited payload capacity of small and micro-UAVs, vision sensors are the main source of information for localization and obstacle detection. However, this can be challenging in textureless environments. Furthermore, the small FOV provided by these sensors encourages more research in developing perception-aware navigation methods (ex. see [80,101–105]) to be able to maintain information about obstacles during avoidance maneuvers.

Developing fully autonomous UAVs targeting specific applications may involve additional layers of control; therefore, some of the applications where UAVs are used or can be potentially deployed are provided in the next section to bring to the reader's attention the complexity and potential challenges in these areas. Moreover, using multiple UAVs in collaboration to carry out tasks can increase efficiency and reduce mission time. This makes the navigation and control problems of multi-vehicle UAVs an active field of research due to the higher complexity of these problems, which is highlighted in Section 8.2.

8.1. UAV Applications

As UAVs continue to emerge in new applications, new challenges arise based on the complexity of required tasks. New developments in technologies related to UAVs can also open research directions to develop new advanced navigation algorithms that would not have been possible with existing older technologies. Over the past decade, UAVs have been utilized in many applications. However, many of these applications still do not adopt fully autonomous solutions due to the involved operational risks and immaturity of research related to some of these particular applications. Thus, this section explores different areas where UAVs are currently used or needed to attract more interest in using UAVs and to guide further developments for UAV technologies supporting these applications with increased autonomy levels.

8.1.1. Precision Agriculture

Precision Agriculture (PA) has attracted a lot of interest recently with a main goal of applying efficient solutions or resource management of soil and crops using different means of technology. UAVs offer great mobile solutions to increase productivity and to save resources in this area. Example applications related to PA include remote sensing [216–218], mapping [219,220], pests control [221,222], weed control [219,223–225] and harvesting [226].

Using UAVs for remote sensing applications in PA provides inspection data at higher temporal and spatial resolution than satellite imagery [227]. UAVs can be equipped with different sensors to provide rich information about soil condition, crop growth and plants biomass and vigor. For example, thermal and RGB images collected by a UAV can help farmers identify crop water stress. Similarly, images collected from multi-spectral and hyperspectral cameras can be used to determine vegetation indices, which is a good way for continuous monitoring of crop variability and stress conditions [228]. These cameras are very expensive compared to thermal and RGB cameras, which can be a limiting factor in some cases.

In remote sensing applications, coverage path planning algorithms are normally applied generating optimal paths (ex. back-and-forth motion patterns) to survey areas of interest. For high-altitude flights, the coverage path planning problem can be simply solved using classical approaches assuming an obstacle-free flight space. In this case, simple path following control can be applied. This problem becomes more complex on the high level coverage planning when considering no-flight zones (i.e., obstacles), multi-UAV cases and low-altitude flights. These cases also require proper local motion planning to handle dynamic obstacles, such as people and other noncooperative UAVs, and static obstacles such as trees, buildings, etc. Moreover, advanced control methods are needed to perform autonomous tasks such as harvesting, irrigating and weed control.

8.1.2. Search and Rescue

Search and Rescue (SAR) operations have evolved over the years in considering robotic aid for improved results. This attracted attention to the field of Disaster Robotics. UAVs can add great value to SAR missions by carrying out different tasks including: localizing and tracking victims; survivors' situation and environment assessments; delivering aid kits such as first aid and self-inflating emergency flotation devices; communicating messages from rescue teams to victims; providing wireless communication networks between SAR

teams in remote inaccessible areas [229]. Clearly, the system will vary depending on the delegated task. For example, UAVs equipped with color and/or thermal cameras are used to localize victims either manually by providing visual feedback to a remote operator or autonomously using computer vision algorithms with proper onboard GPU power. On the other hand, UAVs used to deliver aid kits need to have higher payload and aerial manipulation capability. Providing wireless communication networks can be valuable in marine SAR missions where it is hard to set up ground networks.

Developing fully autonomous SAR-enabled UAVs requires suitable navigation strategies depending on the environment (indoors or outdoors) and the required motion objectives. For example, some cases require UAVs to survey areas of concern where the application of coverage motion planning algorithms is needed similar to remote sensing applications. 3D exploration techniques can also be applied in indoors environments which is still an active research problem. Navigation in harsh indoor environments, such as tunnels and collapsed buildings, adds more challenges towards achieving fully autonomous operations. For example, there is a need for suitable control methods to handle flights in confined spaces, well developed SLAM methods to deal with poor conditions, and proper perception and obstacle avoidance capabilities. Solutions based on the use of cooperative UAVs in SAR missions are also attractive but require further development due to the higher complexity of these solutions.

Examples of UAV-based solutions for SAR applications in different environments and scenarios such as: remote disaster areas and wilderness SAR [1,230–232]; urban SAR (ex. collapsing buildings) [233–235]; underground tunnels [236,237]; and marine SAR [238–241].

8.1.3. Animal Control and Wildlife Monitoring

Another area where UAVs can be very useful is livestock and wildlife monitoring. For example, UAVs can be used to count, classify, and track livestock animals [242,243] to optimize hunting and harvesting in farms. To achieve these tasks, UAVs need to be equipped with proper sensors such as RGB or thermal cameras. Monitoring wildlife is also important to manage the population of threatened and invasive species (see [244–246]). Moreover, UAVs have also been considered for detecting and tracking some marine wildlife swimming close to the surface such as hammerhead sharks [247]. Depending on the environment and UAV sensors configuration, perception-aware trajectory planning methods can be applied in the case of animal tracking to make sure that the targeted animal remains within its field of view. Another interesting application is the use of UAVs for herding of birds [248] and farm animals [249]. In many of these applications, the UAV is usually flying at a higher altitude than the targets, which makes it valid to assume that the flight space is less crowded with obstacles. However, complete autonomous solutions for herding require more development for motion planning as the problem relies on the dynamical behavior of the animals. For example, motion planning can be combined with prediction methods to predict animals' future trajectories. However, it is important to consider some challenging factors in these applications such as the effect of UAV sound on wildlife under study [250] and other general effects which requires a code of practice for the use of UAVs in biological field research [251].

8.1.4. Weather Forecast

The low cost of UAVs makes them good tools to collect more information about hurricanes and tornadoes using dedicated sensors. Such collected data can help scientists to build better understanding of the trajectories of hurricanes and tornadoes. UAVs can also provide advanced warning and damage assessment for thunderstorms and tornadoes. There are few works in this area such as [252–255].

8.1.5. Construction

In the construction domain, deployment of UAVs in construction sites is good for aiding high-level management. Applications of UAVs in construction include [256]: build-

ing inspection; post-disaster damage assessment; site surveying and mapping; safety inspection; and progress monitoring. For example, see [256–260] and references therein.

Navigating in construction sites can be challenging due to being highly dynamic and crowded environments. Some tasks may also require UAVs to fly very close to buildings such as facade inspection and scanning buildings to build 3D models. This can be a potential application for reactive methods to maintain a certain distance from the building.

8.1.6. Oil and Gas

UAVs have also started to attract interest in the oil and gas industry. For example, UAV-based magnetic surveys can be used to detect and identify abandoned wells [261]. The use of UAVs for monitoring and inspection of oil and gas pipeline networks can aid in preventing failures, detecting problems over time and performing repair activities [262–264]. Another possible application is the detection of gas leaks where UAVs with in situ sensors can be used [265].

8.1.7. Other

Other UAV applications include: solar panel inspection [266,267]; power line and tower inspections [268,269]; water quality monitoring [270]; magnetic field mapping [271]; load transportation [272,273]; contact inspection tasks [274]; road safety and traffic monitoring [275–277]; aerial photography and cinematography [278]; entertainment and aerial shows [279–282]; firefighting [283–285].

8.2. Multi-UAV and Networked Systems

The use of multi-UAV systems has become more desirable in many applications for improved performance; thus, developing these systems is currently a very active field of research. Various challenges arise in this area to develop autonomous UAV swarms in tasks assignments, communication, trajectory planning and coordinated control. Different interactions between the vehicles are needed in order to collaboratively achieve a global objective assigned to the group.

Algorithms developed for Multi-UAV systems can be either *centralized*, *decentralized* or *distributed*. *Centralized* approaches are implemented on a central computer where trajectories are computed for all agents/vehicles within the system. This requires measurements from all agents to be available to the central computer. Centralized algorithms can produce globally optimal solutions; however, the overall system is prone to failure if the central computer fails, or a communication problem occurs. On the other hand, *decentralized* and *distributed* algorithms offer more robustness and scalability with a more computationally efficient solution to systems with large number of vehicles. These approaches enable each vehicle to compute their own trajectories and control actions based on local interactions with neighboring UAVs either by relying directly on its sensors' measurements (*decentralized*) or by combining these with information communicated by neighbor vehicles (*distributed*). In other words, distributed methods distribute the computation and communication load among the vehicles [286] to collaboratively serve a global group objective.

Formation control is one of the challenging areas for multi-UAV systems where each vehicle has a specific role with some constraints on its states [287] to achieve global group objective(s). The common formation control structures in the literature are *leader-follower*, *virtual* and *behavioral-based structures* [288]. A physical vehicle is set to be followed by the remaining vehicles within the system in a certain manner in *leader-follower* structures such as [289–295]. On the contrary, approaches with *virtual structure* achieves motion formation through forcing each vehicle to follow a corresponding virtual target (or reference trajectory) such that the selection of these virtual references contributes towards the global objective; for example, see [296–301]. *Behavioral-based* structure comprises of a set of rules followed by the vehicles contributing towards the collective behavior; flocking control, based on artificial potential fields, belongs to this category where a group of interacting agents move together to achieve some global objectives. One of the early models captur-

ing the local interactions between agents under a flocking behavior is Reynolds' model for the aggregate motion of flocks, which is based on three main rules: flock centering (cohesion), collision avoidance (separation) and velocity matching (alignment) [302,303]. Several works have addressed the flocking problem such as [303–311]. Different simplifications are normally made to deal with the high dimensionality of this problem, which includes considering simpler motion models such as single integrator [309,312,313], double integrator [303,304,311,314], nonholonomic models [305–308,315] and Euler–Lagrangian systems [310,316].

Many works have tackled the 3D motion coordination and collision avoidance problem for UAV swarms through a hierarchical approach with trajectory planning formulated as an optimization problem, similar to what was discussed earlier for single-UAV systems. Different centralized, decentralized and distributed approaches can be seen in the literature considering homogeneous and heterogeneous multi-UAV systems in addition to considering multi-vehicle systems working with aerial and ground vehicles; for example, see [79,317–327].

An example task that can be carried out by UAV swarms is carrying and transporting objects individually [168,328–330] or collaboratively [295,331–335]. Other applications where multi-UAV systems were deployed include distributed target search and tracking [336–339], distributed monitoring and surveillance [340–343] and cooperative mapping [344,345]. The research on cooperative mapping in unknown environments, or swarm SLAM, is not mature yet with not enough established methodologies according to [346], which motivates more research in this promising area.

It is also very important to keep in mind communication challenges when designing control methods for large-scale networked multi-vehicle systems. Application of Networked Control Systems (NCSs) theory is thus suitable when analyzing the operation of networks of collaborating autonomous UAVs [347–352]. This can help considering additional practical aspects of NCSs in the overall system design such as delays introduced in communication channels [347,353,354], noises [355,356], loss/corruption of data [353,354] and bandwidth constraints [357–360].

9. Conclusions

Rapid advances in UAV-related technologies allowed great development towards achieving fully autonomous operations in the areas of control, motion planning, perception and localization and mapping. This paper presented a survey about some recent advancements in these areas focusing more on allowing advanced autonomous 3D collision-free navigation for UAVs. The main differences between the adopted motion planning algorithms and control strategies were also highlighted, showing the advantages and disadvantages between different methods. This can provide guidance for researchers to determine suitable navigation methods based on their specific applications. Moreover, a list of some existing open-source projects was provided to aid researchers in quickly developing and deploying innovative technologies for UAVs. Developing fully autonomous UAVs remains very challenging due to the wide range of new applications and the various levels of the tasks' complexity. Hence, active research challenges were also highlighted in this paper, including recent developments in motion control for multi-UAV systems. Additionally, several applications were mentioned to encourage the development of more mature fully autonomous solutions dedicated to emerging UAV applications.

Author Contributions: Conceptualization, T.E. and A.V.S.; methodology, T.E.; validation, T.E.; formal analysis, T.E.; resources, A.V.S.; writing—original draft preparation, T.E.; writing—review and editing, A.V.S.; visualization, T.E.; supervision, A.V.S.; project administration, A.V.S.; funding acquisition, A.V.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Australian Research Council. This work also received funding from the Australian Government, via grant AUSMURIB000001 associated with ONR MURI grant N00014-19-1-2571.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

References

1. Goodrich, M.A.; Morse, B.S.; Gerhardt, D.; Cooper, J.L.; Quigley, M.; Adams, J.A.; Humphrey, C. Supporting wilderness search and rescue using a camera-equipped mini UAV. *J. Field Robot.* **2008**, *25*, 89–110. [[CrossRef](#)]
2. 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](#)]
3. Huang, H.; Savkin, A.V. Towards the Internet of Flying Robots: A Survey. *Sensors* **2018**, *18*, 4038. [[CrossRef](#)] [[PubMed](#)]
4. Pajares, G. Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *Photogramm. Eng. Remote Sens.* **2015**, *81*, 281–330. [[CrossRef](#)]
5. Savkin, A.V.; Huang, H. A method for optimized deployment of a network of surveillance aerial drones. *IEEE Syst. J.* **2019**, *13*, 4474–4477. [[CrossRef](#)]
6. Huang, H.; Savkin, A.V. An algorithm of reactive collision free 3-D deployment of networked unmanned aerial vehicles for surveillance and monitoring. *IEEE Trans. Ind. Inform.* **2020**, *16*, 132–140. [[CrossRef](#)]
7. Savkin, A.V.; Huang, H. Navigation of a network of aerial drones for monitoring a frontier of a moving environmental disaster area. *IEEE Syst. J.* **2020**, *14*, 4746–4749. [[CrossRef](#)]
8. Nex, F.; Remondino, F. UAV for 3D mapping applications: A review. *Appl. Geomat.* **2014**, *6*, 1–15. [[CrossRef](#)]
9. Korpela, C.M.; Danko, T.W.; Oh, P.Y. MM-UAV: Mobile manipulating unmanned aerial vehicle. *J. Intell. Robot. Syst.* **2012**, *65*, 93–101. [[CrossRef](#)]
10. Ruggiero, F.; Lippiello, V.; Ollero, A. Aerial manipulation: A literature review. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1957–1964. [[CrossRef](#)]
11. Li, H.; Savkin, A.V.; Vucetic, B. Autonomous Area Exploration and Mapping in Underground Mine Environments by Unmanned Aerial Vehicles. *Robotica* **2020**, *38*, 442–456. [[CrossRef](#)]
12. Elmokadem, T.; Savkin, A.V. A Method for Autonomous Collision-Free Navigation of a Quadrotor UAV in Unknown Tunnel-Like Environments; *Robotica* **2021**, 1–27. [[CrossRef](#)]
13. Roderick, W.R.; Cutkosky, M.R.; Lentink, D. Touchdown to take-off: At the interface of flight and surface locomotion. *Interface Focus* **2017**, *7*, 20160094. [[CrossRef](#)]
14. Valavanis, K.P.; Vachtsevanos, G.J. *Handbook of Unmanned Aerial Vehicles*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 1.
15. Cai, G.; Peng, K.; Chen, B.M.; Lee, T.H. Design and assembling of a UAV helicopter system. In Proceedings of the 2005 International Conference on Control and Automation (ICCA 2005), Budapest, Hungary, 26–29 June 2005; Volume 2; pp. 697–702.
16. Cai, G.; Feng, L.; Chen, B.M.; Lee, T.H. Systematic design methodology and construction of UAV helicopters. *Mechatronics* **2008**, *18*, 545–558. [[CrossRef](#)]
17. Godbolt, B.; Vitzilaios, N.I.; Lynch, A.F. Experimental validation of a helicopter autopilot design using model-based PID control. *J. Intell. Robot. Syst.* **2013**, *70*, 385–399. [[CrossRef](#)]
18. Cai, G.; Wang, B.; Chen, B.M.; Lee, T.H. Design and implementation of a flight control system for an unmanned rotorcraft using RPT control approach. *Asian J. Control* **2013**, *15*, 95–119. [[CrossRef](#)]
19. Mahony, R.; Kumar, V.; Corke, P. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robot. Autom. Mag.* **2012**, *19*, 20–32. [[CrossRef](#)]
20. Phang, S.K.; Li, K.; Yu, K.H.; Chen, B.M.; Lee, T.H. Systematic design and implementation of a micro unmanned quadrotor system. *Unmanned Syst.* **2014**, *2*, 121–141. [[CrossRef](#)]
21. Segui-Gasco, P.; Al-Rihani, Y.; Shin, H.S.; Savvaris, A. A novel actuation concept for a multi rotor UAV. *J. Intell. Robot. Syst.* **2014**, *74*, 173–191. [[CrossRef](#)]
22. Verbeke, J.; Hulens, D.; Ramon, H.; Goedeme, T.; De Schutter, J. The design and construction of a high endurance hexacopter suited for narrow corridors. In Proceedings of the 2014 International Conference on Unmanned Aircraft Systems (ICUAS 2014), Orlando, FL, USA, 27–30 May 2014; pp. 543–551.
23. Kamel, M.; Verling, S.; Elkhatib, O.; Sprecher, C.; Wulkop, P.; Taylor, Z.; Siegwart, R.; Gilitschenski, I. The voliro omniorientational hexacopter: An agile and maneuverable tilttable-rotor aerial vehicle. *IEEE Robot. Autom. Mag.* **2018**, *25*, 34–44. [[CrossRef](#)]
24. Rashad, R.; Goerres, J.; Aarts, R.; Engelen, J.B.; Stramigioli, S. Fully actuated multirotor UAVs: A literature review. *IEEE Robot. Autom. Mag.* **2020**, *27*, 97–107. [[CrossRef](#)]
25. Shkarayev, S.V.; Ifju, P.G.; Kellogg, J.C.; Mueller, T.J. *Introduction to the Design of Fixed-Wing Micro Air Vehicles Including Three Case Studies*; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2007.
26. Keane, A.J.; Söbester, A.; Scanlan, J.P. *Small Unmanned Fixed-Wing Aircraft Design: A Practical Approach*; John Wiley & Sons: Hoboken, NJ, USA, 2017.
27. Zhao, A.; He, D.; Wen, D. Structural design and experimental verification of a novel split aileron wing. *Aerosp. Sci. Technol.* **2020**, *98*, 105635. [[CrossRef](#)]

28. Cetinsoy, E.; Dikyar, S.; Hançer, C.; Oner, K.; Sirimoglu, E.; Unel, M.; Aksit, M. Design and construction of a novel quad tilt-wing UAV. *Mechatronics* **2012**, *22*, 723–745. [[CrossRef](#)]
29. Ozdemir, U.; Aktas, Y.O.; Vuruskan, A.; Dereli, Y.; Tarhan, A.F.; Demirbag, K.; Erdem, A.; Kalaycioglu, G.D.; Ozkol, I.; Inalhan, G. Design of a commercial hybrid VTOL UAV system. *J. Intell. Robot. Syst.* **2014**, *74*, 371–393. [[CrossRef](#)]
30. Ke, Y.; Wang, K.; Chen, B.M. Design and implementation of a hybrid UAV with model-based flight capabilities. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 1114–1125. [[CrossRef](#)]
31. Chipade, V.S.; Kothari, M.; Chaudhari, R.R. Systematic design methodology for development and flight testing of a variable pitch quadrotor biplane VTOL UAV for payload delivery. *Mechatronics* **2018**, *55*, 94–114. [[CrossRef](#)]
32. Gerdes, J.W.; Gupta, S.K.; Wilkerson, S.A. A review of bird-inspired flapping wing miniature air vehicle designs. *J. Mech. Robot.* **2012**, *4*, 021003. [[CrossRef](#)]
33. Karásek, M. *Robotic Hummingbird: Design of a Control Mechanism for a Hovering Flapping Wing Micro Air Vehicle*; Universite Libre de Bruxelles: Bruxelles, Belgium, 2014.
34. Gerdes, J.; Holness, A.; Perez-Rosado, A.; Roberts, L.; Greisinger, A.; Barnett, E.; Kempny, J.; Lingam, D.; Yeh, C.H.; Bruck, H.A.; et al. Robo Raven: A flapping-wing air vehicle with highly compliant and independently controlled wings. *Soft Robot.* **2014**, *1*, 275–288. [[CrossRef](#)]
35. Hassanalian, M.; Abdelkefi, A.; Wei, M.; Ziaei-Rad, S. A novel methodology for wing sizing of bio-inspired flapping wing micro air vehicles: Theory and prototype. *Acta Mech.* **2017**, *228*, 1097–1113. [[CrossRef](#)]
36. İşbitirici, A.; Altuğ, E. Design and control of a mini aerial vehicle that has four flapping-wings. *J. Intell. Robot. Syst.* **2017**, *88*, 247–265. [[CrossRef](#)]
37. Holness, A.E.; Bruck, H.A.; Gupta, S.K. Characterizing and modeling the enhancement of lift and payload capacity resulting from thrust augmentation in a propeller-assisted flapping wing air vehicle. *Int. J. Micro Air Veh.* **2018**, *10*, 50–69. [[CrossRef](#)]
38. Yousaf, R.; Shahzad, A.; Qadri, M.M.; Javed, A. Recent advancements in flapping mechanism and wing design of micro aerial vehicles. *Proc. Inst. Mech. Eng. Part J. Mech. Eng. Sci.* **2020**. [[CrossRef](#)]
39. Kaufman, E.; Caldwell, K.; Lee, D.; Lee, T. Design and development of a free-floating hexrotor UAV for 6-DOF maneuvers. In Proceedings of the 2014 IEEE Aerospace Conference, Big Sky, MT, USA, 1–8 March 2014; pp. 1–10.
40. Allenspach, M.; Bodie, K.; Brunner, M.; Rinsoz, L.; Taylor, Z.; Kamel, M.; Siegwart, R.; Nieto, J. Design and optimal control of a tiltrotor micro-aerial vehicle for efficient omnidirectional flight. *Int. J. Robot. Res.* **2020**, *39*, 1305–1325. [[CrossRef](#)]
41. Stewart, W.; Weisler, W.; MacLeod, M.; Powers, T.; Defreitas, A.; Gritter, R.; Anderson, M.; Peters, K.; Gopalathnam, A.; Bryant, M. Design and demonstration of a seabird-inspired fixed-wing hybrid UAV-UUV system. *Bioinspir. Biomimetics* **2018**, *13*, 056013. [[CrossRef](#)] [[PubMed](#)]
42. Stewart, W.; Weisler, W.; Anderson, M.; Bryant, M.; Peters, K. Dynamic modeling of passively draining structures for aerial-aquatic unmanned vehicles. *IEEE J. Ocean. Eng.* **2019**, *45*, 840–850. [[CrossRef](#)]
43. Tan, Y.H.; Chen, B.M. Design of a morphable multirotor aerial-aquatic vehicle. In Proceedings of the OCEANS 2019 MTS/IEEE SEATTLE, Seattle, WA, USA, 27–31 October 2019; pp. 1–8.
44. Kalantari, A.; Spenko, M. Modeling and performance assessment of the HyTAQ, a hybrid terrestrial/aerial quadrotor. *IEEE Trans. Robot.* **2014**, *30*, 1278–1285. [[CrossRef](#)]
45. Mulgaonkar, Y.; Araki, B.; Koh, J.S.; Guerrero-Bonilla, L.; Aukes, D.M.; Mäkinen, A.; Tolley, M.T.; Rus, D.; Wood, R.J.; Kumar, V. The flying monkey: A mesoscale robot that can run, fly, and grasp. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA 2016), Stockholm, Sweden, 16–21 May 2016; pp. 4672–4679.
46. Yamada, M.; Nakao, M.; Hada, Y.; Sawasaki, N. Development and field test of novel two-wheeled UAV for bridge inspections. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems (ICUAS 2017), Miami, FL, USA, 13–16 June 2017; pp. 1014–1021.
47. Sabet, S.; Agha-Mohammadi, A.A.; Tagliabue, A.; Elliott, D.S.; Nikravesh, P.E. Rollocopter: An energy-aware hybrid aerial-ground mobility for extreme terrains. In Proceedings of the 2019 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2019; pp. 1–8.
48. Atay, S.; Bryant, M.; Buckner, G.D. The Spherical Rolling-Flying Vehicle: Dynamic Modeling and Control System Design. *J. Mech. Robot.* **2021**, *13*, 050901. [[CrossRef](#)]
49. Huang, H.M. *Autonomy Levels for Unmanned Systems (ALFUS) Framework Volume I: Terminology*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2004.
50. LaValle, S.M. *Planning Algorithms*; Cambridge University Press: Cambridge, UK, 2006.
51. Hoy, M.; Matveev, A.S.; Savkin, A.V. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey. *Robotica* **2015**, *33*, 463–497. [[CrossRef](#)]
52. Tobaruela, J.A.; Rodríguez, A.O. Reactive navigation in extremely dense and highly intricate environments. *PLoS ONE* **2017**, *12*, e0189008.
53. DeSouza, G.N.; Kak, A.C. Vision for mobile robot navigation: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 237–267. [[CrossRef](#)]
54. Bonin-Font, F.; Ortiz, A.; Oliver, G. Visual Navigation for Mobile Robots: A survey. *J. Intell. Robot. Syst.* **2008**, *53*, 263–296. [[CrossRef](#)]
55. Aoude, G.S.; Luders, B.D.; Joseph, J.M.; Roy, N.; How, J.P. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Auton. Robot.* **2013**, *35*, 51–76. [[CrossRef](#)]

56. Yang, K.; Gan, S.K.; Sukkarieh, S. An Efficient Path Planning and Control Algorithm for RUAV's in Unknown and Cluttered Environments. *J. Intell. Robot. Syst.* **2010**, *57*, 101. [[CrossRef](#)]
57. Lin, Y.; Saripalli, S. Sampling-based path planning for UAV collision avoidance. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 3179–3192. [[CrossRef](#)]
58. Schmid, L.; Pantic, M.; Khanna, R.; Ott, L.; Siegwart, R.; Nieto, J. An efficient sampling-based method for online informative path planning in unknown environments. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1500–1507. [[CrossRef](#)]
59. Liu, S.; Watterson, M.; Tang, S.; Kumar, V. High speed navigation for quadrotors with limited onboard sensing. In Proceedings of the 2016 IEEE international conference on robotics and automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1484–1491.
60. Sanchez-Lopez, J.L.; Wang, M.; Olivares-Mendez, M.A.; Molina, M.; Voos, H. A Real-Time 3D Path Planning Solution for Collision-Free Navigation of Multirotor Aerial Robots in Dynamic Environments. *J. Intell. Robot. Syst.* **2019**, *93*, 33–53. [[CrossRef](#)]
61. Miller, B.; Stepanyan, K.; Miller, A.; Andreev, M. 3D path planning in a threat environment. In Proceedings of the 2011 50th IEEE Conference on Decision and Control and European Control Conference, Orlando, FL, USA, 12–15 December 2011; pp. 6864–6869.
62. Chen, Y.b.; Luo, G.c.; Mei, Y.s.; Yu, J.q.; Su, X.l. UAV path planning using artificial potential field method updated by optimal control theory. *Int. J. Syst. Sci.* **2016**, *47*, 1407–1420. [[CrossRef](#)]
63. Roberge, V.; Tarbouchi, M.; Labonté, G. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Trans. Ind. Inform.* **2012**, *9*, 132–141. [[CrossRef](#)]
64. Roberge, V.; Tarbouchi, M.; Labonté, G. Fast genetic algorithm path planner for fixed-wing military UAV using GPU. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 2105–2117. [[CrossRef](#)]
65. Sahingoz, O.K. Generation of bezier curve-based flyable trajectories for multi-UAV systems with parallel genetic algorithm. *J. Intell. Robot. Syst.* **2014**, *74*, 499–511. [[CrossRef](#)]
66. Yang, K.; Sukkarieh, S. An analytical continuous-curvature path-smoothing algorithm. *IEEE Trans. Robot.* **2010**, *26*, 561–568. [[CrossRef](#)]
67. Gasparetto, A.; Boscariol, P.; Lanzutti, A.; Vidoni, R. Path planning and trajectory planning algorithms: A general overview. *Motion Oper. Plan. Robot. Syst.* **2015**, *29*, 3–27.
68. Mellinger, D.; Kumar, V. Minimum snap trajectory generation and control for quadrotors. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 2520–2525.
69. Richter, C.; Bry, A.; Roy, N. Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In *Robotics Research*; Springer International Publishing: Basel, Switzerland, 2016; pp. 649–666.
70. Deits, R.; Tedrake, R. Efficient mixed-integer planning for UAVs in cluttered environments. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 25–30 May 2015; pp. 42–49.
71. Oleynikova, H.; Burri, M.; Taylor, Z.; Nieto, J.; Siegwart, R.; Galceran, E. Continuous-time trajectory optimization for online UAV replanning. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 5332–5339. [[CrossRef](#)]
72. Liu, S.; Watterson, M.; Mohta, K.; Sun, K.; Bhattacharya, S.; Taylor, C.J.; Kumar, V. Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-d complex environments. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1688–1695. [[CrossRef](#)]
73. Watterson, M.; Kumar, V. Safe receding horizon control for aggressive MAV flight with limited range sensing. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–3 October 2015; pp. 3235–3240.
74. Liu, Y.; Rajappa, S.; Montenbruck, J.M.; Stegagno, P.; Bühlhoff, H.; Allgöwer, F.; Zell, A. Robust nonlinear control approach to nontrivial maneuvers and obstacle avoidance for quadrotor UAV under disturbances. *Robot. Autom. Syst.* **2017**, *98*, 317–332. [[CrossRef](#)]
75. Spedicato, S.; Notarstefano, G. Minimum-time trajectory generation for quadrotors in constrained environments. *IEEE Trans. Control Syst. Technol.* **2017**, *26*, 1335–1344. [[CrossRef](#)]
76. Gao, F.; Wu, W.; Lin, Y.; Shen, S. Online safe trajectory generation for quadrotors using fast marching method and bernstein basis polynomial. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–26 May 2018; pp. 344–351.
77. Ryll, M.; Ware, J.; Carter, J.; Roy, N. Efficient trajectory planning for high speed flight in unknown environments. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 732–738.
78. Tordesillas, J.; Lopez, B.T.; How, J.P. FASTER: Fast and Safe Trajectory Planner for Flights in Unknown Environments. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 4–8 November 2019.
79. Tordesillas, J.; How, J.P. MADER: Trajectory Planner in Multiagent and Dynamic Environments. *IEEE Trans. Robot.* **2021**. [[CrossRef](#)]
80. Tordesillas, J.; How, J.P. PANTHER: Perception-Aware Trajectory Planner in Dynamic Environments. *arXiv* **2021**, arXiv:2103.06372.
81. Chen, G.; Sun, D.; Dong, W.; Sheng, X.; Zhu, X.; Ding, H. Computationally Efficient Trajectory Planning for High Speed Obstacle Avoidance of a Quadrotor With Active Sensing. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3365–3372. [[CrossRef](#)]
82. Ye, H.; Zhou, X.; Wang, Z.; Xu, C.; Chu, J.; Gao, F. TGK-Planner: An Efficient Topology Guided Kinodynamic Planner for Autonomous Quadrotors. *IEEE Robot. Autom. Lett.* **2020**, *6*, 494–501. [[CrossRef](#)]

83. Bucki, N.; Lee, J.; Mueller, M.W. Rectangular Pyramid Partitioning Using Integrated Depth Sensors (RAPPIDS): A Fast Planner for Multicopter Navigation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4626–4633. [[CrossRef](#)]
84. Ji, J.; Wang, Z.; Wang, Y.; Xu, C.; Gao, F. Mapless-Planner: A Robust and Fast Planning Framework for Aggressive Autonomous Flight without Map Fusion. *arXiv* **2020**, arXiv:2011.03975.
85. Quan, L.; Zhang, Z.; Xu, C.; Gao, F. EVA-Planner: Environmental Adaptive Quadrotor Planning. *arXiv* **2020**, arXiv:2011.04246.
86. Lee, J.; Wu, X.; Lee, S.J.; Mueller, M.W. Autonomous flight through cluttered outdoor environments using a memoryless planner. *arXiv* **2021**, arXiv:2103.12156.
87. Zhou, B.; Gao, F.; Wang, L.; Liu, C.; Shen, S. Robust and efficient quadrotor trajectory generation for fast autonomous flight. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3529–3536. [[CrossRef](#)]
88. Mohta, K.; Watterson, M.; Mulgaonkar, Y.; Liu, S.; Qu, C.; Makineni, A.; Saulnier, K.; Sun, K.; Zhu, A.; Delmerico, J.; et al. Fast, autonomous flight in GPS-denied and cluttered environments. *J. Field Robot.* **2018**, *35*, 101–120. [[CrossRef](#)]
89. Deits, R.; Tedrake, R. Computing large convex regions of obstacle-free space through semidefinite programming. In *Algorithmic Foundations of Robotics XI*; Springer International Publishing: Basel, Switzerland, 2015; pp. 109–124.
90. Mueller, M.W.; Hehn, M.; D’Andrea, R. A computationally efficient motion primitive for quadcopter trajectory generation. *IEEE Trans. Robot.* **2015**, *31*, 1294–1310. [[CrossRef](#)]
91. Paranjape, A.A.; Meier, K.C.; Shi, X.; Chung, S.J.; Hutchinson, S. Motion primitives and 3D path planning for fast flight through a forest. *Int. J. Robot. Res.* **2015**, *34*, 357–377. [[CrossRef](#)]
92. Lopez, B.T.; How, J.P. Aggressive 3-D collision avoidance for high-speed navigation. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Marina Bay Sands, Singapore, 29 May–3 June 2017; pp. 5759–5765.
93. Tordesillas, J.; Lopez, B.T.; Carter, J.; Ware, J.; How, J.P. Real-time planning with multi-fidelity models for agile flights in unknown environments. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 725–731.
94. González-Sieira, A.; Cores, D.; Mucientes, M.; Bugarín, A. Autonomous navigation for UAVs managing motion and sensing uncertainty. *Robot. Auton. Syst.* **2020**, *126*, 103455. [[CrossRef](#)]
95. LaValle, S.M.; Kuffner Jr, J.J. Randomized Kinodynamic Planning. *Int. J. Robot. Res.* **2001**, *20*, 378–400. [[CrossRef](#)]
96. Lindqvist, B.; Mansouri, S.S.; Agha-mohammadi, A.a.; Nikolakopoulos, G. Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6001–6008. [[CrossRef](#)]
97. Sathya, A.; Sopasakis, P.; Van Parys, R.; Themelis, A.; Pipeleers, G.; Patrinos, P. Embedded nonlinear model predictive control for obstacle avoidance using PANOC. In Proceedings of the 2018 European Control Conference (ECC), Limassol, Cyprus, 12–15 June 2018; pp. 1523–1528.
98. Stella, L.; Themelis, A.; Sopasakis, P.; Patrinos, P. A simple and efficient algorithm for nonlinear model predictive control. In Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, Australia, 12–15 December 2017; pp. 1939–1944.
99. Sopasakis, P.; Fresk, E.; Patrinos, P. OpEn: Code generation for embedded nonconvex optimization. *arXiv* **2020**, arXiv:2003.00292.
100. Mansouri, S.S.; Kanellakis, C.; Lindqvist, B.; Pourkamali-Anaraki, F.; Agha-Mohammadi, A.A.; Burdick, J.; Nikolakopoulos, G. A Unified NMPC Scheme for MAVs Navigation with 3D Collision Avoidance Under Position Uncertainty. *IEEE Robot. Autom. Lett.* **2020**, *5*, 5740–5747. [[CrossRef](#)]
101. Zhang, Z.; Scaramuzza, D. Perception-aware receding horizon navigation for MAVs. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2534–2541.
102. Falanga, D.; Foehn, P.; Lu, P.; Scaramuzza, D. PAMPC: Perception-Aware Model Predictive Control for Quadrotors. In Proceedings of the 2018 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–8.
103. Murali, V.; Spasojevic, I.; Guerra, W.; Karaman, S. Perception-aware trajectory generation for aggressive quadrotor flight using differential flatness. In Proceedings of the 2019 American Control Conference (ACC), Philadelphia, PA, USA, 10–12 July 2019; pp. 3936–3943.
104. Spasojevic, I.; Murali, V.; Karaman, S. Perception-aware time optimal path parameterization for quadrotors. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Virtual, 31 May–15 June 2020; pp. 3213–3219.
105. Sheckells, M.; Garimella, G.; Kobilarov, M. Optimal visual servoing for differentially flat underactuated systems. In Proceedings of the 2016 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Korea, 9–14 October 2016; pp. 5541–5548.
106. Toibero, J.M.; Roberti, F.; Carelli, R. Stable contour-following control of wheeled mobile robots. *Robotica* **2009**, *27*, 1–12. [[CrossRef](#)]
107. Teimoori, H.; Savkin, A.V. A biologically inspired method for robot navigation in a cluttered environment. *Robotica* **2010**, *28*, 637–648. [[CrossRef](#)]
108. Matveev, A.S.; Teimoori, H.; Savkin, A.V. A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance. *Automatica* **2011**, *47*, 515–524. [[CrossRef](#)]
109. Matveev, A.S.; Wang, C.; Savkin, A.V. Real-time navigation of mobile robots in problems of border patrolling and avoiding collisions with moving and deforming obstacles. *Robot. Auton. Syst.* **2012**, *60*, 769–788. [[CrossRef](#)]
110. Savkin, A.V.; Wang, C. A simple biologically inspired algorithm for collision-free navigation of a unicycle-like robot in dynamic environments with moving obstacles. *Robotica* **2013**, *31*, 993–1001. [[CrossRef](#)]

111. Matveev, A.; Savkin, A.; Hoy, M.; Wang, C. *Safe Robot Navigation among Moving and Steady Obstacles*; Elsevier: Amsterdam, The Netherlands, 2015.
112. Choi, Y.; Jimenez, H.; Mavris, D.N. Two-layer obstacle collision avoidance with machine learning for more energy-efficient unmanned aircraft trajectories. *Robot. Auton. Syst.* **2017**, *98*, 158–173. [[CrossRef](#)]
113. McGuire, K.; De Croon, G.; De Wagter, C.; Tuyls, K.; Kappen, H. Efficient optical flow and stereo vision for velocity estimation and obstacle avoidance on an autonomous pocket drone. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1070–1076. [[CrossRef](#)]
114. Matveev, A.S.; Hoy, M.C.; Savkin, A.V. A Globally Converging Algorithm for Reactive Robot Navigation among Moving and Deforming Obstacles. *Automatica* **2015**, *54*, 292–304. [[CrossRef](#)]
115. Mujumdar, A.; Padhi, R. Reactive collision avoidance of using nonlinear geometric and differential geometric guidance. *J. Guid. Control. Dyn.* **2011**, *34*, 303–311. [[CrossRef](#)]
116. Wang, C.; Savkin, A.V.; Garratt, M. A strategy for safe 3D navigation of non-holonomic robots among moving obstacles. *Robotica* **2018**, *36*, 275–297. [[CrossRef](#)]
117. Lin, Z.; Castano, L.; Mortimer, E.; Xu, H. Fast 3D Collision Avoidance Algorithm for Fixed Wing UAS. *J. Intell. Robot. Syst.* **2020**, *97*, 577–604. [[CrossRef](#)]
118. Belkhouche, F.; Bendjilali, B. Reactive path planning for 3-d autonomous vehicles. *IEEE Trans. Control Syst. Technol.* **2012**, *20*, 249–256. [[CrossRef](#)]
119. Belkhouche, F. Reactive optimal UAV motion planning in a dynamic world. *Robot. Auton. Syst.* **2017**, *96*, 114–123. [[CrossRef](#)]
120. Wiig, M.S.; Pettersen, K.Y.; Krogstad, T.R. A 3D reactive collision avoidance algorithm for underactuated underwater vehicles. *J. Field Robot.* **2020**, *37*, 1094–1122. [[CrossRef](#)]
121. Wu, J.; Wang, H.; Zhang, M.; Yu, Y. On obstacle avoidance path planning in unknown 3D environments: A fluid-based framework. *ISA Trans.* **2021**, *111*, 249–264. [[CrossRef](#)]
122. Elmokadem, T. A 3D Reactive Collision Free Navigation Strategy for Nonholonomic Mobile Robots. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 4661–4666. [[CrossRef](#)]
123. Elmokadem, T. A Reactive Navigation Method of Quadrotor UAVs in Unknown Environments with Obstacles based on Differential-Flatness. In Proceedings of the 2019 Australasian Conference on Robotics and Automation (ACRA), Adelaide, Australia, 9–11 December 2019.
124. Elmokadem, T.; Savkin, A.V. A Hybrid Approach for Autonomous Collision-Free UAV Navigation in 3D Partially Unknown Dynamic Environments. *Drones* **2021**, *5*, 57. [[CrossRef](#)]
125. Yang, X.; Alvarez, L.M.; Bruggemann, T. A 3D collision avoidance strategy for UAVs in a non-cooperative environment. *J. Intell. Robot. Syst.* **2013**, *70*, 315–327. [[CrossRef](#)]
126. Tan, C.Y.; Huang, S.; Tan, K.K.; Teo, R.S.H. Three Dimensional Collision Avoidance for Multi Unmanned Aerial Vehicles using Velocity Obstacle. *J. Intell. Robot. Syst.* **2020**, *97*, 227–248. [[CrossRef](#)]
127. Zhu, L.; Cheng, X.; Yuan, F.G. A 3D collision avoidance strategy for UAV with physical constraints. *Measurement* **2016**, *77*, 40–49. [[CrossRef](#)]
128. Roussos, G.; Dimarogonas, D.V.; Kyriakopoulos, K.J. 3D navigation and collision avoidance for nonholonomic aircraft-like vehicles. *Int. J. Adapt. Control Signal Process.* **2010**, *24*, 900–920. [[CrossRef](#)]
129. Santos, M.C.P.; Rosales, C.D.; Sarcinelli-Filho, M.; Carelli, R. A novel null-space-based UAV trajectory tracking controller with collision avoidance. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 2543–2553. [[CrossRef](#)]
130. Hrabar, S. Reactive obstacle avoidance for rotorcraft UAVs. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 4967–4974.
131. Nguyen, P.D.; Recchiuto, C.T.; Sgorbissa, A. Real-time path generation and obstacle avoidance for multirotors: A novel approach. *J. Intell. Robot. Syst.* **2018**, *89*, 27–49. [[CrossRef](#)]
132. Iacono, M.; Sgorbissa, A. Path following and obstacle avoidance for an autonomous UAV using a depth camera. *Robot. Auton. Syst.* **2018**, *106*, 38–46. [[CrossRef](#)]
133. Elmokadem, T. A Control Strategy for the Safe Navigation of UAVs Among Dynamic Obstacles using Real-Time Deforming Trajectories. In Proceedings of the 2020 Australian and New Zealand Control Conference (ANZCC), Gold Coast, Australia, 26–27 November 2020; pp. 97–102.
134. Oleynikova, H.; Honegger, D.; Pollefeys, M. Reactive avoidance using embedded stereo vision for MAV flight. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 25–30 May 2015; pp. 50–56. [[CrossRef](#)]
135. Potena, C.; Nardi, D.; Pretto, A. Joint vision-based navigation, control and obstacle avoidance for UAVs in dynamic environments. In Proceedings of the 2019 European Conference on Mobile Robots (ECMR), Prague, Czech Republic, 4–6 September 2019; pp. 1–7.
136. Ross, S.; Melik-Barkhudarov, N.; Shankar, K.S.; Wendel, A.; Dey, D.; Bagnell, J.A.; Hebert, M. Learning monocular reactive UAV control in cluttered natural environments. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 1765–1772.
137. Zhang, B.; Mao, Z.; Liu, W.; Liu, J. Geometric reinforcement learning for path planning of UAVs. *J. Intell. Robot. Syst.* **2015**, *77*, 391–409. [[CrossRef](#)]

138. Wang, C.; Wang, J.; Zhang, X.; Zhang, X. Autonomous navigation of UAV in large-scale unknown complex environment with deep reinforcement learning. In Proceedings of the 2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Montreal, QC, Canada, 14–16 November 2017; pp. 858–862.
139. Ma, Z.; Wang, C.; Niu, Y.; Wang, X.; Shen, L. A saliency-based reinforcement learning approach for a UAV to avoid flying obstacles. *Robot. Auton. Syst.* **2018**, *100*, 108–118. [[CrossRef](#)]
140. Singla, A.; Padakandla, S.; Bhatnagar, S. Memory-based deep reinforcement learning for obstacle avoidance in UAV with limited environment knowledge. *IEEE Trans. Intell. Transp. Syst.* **2019**, *22*, 107–118. [[CrossRef](#)]
141. Walker, O.; Vanegas, F.; Gonzalez, F.; Koerig, S. A deep reinforcement learning framework for UAV navigation in indoor environments. In Proceedings of the 2019 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2019; pp. 1–14.
142. Yan, C.; Xiang, X.; Wang, C. Towards real-time path planning through deep reinforcement learning for a UAV in dynamic environments. *J. Intell. Robot. Syst.* **2019**, *98*, 297–309. [[CrossRef](#)]
143. Wang, C.; Wang, J.; Shen, Y.; Zhang, X. Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2124–2136. [[CrossRef](#)]
144. Padhy, R.P.; Verma, S.; Ahmad, S.; Choudhury, S.K.; Sa, P.K. Deep neural network for autonomous UAV navigation in indoor corridor environments. *Procedia Comput. Sci.* **2018**, *133*, 643–650. [[CrossRef](#)]
145. 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, 1–23 February 2018; pp. 139–144.
146. Dai, X.; Mao, Y.; Huang, T.; Qin, N.; Huang, D.; Li, Y. Automatic obstacle avoidance of quadrotor UAV via CNN-based learning. *Neurocomputing* **2020**, *402*, 346–358. [[CrossRef](#)]
147. Back, S.; Cho, G.; Oh, J.; Tran, X.T.; Oh, H. Autonomous UAV trail navigation with obstacle avoidance using deep neural networks. *J. Intell. Robot. Syst.* **2020**, *100*, 1195–1211. [[CrossRef](#)]
148. Lee, H.; Ho, H.; Zhou, Y. Deep Learning-based Monocular Obstacle Avoidance for Unmanned Aerial Vehicle Navigation in Tree Plantations. *J. Intell. Robot. Syst.* **2021**, *101*, 1–18. [[CrossRef](#)]
149. Yang, X.; Chen, J.; Dang, Y.; Luo, H.; Tang, Y.; Liao, C.; Chen, P.; Cheng, K.T. Fast Depth Prediction and Obstacle Avoidance on a Monocular Drone Using Probabilistic Convolutional Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2019**, *22*, 156–167. [[CrossRef](#)]
150. Wang, D.; Li, W.; Liu, X.; Li, N.; Zhang, C. UAV environmental perception and autonomous obstacle avoidance: A deep learning and depth camera combined solution. *Comput. Electron. Agric.* **2020**, *175*, 105523. [[CrossRef](#)]
151. Sanket, N.J.; Parameshwara, C.M.; Singh, C.D.; Kuruttukulam, A.V.; Fermüller, C.; Scaramuzza, D.; Aloimonos, Y. EVDodgeNet: Deep dynamic obstacle dodging with event cameras. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Virtual, 31 May–15 June 2020; pp. 10651–10657.
152. Hamel, T.; Mahony, R.; Lozano, R.; Ostrowski, J. Dynamic modelling and configuration stabilization for an X4-flyer. *IFAC Proc. Vol.* **2002**, *35*, 217–222. [[CrossRef](#)]
153. Faessler, M.; Franchi, A.; Scaramuzza, D. Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories. *IEEE Robot. Autom. Lett.* **2017**, *3*, 620–626. [[CrossRef](#)]
154. Lesprier, J.; Biannic, J.M.; Roos, C. Modeling and robust nonlinear control of a fixed-wing UAV. In Proceedings of the 2015 IEEE Conference on Control Applications (CCA), Sydney, NSW, Australia, 21–23 September 2015; pp. 1334–1339.
155. Foehn, P.; Falanga, D.; Kuppuswamy, N.; Tedrake, R.; Scaramuzza, D. *Fast Trajectory Optimization for Agile Quadrotor Maneuvers with a Cable-Suspended Payload*; Robotics: Science and Systems Foundation: Cambridge, MA, USA, 2017.
156. Tang, S.; Wüest, V.; Kumar, V. Aggressive flight with suspended payloads using vision-based control. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1152–1159. [[CrossRef](#)]
157. Bangura, M.; Mahony, R. Real-time model predictive control for quadrotors. *IFAC Proc. Vol.* **2014**, *47*, 11773–11780. [[CrossRef](#)]
158. Yamasaki, T.; Takano, H.; Bab, Y. *Robust Path-Following for UAV Using Pure Pursuit Guidance*; IntechOpen: London, UK, 2009; pp. 671–690. [[CrossRef](#)]
159. Zheng, E.H.; Xiong, J.J.; Luo, J.L. Second order sliding mode control for a quadrotor UAV. *ISA Trans.* **2014**, *53*, 1350–1356. [[CrossRef](#)] [[PubMed](#)]
160. Ambrosino, G.; Ariola, M.; Ciniglio, U.; Corrado, F.; De Lellis, E.; Pironti, A. Path generation and tracking in 3-D for UAVs. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 980–988. [[CrossRef](#)]
161. Yang, K.; Kang, Y.; Sukkarieh, S. Adaptive nonlinear model predictive path-following control for a fixed-wing unmanned aerial vehicle. *Int. J. Control. Autom. Syst.* **2013**, *11*, 65–74. [[CrossRef](#)]
162. Bicego, D.; Mazzetto, J.; Carli, R.; Farina, M.; Franchi, A. Nonlinear model predictive control with enhanced actuator model for multi-rotor aerial vehicles with generic designs. *J. Intell. Robot. Syst.* **2020**, *100*, 1213–1247. [[CrossRef](#)]
163. Li, B.; Zhou, W.; Sun, J.; Wen, C.Y.; Chen, C.K. Development of model predictive controller for a Tail-Sitter VTOL UAV in hover flight. *Sensors* **2018**, *18*, 2859. [[CrossRef](#)]
164. Kamel, M.; Burri, M.; Siegwart, R. Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles. *IFAC-PapersOnLine* **2017**, *50*, 3463–3469. [[CrossRef](#)]
165. Mellinger, D.; Michael, N.; Kumar, V. Trajectory generation and control for precise aggressive maneuvers with quadrotors. *Int. J. Robot. Res.* **2012**, *31*, 664–674. [[CrossRef](#)]

166. Bry, A.; Richter, C.; Bachrach, A.; Roy, N. Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments. *Int. J. Robot. Res.* **2015**, *34*, 969–1002. [[CrossRef](#)]
167. Loianno, G.; Brunner, C.; McGrath, G.; Kumar, V. Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU. *IEEE Robot. Autom. Lett.* **2016**, *2*, 404–411. [[CrossRef](#)]
168. Michael, N.; Fink, J.; Kumar, V. Cooperative manipulation and transportation with aerial robots. *Auton. Robot.* **2011**, *30*, 73–86. [[CrossRef](#)]
169. Fink, J.; Michael, N.; Kim, S.; Kumar, V. Planning and control for cooperative manipulation and transportation with aerial robots. *Int. J. Robot. Res.* **2011**, *30*, 324–334. [[CrossRef](#)]
170. Sreenath, K.; Kumar, V. Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots. *rm* **2013**, *1*, r3.
171. Lee, T.; Leok, M.; McClamroch, N.H. Nonlinear robust tracking control of a quadrotor UAV on SE (3). *Asian J. Control* **2013**, *15*, 391–408. [[CrossRef](#)]
172. Omari, S.; Hua, M.D.; Ducard, G.; Hamel, T. Nonlinear control of VTOL UAVs incorporating flapping dynamics. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 2419–2425.
173. Manjunath, A.; Mehrook, P.; Sharma, R.; Ratnoo, A. Application of virtual target based guidance laws to path following of a quadrotor UAV. In Proceedings of the 2016 International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 7–10 June 2016; pp. 252–260.
174. Lee, H.; Kim, H.J. Trajectory tracking control of multirotors from modelling to experiments: A survey. *Int. J. Control Autom. Syst.* **2017**, *15*, 281–292. [[CrossRef](#)]
175. Nascimento, T.P.; Saska, M. Position and attitude control of multi-rotor aerial vehicles: A survey. *Annu. Rev. Control* **2019**, *48*, 129–146. [[CrossRef](#)]
176. Sujit, P.B.; Saripalli, S.; Sousa, J.B. An evaluation of UAV path following algorithms. In Proceedings of the 2013 European Control Conference (ECC), Zurich, Switzerland, 17–19 July 2013; pp. 3332–3337. [[CrossRef](#)]
177. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [[CrossRef](#)]
178. Taketomi, T.; Uchiyama, H.; Ikeda, S. Visual SLAM algorithms: A survey from 2010 to 2016. *IPSN Trans. Comput. Vis. Appl.* **2017**, *9*, 1–11. [[CrossRef](#)]
179. 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](#)]
180. Zhang, J.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-time. In Proceedings of the Robotics: Science and Systems, Berkeley, CA, USA, 13–15 July 2015; Volume 2.
181. Hess, W.; Kohler, D.; Rapp, H.; Andor, D. Real-time loop closure in 2D LIDAR SLAM. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1271–1278.
182. Koide, K.; Miura, J.; Menegatti, E. A portable 3D lidar-based system for long-term and wide-area people behavior measurement. *IEEE Trans. Hum. Mach. Syst.* **2018**, *16*, 1729881419841532.
183. Shan, T.; Englot, B. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765.
184. Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Daniela, R. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, Nevada, USA, 25–29 October 2020; pp. 5135–5142.
185. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234.
186. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [[CrossRef](#)]
187. Mur-Artal, R.; Tardós, J.D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [[CrossRef](#)]
188. Engel, J.; Schöps, T.; Cremers, D. LSD-SLAM: Large-scale direct monocular SLAM. In *The 13th European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 834–849.
189. Engel, J.; Koltun, V.; Cremers, D. Direct Sparse Odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 611–625. [[CrossRef](#)] [[PubMed](#)]
190. Whelan, T.; Kaess, M.; Johannsson, H.; Fallon, M.; Leonard, J.J.; McDonald, J. Real-time large-scale dense RGB-D SLAM with volumetric fusion. *Int. J. Robot. Res.* **2015**, *34*, 598–626. [[CrossRef](#)]
191. Naudet-Collette, S.; Melbouci, K.; Gay-Bellile, V.; Ait-Aider, O.; Dhome, M. Constrained RGBD-SLAM. *Robotica* **2021**, *39*, 277–290. [[CrossRef](#)]
192. Holz, D.; Nieuwenhuisen, M.; Droschel, D.; Schreiber, M.; Behnke, S. Towards Multimodal Omnidirectional Obstacle Detection for Autonomous Unmanned Aerial Vehicles. *ISPRS-Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2013**, *XL-1/W2*, 201–206. [[CrossRef](#)]
193. Hoy, M.; Matveev, A.S.; Garratt, M.; Savkin, A.V. Collision-free navigation of an autonomous unmanned helicopter in unknown urban environments: Sliding mode and MPC approaches. *Robotica* **2012**, *30*, 537–550. [[CrossRef](#)]

194. Yang, K.; Keat Gan, S.; Sukkarieh, S. A Gaussian process-based RRT planner for the exploration of an unknown and cluttered environment with a UAV. *Adv. Robot.* **2013**, *27*, 431–443. [[CrossRef](#)]
195. Oleynikova, H.; Taylor, Z.; Siegwart, R.; Nieto, J. Safe Local Exploration for Replanning in Cluttered Unknown Environments for Microaerial Vehicles. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1474–1481. [[CrossRef](#)]
196. Meera, A.A.; Popović, M.; Millane, A.; Siegwart, R. Obstacle-aware adaptive informative path planning for UAV-based target search. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 718–724.
197. Blösch, M.; Weiss, S.; Scaramuzza, D.; Siegwart, R. Vision based MAV navigation in unknown and unstructured environments. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA, 3–7 May 2010; pp. 21–28. [[CrossRef](#)]
198. Shen, S.; Michael, N.; Kumar, V. Autonomous multi-floor indoor navigation with a computationally constrained MAV. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 20–25.
199. Perez-Grau, F.J.; Ragel, R.; Caballero, F.; Viguria, A.; Ollero, A. An architecture for robust UAV navigation in GPS-denied areas. *J. Field Robot.* **2018**, *35*, 121–145. [[CrossRef](#)]
200. Bachrach, A.; Winter, A.D.; He, R.; Hemann, G.; Prentice, S.; Roy, N. RANGE—Robust autonomous navigation in GPS-denied environments. *J. Field Robot.* **2011**, *28*, 644–666. [[CrossRef](#)]
201. Oleynikova, H.; Lanegger, C.; Taylor, Z.; Pantic, M.; Millane, A.; Siegwart, R.; Nieto, J. An open-source system for vision-based micro-aerial vehicle mapping, planning, and flight in cluttered environments. *J. Field Robot.* **2020**, *37*, 642–666. [[CrossRef](#)]
202. Palossi, D.; Conti, F.; Benini, L. An open source and open hardware deep learning-powered visual navigation engine for autonomous nano-UAVs. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 29–31 May 2019; pp. 604–611.
203. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast Semi-Direct Monocular Visual Odometry. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014.
204. Labbé, M.; Michaud, F. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *J. Field Robot.* **2019**, *36*, 416–446. [[CrossRef](#)]
205. Labbé, M.; Michaud, F. Long-term online multi-session graph-based SPLAM with memory management. *Auton. Robot.* **2018**, *42*, 1133–1150. [[CrossRef](#)]
206. Whelan, T.; Salas-Moreno, R.F.; Glocker, B.; Davison, A.J.; Leutenegger, S. ElasticFusion: Real-time dense SLAM and light source estimation. *Int. J. Robot. Res.* **2016**, *35*, 1697–1716. [[CrossRef](#)]
207. Zhou, B.; Zhang, Y.; Chen, X.; Shen, S. FUEL: Fast UAV Exploration Using Incremental Frontier Structure and Hierarchical Planning. *IEEE Robot. Autom. Lett.* **2021**, *6*, 779–786. [[CrossRef](#)]
208. Zhou, B.; Gao, F.; Pan, J.; Shen, S. Robust real-time UAV replanning using guided gradient-based optimization and topological paths. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 1208–1214.
209. Pham, H.; Pham, Q. A new approach to time-optimal path parameterization based on reachability analysis. *IEEE Trans. Robot.* **2018**, *34*, 645–659. [[CrossRef](#)]
210. Sundermeyer, M.; Marton, Z.C.; Durner, M.; Brucker, M.; Triebel, R. Implicit 3D Orientation Learning for 6D Object Detection from RGB Images. In Proceedings of the The European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
211. Wada, K.; Sucar, E.; James, S.; Lenton, D.; Davison, A.J. Morefusion: Multi-object reasoning for 6D pose estimation from volumetric fusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 14540–14549.
212. Zampogiannis, K.; Fermuller, C.; Aloimonos, Y. cilantro: A Lean, Versatile, and Efficient Library for Point Cloud Data Processing. In Proceedings of the 26th ACM International Conference on Multimedia, MM’18, Seoul, Republic of Korea, 22–26 October 2018; ACM: New York, NY, USA, 2018; pp. 1364–1367. [[CrossRef](#)]
213. Furrer, F.; Burri, M.; Achtelik, M.; Siegwart, R. RotorS—A Modular Gazebo MAV Simulator Framework. In *Robot Operating System (ROS): The Complete Reference*; Springer International Publishing: Cham, Switzerland, 2016; Volume 1, pp. 595–625. [[CrossRef](#)]
214. Wächter, A.; Biegler, L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* **2006**, *106*, 25–57. [[CrossRef](#)]
215. Winkler, A.W. Ifopt—A modern, light-weight, Eigen-based C++ interface to Nonlinear Programming solvers Ipopt and Snopt. Available online: <https://zenodo.org/record/1135085> (accessed on 15 September 2021).
216. Seelan, S.K.; Laguet, S.; Casady, G.M.; Seielstad, G.A. Remote sensing applications for precision agriculture: A learning community approach. *Remote Sens. Environ.* **2003**, *88*, 157–169. [[CrossRef](#)]
217. Xiang, H.; Tian, L. Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV). *Biosyst. Eng.* **2011**, *108*, 174–190. [[CrossRef](#)]
218. Alsalam, B.H.Y.; Morton, K.; Campbell, D.; Gonzalez, F. Autonomous UAV with vision based on-board decision making for remote sensing and precision agriculture. In Proceedings of the 2017 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2017; pp. 1–12.

219. Albani, D.; Nardi, D.; Trianni, V. Field coverage and weed mapping by UAV swarms. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 4319–4325.
220. Albani, D.; IJsselmuiden, J.; Haken, R.; Trianni, V. Monitoring and mapping with robot swarms for agricultural applications. In Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 29 August–1 September 2017; pp. 1–6.
221. Gonzalez-de Santos, P.; Ribeiro, A.; Fernandez-Quintanilla, C.; Lopez-Granados, F.; Brandstoetter, M.; Tomic, S.; Pedrazzi, S.; Peruzzi, A.; Pajares, G.; Kaplanis, G.; et al. Fleets of robots for environmentally-safe pest control in agriculture. *Precis. Agric.* **2017**, *18*, 574–614. [[CrossRef](#)]
222. Xue, X.; Lan, Y.; Sun, Z.; Chang, C.; Hoffmann, W.C. Develop an unmanned aerial vehicle based automatic aerial spraying system. *Comput. Electron. Agric.* **2016**, *128*, 58–66. [[CrossRef](#)]
223. Slaughter, D.C.; Giles, D.K.; Downey, D. Autonomous robotic weed control systems: A review. *Comput. Electron. Agric.* **2008**, *61*, 63–78. [[CrossRef](#)]
224. Lottes, P.; Khanna, R.; Pfeifer, J.; Siegwart, R.; Stachniss, C. UAV-based crop and weed classification for smart farming. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3024–3031.
225. de Castro, A.; Torres-Sánchez, J.; Peña, J.; Jiménez-Brenes, F.; Csillik, O.; López-Granados, F. An automatic random forest-OBIA algorithm for early weed mapping between and within crop rows using UAV imagery. *Remote Sens.* **2018**, *10*, 285. [[CrossRef](#)]
226. Van Henten, E.J.; Hemming, J.; Van Tuijl, B.A.J.; Kornet, J.G.; Meuleman, J.; Bontsema, J.; Van Os, E.A. An autonomous robot for harvesting cucumbers in greenhouses. *Auton. Robot.* **2002**, *13*, 241–258. [[CrossRef](#)]
227. Zhang, C.; Kovacs, J.M. The application of small unmanned aerial systems for precision agriculture: A review. *Precis. Agric.* **2012**, *13*, 693–712. [[CrossRef](#)]
228. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Lagkas, T.; Moscholios, I. A compilation of UAV applications for precision agriculture. *Comput. Netw.* **2020**, *172*, 107148. [[CrossRef](#)]
229. De Cubber, G.; Doroftei, D.; Serrano, D.; Chintamani, K.; Sabino, R.; Ourevitch, S. The EU-ICARUS project: Developing assistive robotic tools for search and rescue operations. In Proceedings of the 2013 IEEE international symposium on safety, security, and rescue robotics (SSRR), Linköping, Sweden, 21–26 October 2013; pp. 1–4.
230. Scherer, J.; Yahyanejad, S.; Hayat, S.; Yanmaz, E.; Andre, T.; Khan, A.; Vukadinovic, V.; Bettstetter, C.; Hellwagner, H.; Rinner, B. An autonomous multi-UAV system for search and rescue. In Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use, Florence, Italy, 18 May 2015; pp. 33–38.
231. Qi, J.; Song, D.; Shang, H.; Wang, N.; Hua, C.; Wu, C.; Qi, X.; Han, J. Search and rescue rotary-wing UAV and its application to the lushan ms 7.0 earthquake. *J. Field Robot.* **2016**, *33*, 290–321. [[CrossRef](#)]
232. Silvagni, M.; Tonoli, A.; Zenerino, E.; Chiaberge, M. Multipurpose UAV for search and rescue operations in mountain avalanche events. *Geomat. Nat. Hazards Risk* **2017**, *8*, 18–33. [[CrossRef](#)]
233. Tomic, T.; Schmid, K.; Lutz, P.; Domel, A.; Kassecker, M.; Mair, E.; Grixia, I.L.; Ruess, F.; Suppa, M.; Burschka, D. Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue. *IEEE Robot. Autom. Mag.* **2012**, *19*, 46–56. [[CrossRef](#)]
234. Verykokou, S.; Ioannidis, C.; Athanasiou, G.; Doulamis, N.; Amditis, A. 3D reconstruction of disaster scenes for urban search and rescue. *Multimed. Tools Appl.* **2018**, *77*, 9691–9717. [[CrossRef](#)]
235. Mittal, M.; Mohan, R.; Burgard, W.; Valada, A. Vision-based autonomous UAV navigation and landing for urban search and rescue. *arXiv* **2019**, arXiv:1906.01304.
236. Dang, T.; Mascarih, F.; Khattak, S.; Nguyen, H.; Nguyen, H.; Hirsh, S.; Reinhart, R.; Papachristos, C.; Alexis, K. Autonomous search for underground mine rescue using aerial robots. In Proceedings of the 2020 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2020; pp. 1–8.
237. Zimroz, P.; Trybała, P.; Wróblewski, A.; Góralczyk, M.; Szrek, J.; Wójcik, A.; Zimroz, R. Application of UAV in Search and Rescue Actions in Underground Mine—A Specific Sound Detection in Noisy Acoustic Signal. *Energies* **2021**, *14*, 3725. [[CrossRef](#)]
238. Yeong, S.; King, L.; Dol, S. A review on marine search and rescue operations using unmanned aerial vehicles. *Int. J. Mar. Environ. Sci.* **2015**, *9*, 396–399.
239. Xiao, X.; Dufek, J.; Woodbury, T.; Murphy, R. UAV assisted USV visual navigation for marine mass casualty incident response. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 6105–6110.
240. Lygouras, E.; Santavas, N.; Taitzoglou, A.; Tarchanidis, K.; Mitropoulos, A.; Gasteratos, A. Unsupervised human detection with an embedded vision system on a fully autonomous UAV for search and rescue operations. *Sensors* **2019**, *19*, 3542. [[CrossRef](#)] [[PubMed](#)]
241. Qingqing, L.; Taipalmaa, J.; Queraltá, J.P.; Gia, T.N.; Gabbouj, M.; Tenhunen, H.; Raitoharju, J.; Westerlund, T. Towards active vision with UAVs in marine search and rescue: Analyzing human detection at variable altitudes. In Proceedings of the 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Abu Dhabi, UAE, 4–6 November 2020; pp. 65–70.
242. Chamoso, P.; Raveane, W.; Parra, V.; González, A. UAVs applied to the counting and monitoring of animals. In *Ambient Intelligence-Software and Applications*; Springer: Cham, Switzerland, 2014; pp. 71–80.

243. Xu, B.; Wang, W.; Falzon, G.; Kwan, P.; Guo, L.; Sun, Z.; Li, C. Livestock classification and counting in quadcopter aerial images using Mask R-CNN. *Int. J. Remote Sens.* **2020**, *41*, 8121–8142. [[CrossRef](#)]
244. Gonzalez, L.F.; Montes, G.A.; Puig, E.; Johnson, S.; Mengersen, K.; Gaston, K.J. Unmanned aerial vehicles (UAVs) and artificial intelligence revolutionizing wildlife monitoring and conservation. *Sensors* **2016**, *16*, 97. [[CrossRef](#)]
245. Hodgson, J.C.; Baylis, S.M.; Mott, R.; Herrod, A.; Clarke, R.H. Precision wildlife monitoring using unmanned aerial vehicles. *Sci. Rep.* **2016**, *6*, 331. [[CrossRef](#)]
246. Su, X.; Dong, S.; Liu, S.; Cracknell, A.P.; Zhang, Y.; Wang, X.; Liu, G. Using an unmanned aerial vehicle (UAV) to study wild yak in the highest desert in the world. *Int. J. Remote Sens.* **2018**, *39*, 5490–5503. [[CrossRef](#)]
247. Fortuna, J.; Ferreira, F.; Gomes, R.; Ferreira, S.; Sousa, J. Using low cost open source UAVs for marine wild life monitoring-Field Report. *IFAC Proc. Vol.* **2013**, *46*, 291–295. [[CrossRef](#)]
248. Paranjape, A.A.; Chung, S.J.; Kim, K.; Shim, D.H. Robotic herding of a flock of birds using an unmanned aerial vehicle. *IEEE Trans. Robot.* **2018**, *34*, 901–915. [[CrossRef](#)]
249. Yaxley, K.J.; McIntyre, N.; Park, J.; Healey, J. Sky shepherds: A tale of a UAV and sheep. In *Shepherding UxVs for Human-Swarm Teaming: An Artificial Intelligence Approach to Unmanned X Vehicles*; Springer: Cham, Switzerland, 2021; pp. 189–206.
250. Scobie, C.A.; Hugenholtz, C.H. Wildlife monitoring with unmanned aerial vehicles: Quantifying distance to auditory detection. *Wildl. Soc. Bull.* **2016**, *40*, 781–785. [[CrossRef](#)]
251. Hodgson, J.C.; Koh, L.P. Best practice for minimising unmanned aerial vehicle disturbance to wildlife in biological field research. *Curr. Biol.* **2016**, *26*, R404–R405. [[CrossRef](#)] [[PubMed](#)]
252. Eheim, C.; Dixon, C.; Argrow, B.; Palo, S. Tornadochaser: A remotely-piloted UAV for in situ meteorological measurements. In Proceedings of the 1st UAV Conference, Portsmouth, VA, USA, 20–23 May 2002; p. 3479.
253. DeBusk, W. Unmanned aerial vehicle systems for disaster relief: Tornado alley. In Proceedings of the AIAA Infotech@ Aerospace 2010, Atlanta, GA, USA, 20–22 April 2010; p. 3506.
254. Witte, B.M.; Schlagenhaut, C.; Mullen, J.; Helvey, J.P.; Thamann, M.A.; Bailey, S. Fundamental Turbulence Measurement with Unmanned Aerial Vehicles. In Proceedings of the 8th AIAA Atmospheric and Space Environments Conference, Washington, DC, USA, 13–17 June 2016; p. 3584.
255. Avery, A.S.; Hathaway, J.D.; Jacob, J.D. Sensor Suite Development for a Weather UAV. In Proceedings of the 7th AIAA Atmospheric and Space Environments Conference, Dallas, TX, USA, 22–26 June 2015; p. 3022.
256. Zhou, S.; Gheisari, M. Unmanned aerial system applications in construction: A systematic review. In *Construction Innovation*; Emerald Publishing Limited: Bingley, UK, 2018.
257. Ham, Y.; Han, K.K.; Lin, J.J.; Golparvar-Fard, M. Visual monitoring of civil infrastructure systems via camera-equipped Unmanned Aerial Vehicles (UAVs): A review of related works. *Vis. Eng.* **2016**, *4*, 1–8. [[CrossRef](#)]
258. De Melo, R.R.S.; Costa, D.B.; Álvares, J.S.; Irizarry, J. Applicability of unmanned aerial system (UAS) for safety inspection on construction sites. *Saf. Sci.* **2017**, *98*, 174–185. [[CrossRef](#)]
259. Li, Y.; Liu, C. Applications of multirotor drone technologies in construction management. *Int. J. Constr. Manag.* **2019**, *19*, 401–412. [[CrossRef](#)]
260. Jacob-Loyola, N.; Rivera, M.L.; Herrera, R.F.; Atencio, E. Unmanned Aerial Vehicles (UAVs) for Physical Progress Monitoring of Construction. *Sensors* **2021**, *21*, 4227. [[CrossRef](#)] [[PubMed](#)]
261. Nikulin, A.; de Smet, T.S. A UAV-based magnetic survey method to detect and identify orphaned oil and gas wells. *Lead. Edge* **2019**, *38*, 447–452. [[CrossRef](#)]
262. Gómez, C.; Green, D.R. Small unmanned airborne systems to support oil and gas pipeline monitoring and mapping. *Arab. J. Geosci.* **2017**, *10*, 202. [[CrossRef](#)]
263. Shukla, A.; Xiaoqian, H.; Karki, H. Autonomous tracking and navigation controller for an unmanned aerial vehicle based on visual data for inspection of oil and gas pipelines. In Proceedings of the 2016 16th International Conference on Control, Automation and Systems (ICCAS), Gyeongju, South Korea, 16–19 October 2016; pp. 194–200.
264. Yan, Y.; Ma, S.; Yin, S.; Hu, S.; Long, Y.; Xie, C.; Jiang, H. Detection and numerical simulation of potential hazard in oil pipeline areas based on UAV surveys. *Front. Earth Sci.* **2021**, *9*, 272. [[CrossRef](#)]
265. Bretschneider, T.R.; Shetti, K. UAV-based gas pipeline leak detection. In Proceedings of the ARCS, Porto, Portugal, 24–27 March 2015.
266. Quater, P.B.; Grimaccia, F.; Leva, S.; Mussetta, M.; Aghaei, M. Light Unmanned Aerial Vehicles (UAVs) for cooperative inspection of PV plants. *IEEE J. Photovoltaics* **2014**, *4*, 1107–1113. [[CrossRef](#)]
267. Vega Díaz, J.J.; Vlamincq, M.; Lefkaditis, D.; Orjuela Vargas, S.A.; Luong, H. Solar panel detection within complex backgrounds using thermal images acquired by UAVs. *Sensors* **2020**, *20*, 6219. [[CrossRef](#)] [[PubMed](#)]
268. Zhang, Y.; Yuan, X.; Li, W.; Chen, S. Automatic power line inspection using UAV images. *Remote Sens.* **2017**, *9*, 824. [[CrossRef](#)]
269. Jenssen, R.; Roverso, D. Automatic autonomous vision-based power line inspection: A review of current status and the potential role of deep learning. *Int. J. Electr. Power Energy Syst.* **2018**, *99*, 107–120.
270. Cheng, L.; Tan, X.; Yao, D.; Xu, W.; Wu, H.; Chen, Y. A Fishery Water Quality Monitoring and Prediction Evaluation System for Floating UAV Based on Time Series. *Sensors* **2021**, *21*, 4451. [[CrossRef](#)] [[PubMed](#)]
271. Lipovský, P.; Draganová, K.; Novotňák, J.; Szóke, Z.; Fil'ko, M. Indoor Mapping of Magnetic Fields Using UAV Equipped with Fluxgate Magnetometer. *Sensors* **2021**, *21*, 4191. [[CrossRef](#)] [[PubMed](#)]

272. Klausen, K.; Fossen, T.I.; Johansen, T.A. Nonlinear control with swing damping of a multirotor UAV with suspended load. *J. Intell. Robot. Syst.* **2017**, *88*, 379–394. [[CrossRef](#)]
273. Villa, D.K.; Brandao, A.S.; Sarcinelli-Filho, M. A survey on load transportation using multirotor UAVs. *J. Intell. Robot. Syst.* **2019**, *98*, 267–296. [[CrossRef](#)]
274. González-deSantos, L.M.; Martínez-Sánchez, J.; González-Jorge, H.; Ribeiro, M.; de Sousa, J.B.; Arias, P. Payload for contact inspection tasks with UAV systems. *Sensors* **2019**, *19*, 3752. [[CrossRef](#)]
275. Kanistras, K.; Martins, G.; Rutherford, M.J.; Valavanis, K.P. A survey of unmanned aerial vehicles (UAVs) for traffic monitoring. In Proceedings of the 2013 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 28–31 May 2013; pp. 221–234.
276. Silva, L.A.; Sanchez San Blas, H.; Peral García, D.; Sales Mendes, A.; Villarubia González, G. An architectural multi-agent system for a pavement monitoring system with pothole recognition in UAV images. *Sensors* **2020**, *20*, 6205. [[CrossRef](#)]
277. Outay, F.; Mengash, H.A.; Adnan, M. Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges. *Transp. Res. Part A Policy Pract.* **2020**, *141*, 116–129. [[CrossRef](#)]
278. Mademlis, I.; Mygdalis, V.; Nikolaidis, N.; Pitas, I. Challenges in autonomous UAV cinematography: An overview. In Proceedings of the 2018 IEEE international conference on multimedia and expo (ICME), San Diego, CA, USA, 23–27 July 2018; pp. 1–6.
279. Augugliaro, F.; Schoellig, A.P.; D’Andrea, R. Dance of the flying machines: Methods for designing and executing an aerial dance choreography. *IEEE Robot. Autom. Mag.* **2013**, *20*, 96–104. [[CrossRef](#)]
280. Schoellig, A.P.; Siegel, H.; Augugliaro, F.; D’Andrea, R. So you think you can dance? Rhythmic flight performances with quadcopters. In *Controls and Art*; Springer International Publishing: Basel, Switzerland, 2014; pp. 73–105.
281. Kim, S.J.; Jeong, Y.; Park, S.; Ryu, K.; Oh, G. A survey of drone use for entertainment and AVR (augmented and virtual reality). In *Augmented Reality and Virtual Reality*; Springer: Cham, Switzerland, 2018; pp. 339–352.
282. Ang, K.Z.; Dong, X.; Liu, W.; Qin, G.; Lai, S.; Wang, K.; Wei, D.; Zhang, S.; Phang, S.K.; Chen, X.; et al. High-precision multi-UAV teaming for the first outdoor night show in Singapore. *Unmanned Syst.* **2018**, *6*, 39–65. [[CrossRef](#)]
283. Imdoukh, A.; Shaker, A.; Al-Toukhy, A.; Kablaoui, D.; El-Abd, M. Semi-autonomous indoor firefighting UAV. In Proceedings of the 2017 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 10–12 July 2017; pp. 310–315.
284. Qin, H.; Cui, J.Q.; Li, J.; Bi, Y.; Lan, M.; Shan, M.; Liu, W.; Wang, K.; Lin, F.; Zhang, Y.; et al. Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions. In Proceedings of the 2016 12th IEEE International Conference on Control and Automation (ICCA), Kathmandu, Nepal, 1–3 June 2016; pp. 62–67.
285. Sudhakar, S.; Vijayakumar, V.; Kumar, C.S.; Priya, V.; Ravi, L.; Subramaniaswamy, V. Unmanned Aerial Vehicle (UAV) based Forest Fire Detection and monitoring for reducing false alarms in forest-fires. *Comput. Commun.* **2020**, *149*, 1–16. [[CrossRef](#)]
286. Chung, S.J.; Paranjape, A.A.; Dames, P.; Shen, S.; Kumar, V. A survey on aerial swarm robotics. *IEEE Trans. Robot.* **2018**, *34*, 837–855. [[CrossRef](#)]
287. Oh, K.K.; Park, M.C.; Ahn, H.S. A survey of multi-agent formation control. *Automatica* **2015**, *53*, 424–440. [[CrossRef](#)]
288. Saif, O.; Fantoni, I.; Zavala-Río, A. Distributed integral control of multiple UAVs: Precise flocking and navigation. *IET Control Theory Appl.* **2019**, *13*, 2008–2017. [[CrossRef](#)]
289. Mercado, D.; Castro, R.; Lozano, R. Quadrotors flight formation control using a leader-follower approach. In Proceedings of the 2013 European Control Conference (ECC), Zurich, Switzerland, 17–19 July 2013; pp. 3858–3863.
290. Hou, Z.; Fantoni, I. Distributed leader-follower formation control for multiple quadrotors with weighted topology. In Proceedings of the 2015 10th System of Systems Engineering Conference (SoSE), San Antonio, TX, USA, 17–20 May 2015; pp. 256–261.
291. Dehghani, M.A.; Menhaj, M.B. Communication free leader-follower formation control of unmanned aircraft systems. *Robot. Autom. Syst.* **2016**, *80*, 69–75. [[CrossRef](#)]
292. Xuan-Mung, N.; Hong, S.K. Robust adaptive formation control of quadcopters based on a leader-follower approach. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419862733. [[CrossRef](#)]
293. Walter, V.; Staub, N.; Franchi, A.; Saska, M. UVDAR System for Visual Relative Localization with Application to Leader-Follower Formations of Multirotor UAVs. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2637–2644. [[CrossRef](#)]
294. Wang, X.; Shen, L.; Liu, Z.; Zhao, S.; Cong, Y.; Li, Z.; Jia, S.; Chen, H.; Yu, Y.; Chang, Y.; et al. Coordinated flight control of miniature fixed-wing UAV swarms: methods and experiments. *Sci. China Inf. Sci.* **2019**, *62*, 212204. [[CrossRef](#)]
295. Tagliabue, A.; Kamel, M.; Siegwart, R.; Nieto, J. Robust collaborative object transportation using multiple MAVs. *Int. J. Robot. Res.* **2019**, *38*, 1020–1044. [[CrossRef](#)]
296. Ren, W.; Beard, R.W. Decentralized scheme for spacecraft formation flying via the virtual structure approach. *J. Guid. Control. Dyn.* **2004**, *27*, 73–82. [[CrossRef](#)]
297. Li, N.H.; Liu, H.H. Formation UAV flight control using virtual structure and motion synchronization. In Proceedings of the 2008 American Control Conference, Seattle, WA, USA, 11–13 June 2008; pp. 1782–1787.
298. Yoshioka, C.; Namerikawa, T. Formation control of nonholonomic multi-vehicle systems based on virtual structure. *IFAC Proc. Vol.* **2008**, *41*, 5149–5154. [[CrossRef](#)]
299. Bayezit, İ.; Fidan, B. Distributed cohesive motion control of flight vehicle formations. *IEEE Trans. Ind. Electron.* **2012**, *60*, 5763–5772. [[CrossRef](#)]
300. Kushleyev, A.; Mellinger, D.; Powers, C.; Kumar, V. Towards a swarm of agile micro quadrotors. *Auton. Robot.* **2013**, *35*, 287–300. [[CrossRef](#)]

301. Zhihao, C.; Longhong, W.; Jiang, Z.; Kun, W.; Yingxun, W. Virtual target guidance-based distributed model predictive control for formation control of multiple UAVs. *Chin. J. Aeronaut.* **2020**, *33*, 1037–1056.
302. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH Computer Graphics*; ACM: New York, NY, USA, 1987; Volume 21; pp. 25–34.
303. Olfati-Saber, R. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Trans. Autom. Control* **2006**, *51*, 401–420. [[CrossRef](#)]
304. Tanner, H.G.; Jadbabaie, A.; Pappas, G.J. Flocking in fixed and switching networks. *IEEE Trans. Autom. Control* **2007**, *52*, 863–868. [[CrossRef](#)]
305. Savkin, A.V.; Teimoori, H. Decentralized navigation of groups of wheeled mobile robots with limited communication. *IEEE Trans. Robot.* **2010**, *26*, 1099–1104. [[CrossRef](#)]
306. Reyes, L.A.V.; Tanner, H.G. Flocking, formation control, and path following for a group of mobile robots. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 1268–1282. [[CrossRef](#)]
307. Khaledyan, M.; Liu, T.; Fernandez-Kim, V.; de Queiroz, M. Flocking and target interception control for formations of nonholonomic kinematic agents. *IEEE Trans. Control Syst. Technol.* **2019**, *28*, 1603–1610. [[CrossRef](#)]
308. Do, K.D. Flocking for multiple elliptical agents with limited communication ranges. *IEEE Trans. Robot.* **2011**, *27*, 931–942. [[CrossRef](#)]
309. Antonelli, G.; Arrichiello, F.; Chiaverini, S. Flocking for multi-robot systems via the null-space-based behavioral control. *Swarm Intell.* **2010**, *4*, 37. [[CrossRef](#)]
310. Ghapani, S.; Mei, J.; Ren, W.; Song, Y. Fully distributed flocking with a moving leader for Lagrange networks with parametric uncertainties. *Automatica* **2016**, *67*, 67–76. [[CrossRef](#)]
311. Jafari, M.; Xu, H. A biologically-inspired distributed fault tolerant flocking control for multi-agent system in presence of uncertain dynamics and unknown disturbance. *Eng. Appl. Artif. Intell.* **2019**, *79*, 1–12. [[CrossRef](#)]
312. Ren, W.; Atkins, E. Distributed multi-vehicle coordinated control via local information exchange. *Int. J. Robust Nonlinear Control. IFAC-Affil. J.* **2007**, *17*, 1002–1033. [[CrossRef](#)]
313. Saulnier, K.; Saldana, D.; Prorok, A.; Pappas, G.J.; Kumar, V. Resilient flocking for mobile robot teams. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1039–1046. [[CrossRef](#)]
314. Cao, Y.; Stuart, D.; Ren, W.; Meng, Z. Distributed containment control for multiple autonomous vehicles with double-integrator dynamics: Algorithms and experiments. *IEEE Trans. Control Syst. Technol.* **2011**, *19*, 929–938. [[CrossRef](#)]
315. Dimarogonas, D.V.; Kyriakopoulos, K.J. A connection between formation infeasibility and velocity alignment in kinematic multi-agent systems. *Automatica* **2008**, *44*, 2648–2654. [[CrossRef](#)]
316. Yang, D.; Ren, W.; Liu, X. Fully distributed adaptive sliding-mode controller design for containment control of multiple Lagrangian systems. *Syst. Control Lett.* **2014**, *72*, 44–52. [[CrossRef](#)]
317. Augugliaro, F.; Schoellig, A.P.; D’Andrea, R. Generation of collision-free trajectories for a quadcopter fleet: A sequential convex programming approach. In Proceedings of the 2012 IEEE/RSJ international conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 1917–1922.
318. Alonso-Mora, J.; Naegeli, T.; Siegwart, R.; Beardsley, P. Collision avoidance for aerial vehicles in multi-agent scenarios. *Auton. Robot.* **2015**, *39*, 101–121. [[CrossRef](#)]
319. Preiss, J.A.; Hönig, W.; Ayanian, N.; Sukhatme, G.S. Downwash-aware trajectory planning for large quadrotor teams. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 250–257.
320. Hönig, W.; Preiss, J.A.; Kumar, T.S.; Sukhatme, G.S.; Ayanian, N. Trajectory planning for quadrotor swarms. *IEEE Trans. Robot.* **2018**, *34*, 856–869. [[CrossRef](#)]
321. Alonso-Mora, J.; Montijano, E.; Nägele, T.; Hilliges, O.; Schwager, M.; Rus, D. Distributed multi-robot formation control in dynamic environments. *Auton. Robot.* **2019**, *43*, 1079–1100. [[CrossRef](#)]
322. Madridano, Á.; Al-Kaff, A.; Martín, D. 3D Trajectory Planning Method for UAVs Swarm in Building Emergencies. *Sensors* **2020**, *20*, 642. [[CrossRef](#)]
323. Zhou, X.; Wang, Z.; Wen, X.; Zhu, J.; Xu, C.; Gao, F. Decentralized Spatial-Temporal Trajectory Planning for Multicopter Swarms. *arXiv* **2021**, arXiv:2106.12481.
324. Zhou, X.; Wen, X.; Zhu, J.; Zhou, H.; Xu, C.; Gao, F. Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments. *arXiv* **2020**, arXiv:2011.04183.
325. Park, J.; Kim, H.J. Online Trajectory Planning for Multiple Quadrotors in Dynamic Environments Using Relative Safe Flight Corridor. *IEEE Robot. Autom. Lett.* **2020**, *6*, 659–666. [[CrossRef](#)]
326. Park, J.; Kim, J.; Jang, I.; Kim, H.J. Efficient multi-agent trajectory planning with feasibility guarantee using relative Bernstein polynomial. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 434–440.
327. Jackson, B.E.; Howell, T.A.; Shah, K.; Schwager, M.; Manchester, Z. Scalable cooperative transport of cable-suspended loads with UAVs using distributed trajectory optimization. *IEEE Robot. Autom. Lett.* **2020**, *5*, 3368–3374. [[CrossRef](#)]
328. Lee, T.; Sreenath, K.; Kumar, V. Geometric control of cooperating multiple quadrotor UAVs with a suspended payload. In Proceedings of the 52nd IEEE Conference on Decision and Control, Firenze, Italy, 10–13 December 2013; pp. 5510–5515.

329. Tognon, M.; Gabellieri, C.; Pallottino, L.; Franchi, A. Aerial co-manipulation with cables: The role of internal force for equilibria, stability, and passivity. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2577–2583. [[CrossRef](#)]
330. Loianno, G.; Spurny, V.; Thomas, J.; Baca, T.; Thakur, D.; Hert, D.; Penicka, R.; Krajnik, T.; Zhou, A.; Cho, A.; et al. Localization, grasping, and transportation of magnetic objects by a team of mavs in challenging desert-like environments. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1576–1583. [[CrossRef](#)]
331. Lindsey, Q.; Mellinger, D.; Kumar, V. Construction of cubic structures with quadrotor teams. In Proceedings of the Robotics: Science & Systems VII, Los Angeles, CA, USA, 27–30 June 2011.
332. Mellinger, D.; Shomin, M.; Michael, N.; Kumar, V. Cooperative grasping and transport using multiple quadrotors. In *Distributed Autonomous Robotic Systems*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 545–558.
333. Loianno, G.; Kumar, V. Cooperative transportation using small quadrotors using monocular vision and inertial sensing. *IEEE Robot. Autom. Lett.* **2017**, *3*, 680–687. [[CrossRef](#)]
334. Sanalitra, D.; Savino, H.J.; Tognon, M.; Cortés, J.; Franchi, A. Full-pose Manipulation Control of a Cable-suspended load with Multiple UAVs under Uncertainties. *IEEE Robot. Autom. Lett.* **2020**, *5*, 2185–2191. [[CrossRef](#)]
335. Shi, F.; Zhao, M.; Murooka, M.; Okada, K.; Inaba, M. Aerial regrasping: Pivoting with transformable multilink aerial robot. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 200–207.
336. Bayram, H.; Stefas, N.; Engin, K.S.; Isler, V. Tracking wildlife with multiple UAVs: System design, safety and field experiments. In Proceedings of the 2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS), Los Angeles, CA, USA, 4–5 December 2017; pp. 97–103.
337. Wu, J.; Wang, H.; Li, N.; Yao, P.; Huang, Y.; Su, Z.; Yu, Y. Distributed trajectory optimization for multiple solar-powered UAVs target tracking in urban environment by Adaptive Grasshopper Optimization Algorithm. *Aerosp. Sci. Technol.* **2017**, *70*, 497–510. [[CrossRef](#)]
338. Bourne, J.R.; Goodell, M.N.; He, X.; Steiner, J.A.; Leang, K.K. Decentralized Multi-agent information-theoretic control for target estimation and localization: Finding gas leaks. *Int. J. Robot. Res.* **2020**, *39*, 1525–1548. [[CrossRef](#)]
339. Zhou, L.; Leng, S.; Liu, Q.; Wang, Q. Intelligent UAV Swarm Cooperation for Multiple Targets Tracking. *IEEE Internet Things J.* **2021**. [[CrossRef](#)]
340. Schwager, M.; Julian, B.J.; Angermann, M.; Rus, D. Eyes in the sky: Decentralized control for the deployment of robotic camera networks. *Proc. IEEE* **2011**, *99*, 1541–1561. [[CrossRef](#)]
341. Nigam, N.; Bieniawski, S.; Kroo, I.; Vian, J. Control of multiple UAVs for persistent surveillance: Algorithm and flight test results. *IEEE Trans. Control Syst. Technol.* **2011**, *20*, 1236–1251. [[CrossRef](#)]
342. Julian, B.J.; Angermann, M.; Schwager, M.; Rus, D. Distributed robotic sensor networks: An information-theoretic approach. *Int. J. Robot. Res.* **2012**, *31*, 1134–1154. [[CrossRef](#)]
343. Keller, J.; Thakur, D.; Dobrokhodov, V.; Jones, K.; Pivtoraiko, M.; Gallier, J.; Kaminer, I.; Kumar, V. A computationally efficient approach to trajectory management for coordinated aerial surveillance. *Unmanned Syst.* **2013**, *1*, 59–74. [[CrossRef](#)]
344. Forster, C.; Lynen, S.; Kneip, L.; Scaramuzza, D. Collaborative monocular slam with multiple micro aerial vehicles. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3962–3970.
345. Schmuck, P.; Chli, M. Multi-uav collaborative monocular slam. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3863–3870.
346. Kegeleirs, M.; Grisetti, G.; Birattari, M. Swarm SLAM: Challenges and Perspectives. *Front. Robot. AI* **2021**, *8*, 23. [[CrossRef](#)]
347. Tipsuwan, Y.; Chow, M.Y. Control methodologies in networked control systems. *Control Eng. Pract.* **2003**, *11*, 1099–1111. [[CrossRef](#)]
348. Hespanha, J.P.; Naghshtabrizi, P.; Xu, Y. A survey of recent results in networked control systems. *Proc. IEEE* **2007**, *95*, 138–162. [[CrossRef](#)]
349. Wang, F.Y.; Liu, D. Networked control systems. In *Theory and Applications*; Springer: London, UK, 2008.
350. Matveev, A.S.; Savkin, A.V. *Estimation and Control over Communication Networks*; Birkhauser: Boston, MA, USA, 2009.
351. Bemporad, A.; Heemels, M.; Johansson, M. *Networked Control Systems*; Springer: London, UK, 2010; Volume 406.
352. Ge, X.; Yang, F.; Han, Q.L. Distributed networked control systems: A brief overview. *Inf. Sci.* **2017**, *380*, 117–131. [[CrossRef](#)]
353. Matveev, A.S.; Savkin, A.V. The problem of state estimation via asynchronous communication channels with irregular transmission times. *IEEE Trans. Autom. Control* **2003**, *48*, 670–676. [[CrossRef](#)]
354. Onat, A.; Naskali, T.; Parlakay, E.; Mutluer, O. Control over imperfect networks: Model-based predictive networked control systems. *IEEE Trans. Ind. Electron.* **2010**, *58*, 905–913. [[CrossRef](#)]
355. Petersen, I.R.; Savkin, A.V. *Robust Kalman Filtering for Signals and Systems with Large Uncertainties*; Birkhauser: Boston, MA, USA, 1999.
356. Goodwin, G.C.; Silva, E.I.; Quevedo, D.E. Analysis and design of networked control systems using the additive noise model methodology. *Asian J. Control* **2010**, *12*, 443–459. [[CrossRef](#)]
357. Savkin, A.V.; Petersen, I.R. Set-valued state estimation via a limited capacity communication channel. *IEEE Trans. Autom. Control* **2003**, *48*, 676–680. [[CrossRef](#)]

-
358. Malyavej, V.; Savkin, A.V. The Problem of Optimal Robust Kalman State Estimation via Limited Capacity Digital Communication Channels. *Syst. Control Lett.* **2005**, *54*, 283–292. [[CrossRef](#)]
 359. Savkin, A.V. Analysis and synthesis of networked control systems: Topological entropy, observability, robustness and optimal control. *Automatica* **2006**, *42*, 51–62. [[CrossRef](#)]
 360. Savkin, A.V.; Cheng, T.M. Detectability and output feedback stabilizability of nonlinear networked control systems. *IEEE Trans. Autom. Control* **2007**, *52*, 730–735. [[CrossRef](#)]