# A Multiple-UAV Architecture for Autonomous Media Production

**Ioannis Mademlis* · Arturo Torres-González · Jesús Capitán · Maurizio Montagnuolo · Alberto Messina · Fulvio Negro · Cedric Le Barz · Tiago Gonçalves · Rita Cunha · Bruno Guerreiro · Fan Zhang · Stephen Boyle · Gregoire Guerout · Anastasios Tefas · Nikos Nikolaidis · David Bull · Ioannis Pitas**

**Abstract** Cinematography with *Unmanned Aerial Vehicles* (UAVs) is an emerging technology promising to revolutionize media production. On the one hand, manually controlled drones already provide advantages, such as flexible shot setup, opportunities for novel shot types and access to difficult-to-reach spaces and/or viewpoints. Moreover, little additional ground infrastructure is required. On the other hand, enhanced UAV cognitive autonomy would allow both easier cinematography planning (from the *Director*'s perspective) and safer execution of that plan during actual filming; while integrating multiple UAVs can additionally augment the cinematic potential. In this paper, a novel multiple-UAV software/hardware architecture for media production in outdoor settings is proposed. The architecture encompasses mission planning and control under safety constraints, enhanced cognitive autonomy through visual analysis, human-computer interfaces and communication infrastructure for platform scalability with Quality-of-Service provisions. Finally, the architecture is demonstrated via a relevant subjective study on the adequacy of UAV and camera parameters for different cinematography shot types, as well as with field experiments where multiple UAVs film outdoor sports events.

**Keywords** Multiple-UAV cooperation · media production · UAV cinematography · autonomous drones

## 1 Introduction

Originally conceived as strategic military tools, *Unmanned Aerial Vehicles* (UAVs or "drones") have gradually become highly useful in several domains, e.g., for scientific data collection, agricultural applications, or in infrastructure inspection. Particularly, their use is on a steep rise for amateur and professional media production, where they mainly serve as compact aerial cameras. Camera-equipped UAVs possess interesting features: they can cover a scene of interest from different viewpoints, producing more aesthetically pleasing shots; they can operate in places that are difficult to access; they are ideal to cover outdoor events in large spaces, as they do not depend on previously installed infrastructure; etc. UAVs for media production can substitute both dollies (static cranes) and helicopters, thanks to their higher flexibility, easier deployment and lower cost. Their prominence in

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE). This publication reflects the authors' views only. The European Commission is not responsible for any use that may be made of the information it contains.

Ioannis Mademlis (corresponding author) E-mail: imademlis@csd.auth.gr · Anastasios Tefas E-mail: tefas@csd.auth.gr · Nikos Nikolaidis E-mail: nnik@csd.auth.gr · Ioannis Pitas E-mail: pitas@csd.auth.gr
Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

Arturo Torres-González E-mail: arturotorres@us.es · Jesús Capitán E-mail: jcapitan@us.es
GRVC Robotics Lab, University of Seville, Seville, Spain

Maurizio Montagnuolo E-mail: maurizio.montagnuolo@rai.it · Alberto Messina E-mail: alberto.messina@rai.it · Fulvio Negro E-mail: fulvio.negro@rai.it
RAI - Centre for Research and Technological Innovation, Torino, Italy

Cedric Le Barz E-mail: cedric.lebarz@thalesgroup.com · Tiago Gonçalves E-mail: tiago-f.goncalves@thalesgroup.com
Thales - Advanced studies department THERESIS, Palaiseau, France

Rita Cunha E-mail: rita@isr.tecnico.ulisboa.pt · Bruno Guerreiro E-mail: bguerreiro@isr.ist.utl.pt
Institute for Systems and Robotics (ISR/LARSyS), Instituto Superior Tecnico, Lisbon, Portugal

Fan Zhang E-mail: fan.zhang@bristol.ac.uk · Stephen Boyle E-mail: stephen.boyle@bristol.ac.uk · David Bull E-mail: dave.bull@bristol.ac.uk
Department of Electrical and Electronic Engineering, University of Bristol, Bristol, United Kingdom

Gregoire Guerout E-mail: gregoire.guerout@alerion.fr
Alerion, Nancy, France

cinematography applications has been highly boosted during the past decade, as they became more affordable and their technology improved at a rapid pace.

Shooting with a remotely-operated drone is still the norm in industry practice for professional aerial cinematography. Two operators are typically required, a pilot to control the vehicle and a camera operator to handle the camera. Pilots must also take care of safety and regulatory issues, like keeping a maximum flight height or a permissible proximity to human crowds. This burden can be reduced by employing recent commercial drones that incorporate some autonomous/cognitive functionalities [10, 49, 52]. However, these systems are limited to rudimentary single-UAV filming, either in the form of chasing a moving target, or by following pre-defined routes, and they do not address all cinematographic principles.

Using fleets of cooperating drones for media production would allow operators to film different targets concurrently, permitting novel three-dimensional visual effects. Additionally, it would increase spatiotemporal scene coverage, by reducing "dead" time intervals due to vehicles traveling between desired viewpoints. However, deploying manually-operated drone fleets for professional filming significantly raises logistics costs: both the number of human operators and their cognitive load are increased. Therefore, UAV fleets with advanced autonomous functionalities, minimizing the need for human intervention, are clearly the future of aerial media production.

Considering the above, this paper presents a novel multiple-UAV architecture for media production. The architecture integrates all relevant players, namely a media *Director*, a flight safety *Supervisor* and several UAVs with professional onboard cameras. The Director is in charge of defining shots and organizing media production from the aesthetic point of view; whereas the Supervisor is in charge of ensuring flight safety during production. A multiple-UAV architecture for media production operating in outdoor events is challenging from several aspects: (i) it should provide interfaces for human operators with different background, e.g., media production crew and flight safety staff (pilots); (ii) a significant degree of decisional and functional autonomy is required to reduce the cognitive load of human operators; (iii) UAVs must satisfy safety and aesthetic constraints while filming the desired shots; and (iv) an efficient communication infrastructure is necessary for sharing video streams and other information (e.g., telemetry).

The main contribution of this paper is the multiple-UAV system architecture itself, which integrates mission design and description, as well as mission planning and execution, to yield autonomous media production in outdoor settings using a coordinated UAV fleet. This was the main objective of the EU-funded project MULTIDRONE [1], which served as a framework for the work proposed. Previously published articles have presented the specific algorithms implemented for each individual component in the system. In contrast, here the focus is on the overall architecture that integrates the multiple hardware platforms/modules and perception, planning or control algorithms, which is showcased for the first time in detail.

In general, the MULTIDRONE architecture includes the following features:

- A human-computer interface for the media production team. Different shots can be specified and encoded by a novel language to describe cinematography missions.
- A human-computer interface for the security Supervisor.
- Functionalities for autonomous multiple-UAV mission planning and execution. The UAVs can autonomously perform a wide range of shots.
- A communication infrastructure for video streaming with Quality-of-Service functionalities.
- Embedded computing on-board the UAVs allows for 3D target (e.g., sportsmen) localization and tracking, while avoiding obstacles, collisions between them and entering into each other's field of view.

The remainder of this paper is structured as follows: Section 2 overviews the current state of the art in UAV media production; Section 3 formulates the problem to solve; Section 4 presents an overview of the proposed architecture; Section 5 introduces the relevant Graphical User Interfaces that have been designed and implemented; Sections 6 and 7 detail the software and hardware components, respectively; Section 8 describes the communication architecture; Section 9 discusses relevant aspects of the architecture, particularly, its scalability and its safety; Section 10 presents an empirical evaluation of the proposed architecture, both on real hardware and in simulation; while Section 11 provides conclusions drawn from the preceding discussion, as well as possible avenues for future research and development.

## 2 State-of-the-art in UAV Media Production

The main shooting scenarios in typical media production [29, 30] are the following ones:

- off-line shooting with full post-production editing (i.e., for TV programmes or movies);
- filming of live events for deferred broadcast and, thus, with potential post-production modifications (i.e., for deferred TV programmes);
- full live event shooting (i.e., for live TV programmes) with limited post-production effects.

In all scenarios, current practice in media production with UAVs requires the Director to prespecify the targets to be filmed, i.e., subjects or areas of interest within the scene.

[1] https://multidrone.eu/

Then, she/he designs a *cinematography plan* in preproduction, composed of a temporally ordered sequence of target assignments and desired filming shot types, which the pilot and the cameraman, acting in coordination, attempt subsequently to implement during shooting. When performed in fully manual mode, this becomes a challenging task.

Relevant research has been carried out on standardization for drone cinematography. One aspect to study is a useful UAV shot type taxonomy, serving as an aesthetically meaningful vocabulary of visual building blocks. The various shot types in UAV cinematography can be described using two complementary criteria: the framing shot type (FST) and the camera motion type (CMT). Each CMT can be successfully combined with a subset of the possible FSTs, according to Director's specifications, so as to achieve a pleasant visual result. The FSTs are primarily defined by the relative size of the main subject/target being filmed (if any) to the video frame size (e.g., medium shot, close-up, etc.); while the CMTs are defined by the drone trajectory relative to the moving or still target (e.g., orbit, chase, fly-by, etc.).

In the context of the proposed system, a full UAV shot type taxonomy has been developed and formalized, consisting of 26 CMTs, 8 FSTs and 4 multiple-UAV shot types. The latter have been derived by identifying different pleasing combinations of multiple single-UAV camera motion types, assembled in meaningful sequences. Additionally, a set of constraints and relevant rules have been analytically derived for several CMTs. These estimate the achievable zoom level (and, therefore, feasible FSTs), so that computer vision algorithms for visual target tracking do not fail. The above work has been fully detailed in [23–25, 28, 31].

A second important aspect of UAV cinematography is the need for a language that can compactly and accurately describe a shooting mission in a formal manner. In the context of the proposed architecture, this would facilitate the interaction between the Director and the envisioned autonomous system. Thus, an attempt at building such a language for the case of full live event shooting was carried out and described in [28, 35].

Regarding autonomous platforms, relevant research and commercial product availability has intensified over the past years. For instance, there are some end-to-end solutions for semi-autonomous aerial cinematographers [15, 20], where a Director specifies high-level commands such as shot types and positions, while the drone is in charge of autonomously implementing the navigation functionality. An outdoor application for filming people is proposed in [20], where different types of shots from the cinematography literature are introduced (e.g., close-up, external, over-the-shoulder, etc). Timing for the shots is considered by means of an easing curve that drives the drone along the planned trajectory (i.e., the curve can modify its velocity profile). In [15], an iterative quadratic optimization problem is formulated to obtain smooth trajectories for the camera and the look-at point (i.e., place where the camera is pointing at). In general, these works present interesting theoretical properties, but they are restricted to offline optimization with a fully known map of the scenario, and static or close-to-static guided tour scenes, i.e., without moving actors.

In the robotics literature, some recent works have also considered cinematographic principles for filming dynamic targets in outdoor scenarios. In this vein, visual tracking and camera motion planning considering collisions and aesthetic constraints is performed in [4]. A method based on reinforcement learning has also been proposed [16] to achieve visually pleasant shots. Similarly, the authors in [19] present an algorithm to imitate (learning from demonstration) professional cameraman's intentions, for capturing aerial footage of a single subject. The authors in [5] propose a system for aerial cinematography with a single UAV that combines vision-based target localization with a real-time camera motion planner optimizing smoothness, collisions and artistic guidelines. They show impressive field experiments, but their focus is mainly on mapping and obstacle avoidance, rather than multi-shot scheduling. Moreover, only a simplified set of shots is considered: left, right, front, back. In general, these papers provide results quite interesting in terms of outdoors operation or online trajectory planning, but are always restricted to a single-UAV scenario.

In the case of multiple cameras filming simultaneously, additional challenges arise. In [3], an optimization-based algorithm is proposed for computing a single, aesthetically pleasing video, conforming to basic cinematic guidelines (such as the 180-degree rule and jump cut avoidance). Raw feeds coming from multiple cameras are used. Operating also within a multiple-camera context, automated editing has been considered as a problem of camera selection over time [9]. A framework for automatically computing a variety of cinematically plausible shots from a single input video, suitable to the special case of live performances, is presented in [13]. It acts as a virtual camera assistant to the film editor, who can assemble novel shots in the editing room with a combination of high-level instructions and manually selected key-frames. The work in [48] proposes a method to place as few UAVs as possible to cover all the available targets in the scenario without occlusion. However, the relevant calculations are performed in a 2D space and assume that the cameras always face the targets. Moreover, smooth transitions are not considered, with only the best shooting points being computed. In [38], a method is presented to optimize 3D UAV trajectories for cinematography purposes. It resolves a non-linear optimization problem in a receding horizon fashion, taking into account collision avoidance constraints for multiple UAVs. This method extends a previous, single-UAV method [37] that only optimizes local trajectory segments. However, the approach is restricted to indoor settings and flight time constraints are not considered. The system in [12] is closer to the one presented in this paper, as the authors also propose a complete architecture for cinematography with multiple UAVs. A master-slave approach is used to coordinate the motion of the UAVs around dynamic targets: a single master UAV is supposed to be shooting the scene at a time, while the slaves offer alternative viewpoints or act as replacements. In comparison, the MULTIDRONE architecture presented here is more flexible, since different types of shots can be filmed concurrently. Moreover, it has also been demonstrated in outdoor settings, while [12, 38] used an indoor Vicon motion capture system that provided accurate positioning for all targets and UAVs.

Overall, these works present quite valuable contributions for cinematography with multiple UAVs, but the specifics of outdoor scenarios differ in several aspects, as the environment is less controlled: UAVs require more payload to carry on-board cameras with better lenses and equipment for larger range communication; achieving smooth trajectories is more complex due to external factors, such as wind gusts or communication delays; UAV positioning is less accurate in general; etc. [6] does propose quite an interesting multi-UAV architecture which is evaluated outdoors, although it does not consider either user interfaces nor hardware integration.

Certain commercial applications, also oriented towards outdoor single-UAV cinematography planning, have been released within the last few years. Notably, *Skywand* [14] is a virtual reality system, allowing the user to aerially explore a 3D graphics model of the scene that she/he wants to cover and identify/place desired key-frames within the virtual environment. The system then computes the real drone trajectory, as well as the corresponding sequence of camera rotations, required for a smooth shot containing these key-frames to actually be filmed. *FreeSkies CoPilot* [11] is a mobile software suite, offering similar functionality but with a simple 3D map instead of a virtual reality interface. In both cases, the resulting drone autonomy and environment perception are minimal, the cinematography plan consists of example key-frames, the computed flight paths cannot be adjusted online and no-fly zones (due to legal restrictions) are not integrated.

In summary, there is still a need for comprehensive approaches to achieve autonomous aerial cinematography with multiple UAVs. The existing methods only focus on some of the sub-problems addressed in MULTIDRONE, are limited to controlled indoor settings, do not cope with the limitations of video communication channels, or support only a reduced set of specific cinematic shot types.

## 3 Problem Formulation

In the proposed multiple-UAV architecture, the Director can design *Shooting Missions* using a graphical interface. In a Shooting Mission, the Director describes the set of shots (i.e., *Shooting Actions*) that need to be performed. Each shot or Shooting Action contains specific parameters like the shot type (e.g., CMT, FST, etc.), the duration, the starting position, and so on; and it is associated with a triggering *Event* (e.g., the start/end of a race, targets reaching a point of interest, etc.). Shooting Missions are translated into an XML-based language [35] that can be interpreted by the autonomous multi-UAV system. Then, the system is able to compute feasible plans to assign Shooting Actions to the different UAVs and execute them. In the mission description, it is assumed that the Director can know or predict the occurrence time for the Events which trigger Shooting Actions, as well as the target trajectories. During mission execution, the Director may command on-the-fly replanning if the actual target motion differs significantly from the prespecified/foreseen one.

In order to carry out a Shooting Mission, the architecture needs to consider components to design the mission itself, to plan it and to execute it. The architecture splits mission planning and execution into two problems to be solved sequentially:

**Problem 1** Given a set of UAVs with initial positions and remaining battery levels, and a set of Shooting Actions designed by the Director to be performed; assign Shooting Actions to UAVs (the complete shot or a portion) to cover as much percentage of the whole mission as possible, respecting the battery duration of each UAV.

**Problem 2** Given a set of UAVs with different shots assigned, and an estimated trajectory for the target to be filmed; compute the required UAV trajectories and UAV-mounted gimbal/camera rotations over time, in order to film the assigned shots as smoothly as possible, while avoiding collisions and camera occlusions.

Plans computed to solve Problem 1 need to comply with temporal constraints (i.e., starting time of each Shooting Action) and battery duration, thus it is necessary to predict the arrival time of each UAV to the starting position of its Shooting Actions, as well as its remaining battery. The UAV controllers developed to solve Problem 2 can be used to derive those estimations. Then, during mission execution, inaccuracies in those predictions can be addressed by triggering a replanning mechanism to adjust plan deviations. Mathematical details about the formulation and solution for Problem 1 and Problem 2 can be found in [7] and [1], respectively.

## 4 Multiple-UAV System Overview

In order to solve the problems stated in Section 3, the MULTIDRONE consortium followed a thorough process for designing, implementing, and testing the envisioned system. This process was divided in four phases. First, end-users (in this case, the Italian and German television broadcasters RAI and DW) provided a set of media production requirements for the system, which were collected in a public document [36]. Those requirements were organized in three fundamental layers: usage scenarios (the objectives), media production (the process through which the objectives were to be achieved), and system platform (the infrastructure enabling and supporting the process). In the second phase, the end-user requirements seved as a basis for deriving the specifications and the design of the envisioned modular multi-actor system architecture, its communications, as well as its functionalities. The third phase entailed development of the software and its integration on the hardware. During this phase, the software was continuously tested in simulation and in preliminary experiments with the actual hardware. Finally, the system was evaluated with experiments in mock-up and real scenarios, as described later in Section 10.

The MULTIDRONE system developed as a product of this methodology consists of multiple interacting components that reside either on-board the UAVs, or on a central *Ground Station*. Figure 1 depicts an overview of these components.

The Ground Station contains the following relevant parts:

- ***Director's Dashboard***. The Director's Dashboard is a component for Human-Computer Interaction (HCI) with the media production team. The Director and her/his editorial team can specify the Shooting Missions with different Shooting Actions associated with Events. This information is provided before shooting commences, but during production the Director can also trigger new shots or stop/replace previous ones on-the-fly. Additionally, the Dashboard allows for manual control of the cameras on-board the UAVs.
- ***Supervision Station***. This component is also used for HCI with the flight safety Supervisor, a person in charge of validating and monitoring the safety and security of a mission. After checking that a preplanned mission is safe, this Supervisor operator will validate it for execution. She/he will also monitor the status of the UAVs through the Supervision Station, so as to cancel the mission in case any security issue arises during execution.
- ***Mission Planning and Execution***. These components are responsible for transforming the Shooting Mission specified by the Director into individual plans for each UAV. A plan is computed to allocate the requested shots to the available UAVs and compute their corresponding paths. During production, the mission progress is monitored and replanning is triggered if any UAV facing an emergency is not available anymore, or if the Director decides so (e.g., by sending new shots).
- ***Perception and Mapping***. These components provide semantic mapping functionalities to constantly update a preconstructed map of the environment, for instance with information about no-fly zones due to human crowd gathering. Additionally, there are components for fusing information derived from multiple sources (i.e., cameras on-board the UAVs, GNSS receivers on the target, etc.) and estimating the 3D position of targets that hold interest for media production.

Apart from the Ground Station, the architecture is distributed along the fleet of UAVs with additional modules that run on-board. These modules are in charge of executing the plan assigned to each UAV and performing visual target tracking with the on-board cameras. Multiple functionalities, such as autonomous UAV localization and navigation, collision avoidance and camera control are implemented. Furthermore, bidirectional connectivity between the UAVs and the Ground Station is provided by a communication module based on LTE with Quality of Service (QoS) capacities, whereas inter-UAV connectivity is achieved via Wi-Fi.

The various architectural components are detailed in the next sections: Section 5 will expand upon the HCI components, i.e., Dashboard and Supervision Station, the remaining software modules will be explained in Section 6, while the system hardware and the communication infrastructure will be presented in Sections 7 and 8, respectively.

## 5 Human-Computer Interfaces

This Section describes the Human-Computer Interfaces of the MULTIDRONE architecture. The Director and her/his media production team interact with the system through the Dashboard, while the Supervisor does so through the Supervision Station.

### 5.1 Director's Dashboard

The Dashboard is a software tool used by the production team to govern the multiple-UAV system from the editorial point of view, both in the preproduction and the production phases. The main objective of the Dashboard is to support the Director in the creation and execution of Shooting Missions.

Shooting Mission descriptions may be inserted and modified through the Dashboard GUI (Graphical User Interface) during the preproduction stage, until a satisfactory result (from the editorial point of view) is achieved. Then, data are exported to an XML file and sent to the mission planning and execution component.

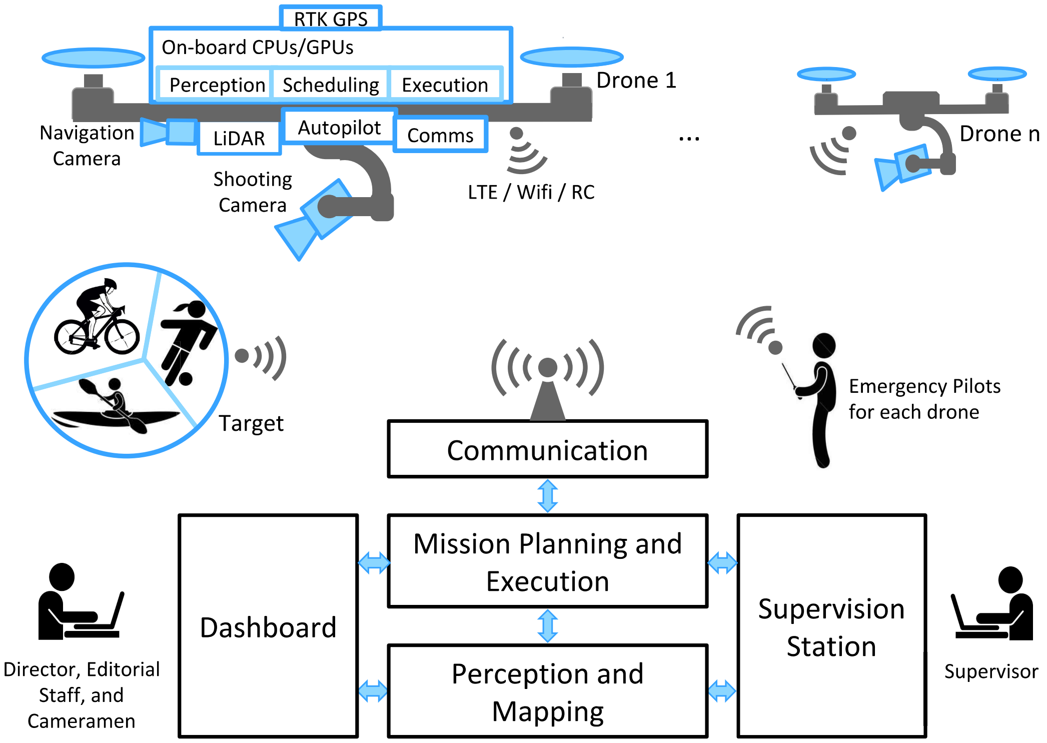

Fig. 1: Overview of the proposed multiple-UAV architectural components: on-board (top) and Ground Station (bottom).

The GUI is designed to be responsive and accessible through any kind of device, from smart-phones to desktop computers.It allows the Director to configure the UAV fleet formation, to define the Shooting Mission via the live Shooting Action timeline (the timeline of Events for which there are shots planned), and to intervene to filming parameters through the live camera/gimbal controls for each UAV. The UAV fleet formation is configured using a map of the geographical region where the mission will take place, overlayed with graphics showing the position of the formation, as well as the position of the target and its expected trajectory. Upon Director's request, the current Shooting Action can be visualized on the map.

Further details concerning the Dashboard can be found in [28, 35].

### 5.2 Supervision Station

The role of the Supervision Station is to reduce the workload of the Supervisor, allowing her/him to guarantee the safe execution of multiple-UAV missions. The ultimate purpose of the Supervision Station is to replace all UAV pilots as soon as regulation allows for it, by providing the same flight safety mechanisms available on a pilot's Radio Control station.

Thus, the Supervision Station includes a suitably designed GUI that displays all required information, so that the Supervisor can have a clear overview of the situation at any time. This GUI includes:

- a map, where flying UAVs are visualized with overlaid information (planned trajectories, forbidden no-fly zones and various aerial semantic annotations, e.g., NOTAM - Notice To AirMan);
- all video streams from the UAV navigation cameras;
- telemetry information (battery status, altitude above ground, vertical speed, etc.) for each UAV;
- the action status for each UAV, i.e., whether the UAV is taking off, heading towards the starting point of a Shooting Action, following a target, etc.

Through this GUI, the operator can: i) check and validate the safety of the flight plan that originates from the Shooting Mission, both when it is created and whenever changes are introduced to it; ii) monitor the mission execution, including the overall state of the UAVs; iii) abort the mission for safety reasons; and iv) insert manually safety- and logistics-related annotations in a semantic map.

The Supervision Station receives requests from the Mission Controller module, in order to check and validate the safety of a mission plan. Based on telemetry information, it automatically performs basic security checks (e.g., whether the UAV altitude exceeds a security threshold, vertical speed, battery level, etc.). In case any problems arise, an alarm notifies both the human Supervisor and the Mission Controller with a message, describing the type of alarm and the requested action. A response from the Supervisor is then expected, if the decision can wait, otherwise the Supervision Station is able to decide by itself. During mission execution, the Supervisor may also decide to abort the mission at any time due to a risky situation, by sending a notification to the Mission Controller.

## 6 Software Architecture

This Section presents the software architecture of the overall system, describing the functionality of the different modules on the Ground Station and on-board each UAV.

### 6.1 Ground Station Software

The main goal of the software modules on the Ground Station is to receive a Shooting Mission from the Director, compute a plan for that mission and distribute it among the available UAVs. Afterwards, the mission execution is constantly monitored by the Ground Station, for double-checking security and replanning, if needed. Additionally, the modules on the Ground Station provide important functionalities, such as centralized target tracking, human crowd detection and semantic map management. Figure 2 shows the software components on the Ground Station, which are enumerated below.

#### *6.1.1 Dashboard*

This module contains the graphical tool described in Section 5 so that the editorial team can specify Shooting Missions, i.e., multiple Shooting Actions associated with different Events happening in time. This information is written into an XML format and sent to the Mission Controller to start a mission. During mission execution, the Director can incorporate new Shooting Actions, or start/stop specific Shooting Actions. The functionalities and the design of this module have been described in detail in [34].

#### *6.1.2 Supervision Station*

This module contains the GUI software for the Supervisor described in Section 5. The Supervision Station is used to check security before and during the mission. The Mission Controller communicates with this module to ask the Supervisor for security permission before starting the execution of a mission.

#### *6.1.3 Mission Controller*

This module is the center of the planning architecture. It receives the Shooting Mission from the Dashboard, asks the High-level Planner for a feasible plan, double-checks the plan with the Supervision Station and finally sends its corresponding actions to each UAV. Afterwards, it monitors mission execution and triggers new replanning procedures if requested so by the Director, or if there is an emergency with any UAV (not available anymore).

#### *6.1.4 High-level Planner*

This module computes a plan for a Shooting Mission, solving Problem 1 in Section 3. This plan consists of a list of single-UAV shots assigned to each drone and the navigation actions required to travel between them. Thus, the module integrates a UAV path planner that is used to estimate collision-free paths for the drones when performing the desired shots, as well as their navigation actions. The particular planning algorithm employed, detailed in [7], is a graph-based method able to find an optimal solution of a discrete optimization problem, in order to maximize the filming time while considering battery constraints for a single UAV in polynomial time. Then, a greedy strategy is applied to solve the problem sequentially for multiple UAVs.

#### *6.1.5 Event Manager*

This module manages the generation of Events, which are used to trigger certain actions on the UAVs. First, a system Event is generated in case a UAV reports an emergency. In such a scenario, the Mission Controller decides whether a new plan is necessary, while the UAV executes an emergency maneuver. The rest of the Events are related to filming and trigger associated Shooting Actions. For instance, possible Events are a race start, the approach to a prespecified point of interest that can be used for opportunistic shooting, etc. The Event Manager generates some Events automatically (e.g., a cyclist reaching a certain position), or waits for Director commands to generate them (e.g., the start of a race).

#### *6.1.6 Global 3D Tracker*

This module fuses information from all visual target detectors and trackers on-board the UAVs (i.e., from 2D Visual Information Analysis modules) and from on-target GNSS sensors (if available), in order to compute 3D target positions. This 3D target tracker is a stochastic filter that produces an estimation of the pose of each target in the global coordinate system.

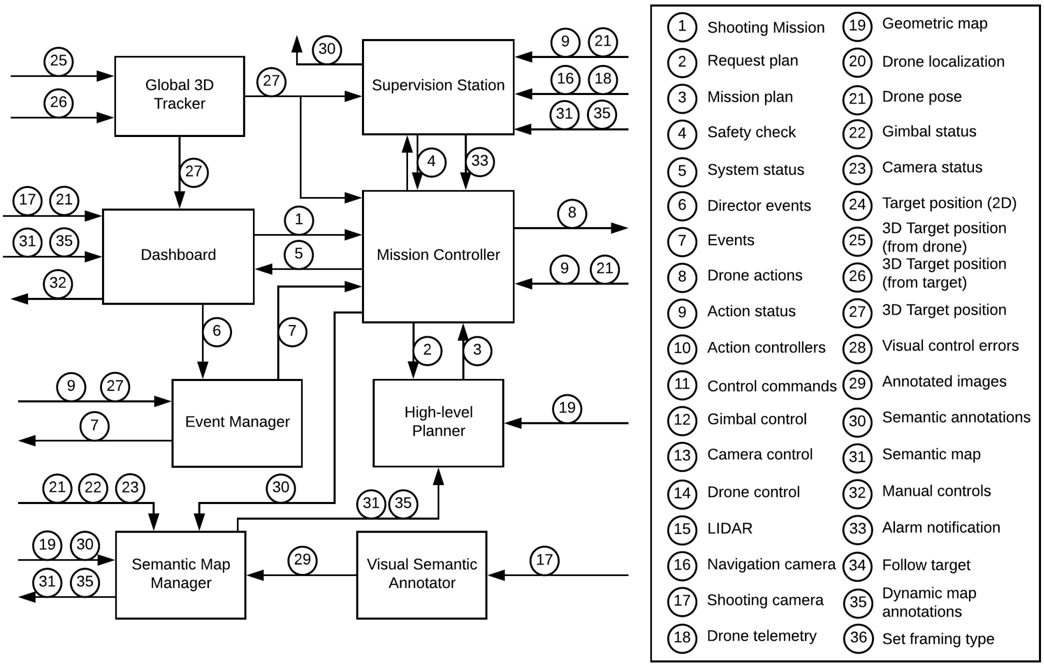

Fig. 2: Software functional diagram for the Ground Station.

### *6.1.7 Visual Semantic Annotator*

This module processes video streams from the cameras onboard the UAVs to detect (in 2D pixel coordinates) human crowds on the acquired video frames and localize them per-pixel. A Convolutional Neural Network (CNN) is employed to achieve this goal, able to independently extract a 2D crowd heatmap from each successive video frame coming from the UAVs. From an algorithmic perspective, two different CNNs were integrated into this module as alternative options. The first one [51] is a custom, lightweight, fully convolutional neural architecture. The second one [42] is a more complex and memory-demanding, but more accurate, hybrid of a CNN and a conditional Generative Adversarial Network (GAN), built on top of a ResNet-18 backbone [17], that actually performs human crowd semantic segmentation on the input image. Both models were pretrained on a desktop PC using manually annotated datasets, before being integrated into the Visual Semantic Annotator module at inference mode only.

### *6.1.8 Semantic Map Manager*

This module manages a semantic map with two types of (geo-localized) annotations useful for mission planning. Static annotations in Keyhole Markup Language (KML) format, that are specified in preproduction through the Supervisor Station, indicate immutable no-fly zones, landing zones, points of interest, etc. Dynamic map annotations are computed during mission execution in the form of polygon lines, representing evolving no-fly zones that arise due to human crowd gatherings. This module's functionality and design is described in [21, 22]. In short, it is composed of two submodules: a) the *Map Manager* is responsible for storing and updating the annotated 3D map (stored as an Octomap [18]), and b) the *Region Projector* is responsible for delineating 3D map regions obtained from the 2D-to-3D back-projection of image plane annotations, derived by the Visual Semantic Annotator, by exploiting known UAV position, gimbal orientation and camera parameters. As the UAV moves and its camera views new 3D terrain areas, the newly generated regions are merged with previously acquired ones, using the set union operator.

## 6.2 On-board UAV Software

The main goal of the software modules on-board each UAV is to execute its assigned shooting and navigation plan. The UAV has to localize itself, navigate while avoiding collisions and perform Shooting Actions. Figure 3 shows the software components on-board each UAV. Further details of the modules in charge of executing multi-UAV cinematography missions can be seen in [2].

### *6.2.1 On-board Scheduler*

This software module receives the list of actions corresponding to each UAV from the Mission Controller. Anytime the

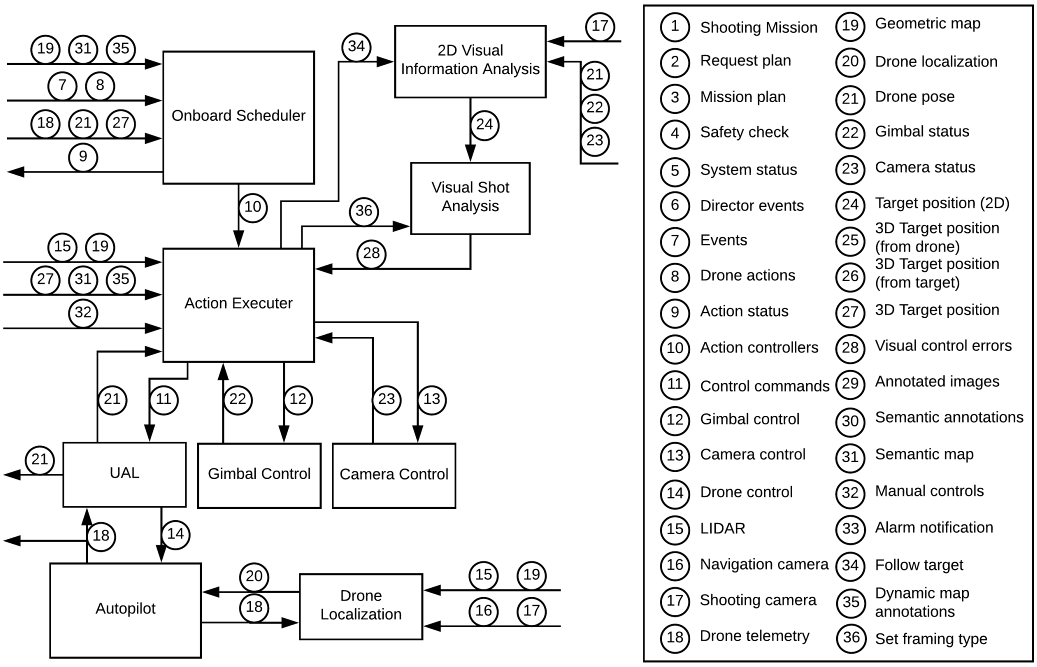

Fig. 3: Scheme of the software modules running on-board the UAVs.

Mission Controller decides to compute a new plan, the new list of actions is sent to each involved On-board Scheduler. The On-board Scheduler listens to the Event Manager and is in charge of executing the actions sequentially, when they are triggered, by sending requests to the Action Executer for each specific shot, as well as monitoring their execution status. This module is also in charge of computing a safe path to a landing spot in case of an emergency.

### 6.2.2 *Action Executer*

This module is responsible for the real-time control of the UAV, the camera gimbal, and the camera parameters (e.g., focal length) to yield the expected behavior of each shot. Basically, it solves Problem 2 in Section 3, using a controller that depends on the desired CMT, current drone position and target location. As detailed in [2], in order to obtain smooth reference trajectories, the motion of virtual trailer is simulated so that the reference to be tracked simply becomes a point on that trailer. This strategy also provides a reference frame tangent to the generated path, which can be used as heading reference. Based on these references, position and angular errors are defined and used to generate commands sent to a *UAV Abstraction Layer* [45], in the form of velocity or waypoint references. UAL translates them to autopilot commands. The controller also acts so that the camera gimbal tracks the target and retains its visual pose. The gimbal controller may be GPS-based or vision-based, in which case the 3D position of the target is not used and the gimbal velocity commands are generated from the estimated orientation of the camera and the visual control error defined directly in the image plane [8], as computed by the Visual Shot Analysis module (see Subsection 6.2.5). Finally, intrinsic camera parameters such as focal length are controlled to obtain the desired FST.

### 6.2.3 *UAV Localization*

This module is in charge of estimating the UAV pose based on the available on-board sensors, namely GNSS positioning, LiDAR, and video streams coming from cameras. All this information is fused by means of a Bayesian filter and compared with the pre-computed geometric map to estimate the current UAV pose. The LiDAR measurements are matched from one iteration to the next one to compute UAV odometry that feeds the filter.

### 6.2.4 *2D Visual Information Analysis*

This software module consists of a visual object detector and tracker, localizing on-frame (in 2D pixel coordinates) the filming target of each UAV. It receives an uncompressed video stream from the cinematic camera in real time and generates 2D positions of the tracked targets as bounding boxes, thus performing semantic visual analysis. Each 2D Region-of-Interest (ROI) on the image contains attached the camera and UAV poses, so that it can be later back-projected onto 3D space by the Global 3D Tracker. High-performance, real-time object detection and 2D visual tracking for embedded devices is achievable today with deep Convolutional Neural Networks [27] [47] [50] and neural correlation-based trackers [26] [53], respectively, that are typically executed in parallel on GP-GPUs. The algorithms employed in this module are detailed in [40] [39] [41] [44]; its most important building block is a lightweight deep neural object detector integrated with a much faster 2D visual object tracker, so that the first one automatically initializes the latter one to the target ROI detected the closest to the image center, while subsequently re-initializing it periodically on a need-to-run basis (i.e., when the module decides that the tracker has drifted and has lost track of the target). The involved lightweight deep CNNs were pretrained in a supervised manner on a regular desktop PC, using domain-specific, manually annotated datasets, and then deployed on the UAV computational hardware at inference mode only.

#### *6.2.5 Visual Shot Analysis*

This module receives desired shot specifications (target 2D position on the video frame, desired FST) and computes accordingly visual control errors. These errors encode deviations in target ROI size and position from the desired ones, relative to the video frame. Subsequently, current visual control errors are sent to the Action Executer, in order to appropriately control the gimbal pose and the camera focal length. The employed controller may also feed the gimbal interface with angular velocities, by operating either in the global 3D coordinate system or in a local 3D coordinate system, as suggested in [43].

## 7 Hardware Architecture

This Section describes the hardware architecture of the system, providing details about the equipment used in the Ground Station and on-board each UAV.

### 7.1 Ground Station Hardware

The Ground Station hardware is designed to be compact and lightweight, in order to facilitate logistics and deployment. Main components are depicted in Figure 4:

- A standard laptop equipped with a web browser to run the Dashboard.
- Two lightweight Intel NUCs with Intel Core i7 processors to run the Supervision Station and the modules for mission planning and execution, which are not computationally demanding.
- A more powerful workstation equipped with a GP-GPU for semantic visual analysis. This will run the modules related to perception and semantic mapping, i.e., the Visual Semantic Annotator and the Semantic Map Manager.
- An LTE radio base station for communication with the UAVs.
- An RTK-GNSS base station to broadcast RTK corrections. These corrections are used for precise 3D localization by GPS receivers on the UAVs. Relevant targets for media production may also be equipped with GPS receivers acquiring the RTK corrections. This information is then fused with the 2D detections from the cameras on-board the UAVs for better target positioning.

### 7.2 On-board UAV Hardware

The hardware on-board the UAVs is designed to provide the required functionalities for autonomous navigation and video shooting. Figure 5 depicts these components:

#### *7.2.1 UAV platform and core*

The DJI S1000+ frame was chosen for the system prototype, as it offers a good balance between, among others, payload (weight it can lift) and size. However, DJI stopped supporting this frame in the middle of the project, so other similar frames were selected for the experimental evaluation. The UAV core includes the Flight Control Unit. A Pixhawk 2.1, using the PX4 autopilot, an integrated Inertial Measurement Unit (IMU) and an external RTK-GPS, have been selected. The RTK-GPS provides decimeter-level position accuracy using a sensitive antenna. Additionally, an LTE/WiFi module provides communication with the Ground Station and with other UAVs, as described in Section 8. A parachute safety system may also be used in emergencies, when other contingency plans fail.

#### *7.2.2 Navigation payload*

It includes the additional hardware components required for autonomous navigation. There is a LiDAR used for obstacle avoidance and environment mapping. Furthermore, there is a navigation camera used by the Supervisor to monitor UAV movements (in First-Person-View mode) and double-check flight safety. These sensors are connected to the two on-board computers: an NVIDIA Jetson Tegra X2 and an Intel NUC. The software modules on-board the UAVs, described in Section 6, run on these computers, with the Tegra dedicated to visual analysis due to its powerful GP-GPU.

#### *7.2.3 Audiovisual payload*

This contains all the hardware components needed to acquire video for media production. It includes a Blackmagic

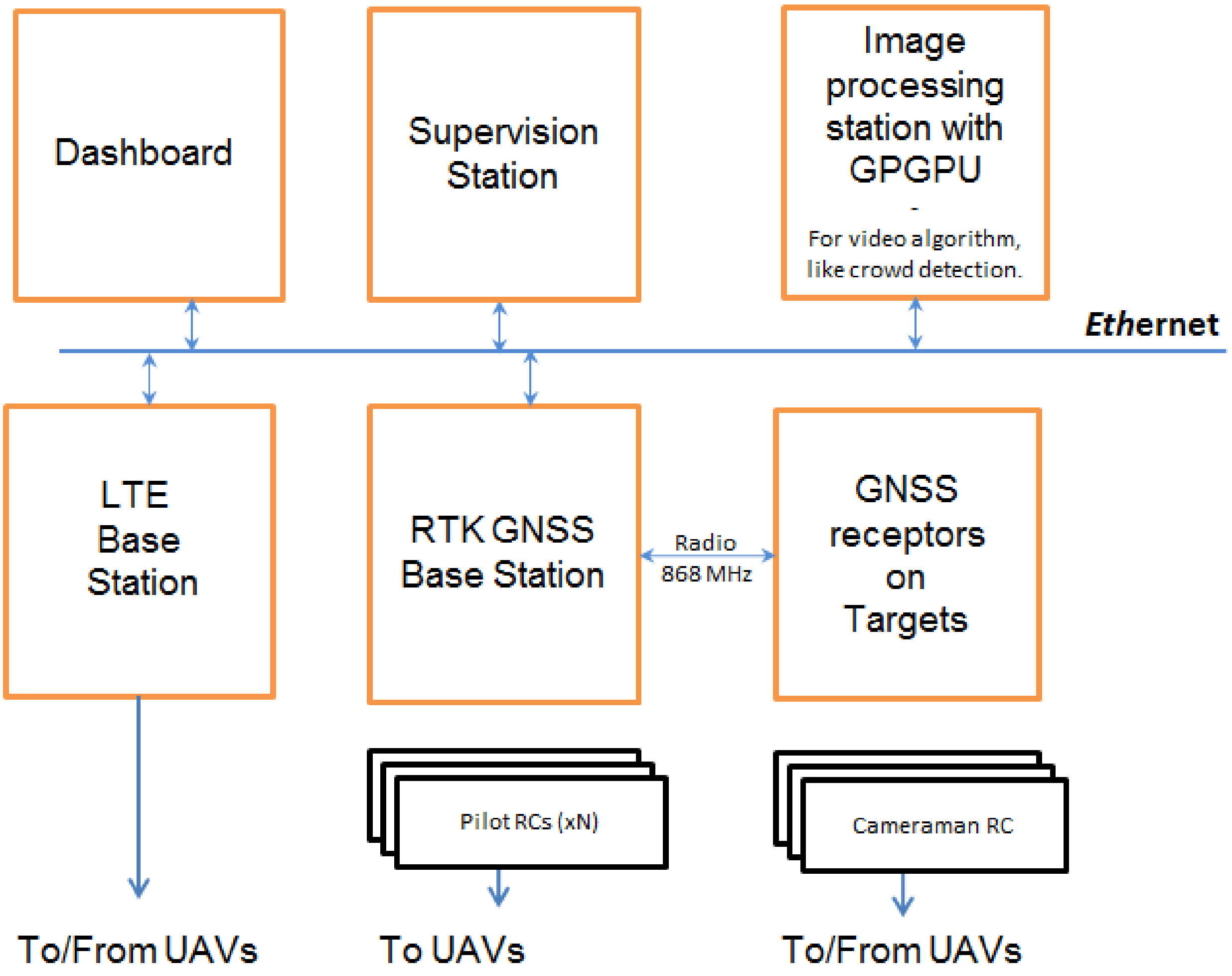

Fig. 4: Hardware architecture of the Ground Station.

Micro Cinema Camera camera with a motorized Panasonic x3 lens, providing high-quality and high-resolution (HDTV) images for fulfilling media production requirements. The camera is connected to the NVIDIA Jetson Tegra X2 computer and mounted on a 3-axis gimbal controlled by a Base-Cam controller.

## 8 Communication Infrastructure

An LTE communication module undertakes the majority of the communication exchanges between the UAVs and the Ground Station, composed of an LTE user equipment on-board and an LTE base station on-ground. Redundant RF communications are also provided for safety, through additional links. Inter-UAV communications are assured by a WiFi mesh. Each UAV carries on-board a dedicated *Communication* module that is responsible for:

- Acting as a default IP communication gateway/router to the ground and to other UAVs.
- Scheduling IP flows depending on applications' precedence and assigned IP Quality of Service (QoS).
- Traffic shaping/admission control when congestion occurs.
- Authentication, encryption and other security-related mechanisms.

The Communication module can be considered as a default IP router for the rest of the system. As such, it exposes an Ethernet interface to the computers on-board the UAV and implements a full IP protocol stack. Since it is fully independent from the other modules in the architecture, it has its own hardware and operating system (Linux OpenWRT).

In addition, a separate *Video Streamer* module is necessary for video transmission and interacts heavily with the Communication module. For each UAV, two video streams are generated: one by the navigation camera (H.264 compressed, 4:2:0 chroma sub-sampling, 640x480 pixels resolution) and another one by the cinematic camera (H.264 compressed, 4:2:2 chroma subsampling, 1920x1080 pixels resolution, @25fps). Since the main purpose of the navigation

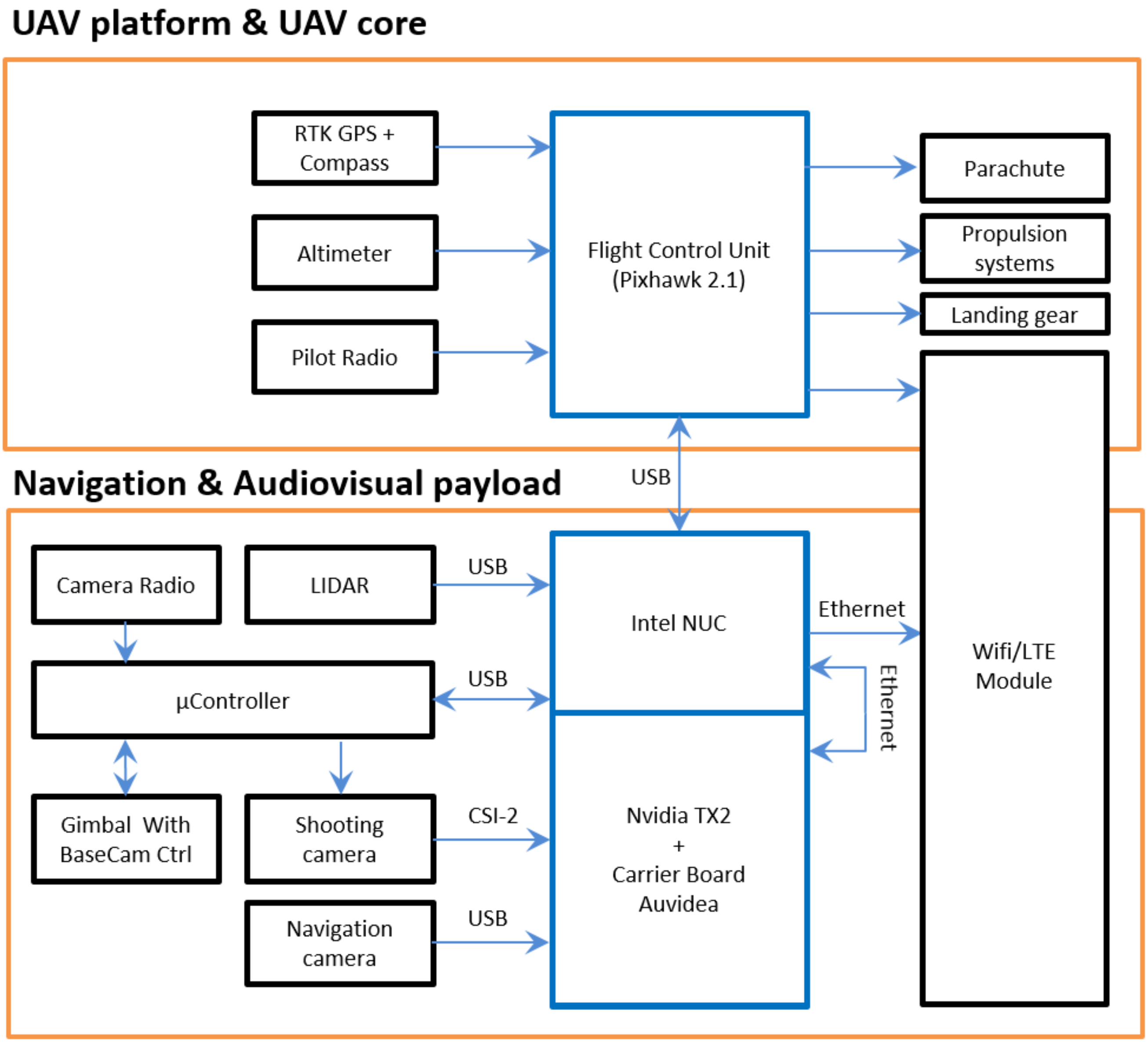

Fig. 5: Hardware architecture on-board each UAV.

camera is simply to provide the Supervisor with good situational awareness, Full HD resolution is not required. Video streams are then transmitted through the LTE radio network using the RTP protocol. The RTCP protocol is also used for on-ground synchronization of video streams coming from different UAVs, since RTP packets hold a timestamp. The Sender Report packet holds the correspondence between the RTP timestamp and the absolute timestamp (system hour), that is broadcasted through the LTE network thanks to the NTP protocol.

Figure 6 depicts the data flow for all video streams. It is assumed that, at each time instance, several UAVs are simultaneously connected to the Ground Station. The cinematic camera video streams are transmitted to the Dashboard through the radio network. In parallel, the streams are also resized so that they can be processed on-board by the perception modules. The Video Streamer can process either the images coming from the cinematic camera, or from the navigation camera, depending on the situation. On-ground, these streams will be uncompressed to be displayed on the Dashboard and, also, resized to be processed by the Visual Semantic Annotator.

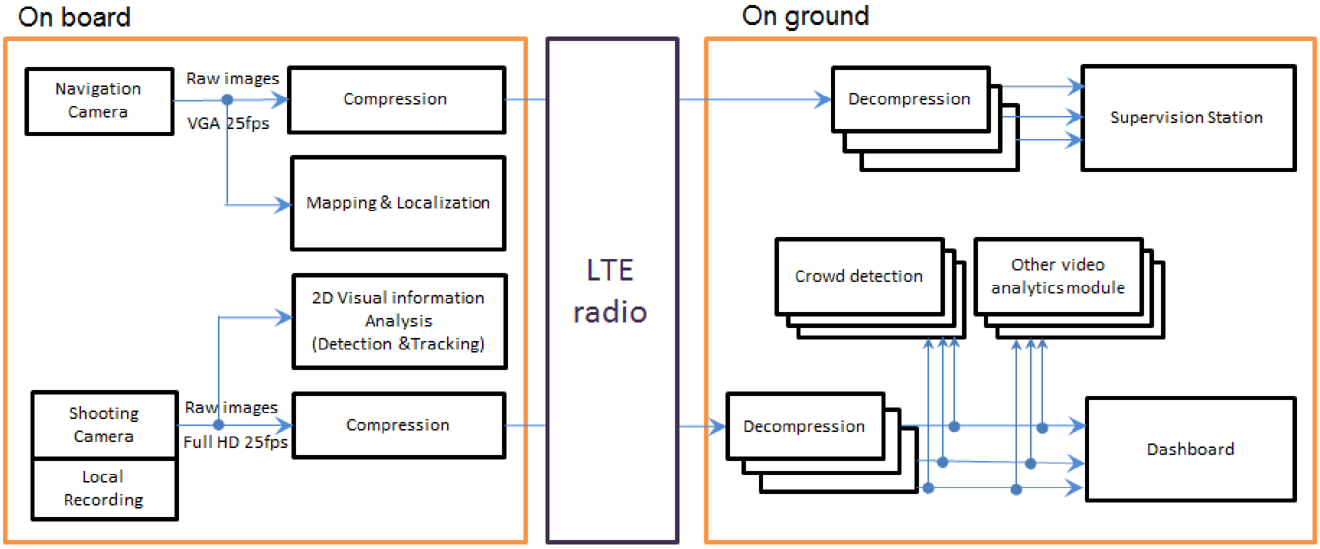

Fig. 6: Data flow for video streaming from the UAVs to the ground.

## 9 System Scalability and Safety

This Section discusses additional aspects of the proposed multiple-UAV architecture. In particular, scalability and safety are key factors in determining the usefulness of the architecture for media production.

### 9.1 Scalability

The proposed architecture was designed primarily for small fleets of 3-4 UAVs, as these UAVs are enough to cover outdoor sports events. However, the architecture may easily handle larger fleets (in the order of ten) with slight modifications. For instance, the Dashboard scales automatically to larger UAV teams without any issues, while the mission planning algorithms (implemented as centralized modules on the Ground Station) may be transparently replaced with alternative distributed versions, able to scale efficiently with an increasing number of UAVs. Once the plan is computed, it is immediately distributed along the UAVs; thus, mission execution is not affected by fleet size.

In contrast, the requirements of the visual analysis and semantic annotation algorithms (e.g., for crowd detection) running on the Ground Station scale linearly with the number of input video streams/UAVs. Thus, a larger fleet would require a higher-performance Ground Station. Similarly, in terms of communication, increasing the number of UAVs implies updating the configuration and hardware of the LTE modules. Of course, the required bandwidth should be manageable by the communication base station. The LTE data rate (currently 50 Mbps) poses the hardest limit to the number of video streams (2 per UAV) it can accommodate. The only scalable solutions are to more aggressively compress the video streams, or to reduce the number of simultaneous live video streams from the fleet to the Ground Station. Finally, the Supervision Station has been designed to be ergonomic for an operator handling three UAVs, as more vehicles may create too much mental overload. In case of larger fleet sizes, several Supervision Stations would have to be set-up to control the UAVs in groups of three.

### 9.2 Safety

Safety is a very relevant issue in the proposed architecture, since it is essential to comply with UAV regulations for flying in civil airspace. This issue is addressed in both the preproduction and the production stage:

- Through the Supervision Station, a flight Supervisor is in charge of: a) double-checking mission security (in preproduction), and b) monitoring the entire execution, so as to take care of possible emergencies that may require canceling the mission (during actual production). Certain emergencies may also be detected automatically during execution by the Supervision Station, which alerts the Supervisor.
- The Supervisor can annotate information on a semantic map before production through the Supervision Station. She/he can indicate no-fly zones, i.e., primarily urban areas where UAVs are not allowed to fly, and emergency landing spots. The former are used by the flight planning modules so that UAVs never navigate through them. The latter are used by UAVs to land safely during emergencies (e.g., running out of battery or hardware failure).
- During production, the Visual Semantic Annotator and the Semantic Map Manager, acting in synergy, can an-

notate no-fly zones automatically. These annotations are extracted by processing the video streams from the UAVs and detecting crowded areas, since UAVs should not fly over people. This information is shown on the Supervision Station to evaluate risks, while the planning modules take such no-fly zones into account when replanning.

- The UAV Localization module enhances the system with some redundancy for safety. RTK-GPS is used as the main source for UAV localization, but in GPS-denied environments, the vehicles can also localize themselves temporarily by means of their on-board LiDAR and cameras.
- During mission execution, the Action Executer module provides functionalities for UAV collision avoidance. In particular, each UAV uses its on-board sensors and information sent by other UAVs (neighbors can share their position through the Communication module) to perform safe navigation by avoiding obstacles.
- Communication losses are a major threat for UAV safety. Therefore, a redundant link (LTE plus WiFi) is foreseen. The main communication channel between each UAV and the Ground Station is through LTE. However, a WiFi mesh connects all UAVs to each other, for facilitating communications if necessary.

## 10 Architecture Evaluation

The presented MULTIDRONE architecture was evaluated as an integrated robotic system for media production in realistic mock-ups of sports events, using experimental UAV platforms assembled for the project according to the specifications of Section 7. Additionally, a subjective video quality study was conducted in simulation, in order to define the best filming parameters when using the proposed architecture, in cinematographic terms. This set of evaluation sessions is briefly described below.

### 10.1 Subjective Parameter Study for UAV Cinematography

This Section presents the results of a relevant subjective video quality study, using the components of the proposed architecture. The goal was to evaluate the adequacy of UAV and camera parameters for different aerial cinematography shots. Thus, human feedback was employed to determine the best values for certain UAV shot parameters when performing them autonomously.

During aerial video capture, UAV platform and camera/gimbal parameters, as well as the relative motion between camera and target, can have a major aesthetic impact and influence on visual quality. However, the relationship between various scenarios, shot types and UAV parameters has yet to be fully understood. This is particularly important for live events, such as sports or festivals, where there is only one chance to get it right. A series of experiments were therefore conducted in order to characterize the preferred UAV parameters (or their optimal operating envelopes) for specific scenarios and shot types through subjective evaluation. The results were helpful to understand the perceptual influence of these parameters in UAV cinematography, to constrain flight planning within acceptable limits in the Dashboard, to recommend optimal shot parameters and to design innovative shooting techniques and shot transitions.

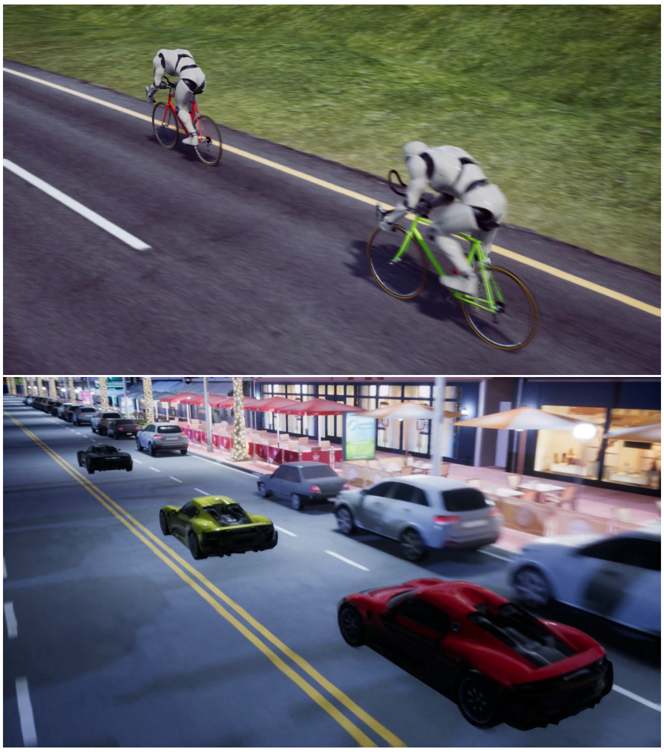

Fig. 7: Sample frames of cycling and car race scenarios created for the subjective study on UAV cinematography.

All test material used in the experiments was generated using the advanced real-time 3D graphics engine Unreal Engine 4, from Epic Games. As an alternative to acquiring real footage, this type of simulation provides a much lower cost solution to generate large amounts of data, with higher flexibility over the choice of environment, target(s), actions, and test scenarios. Camera and UAV parameters can also be carefully controlled and easily changed.

The subjective study was designed in two phases: an optimal UAV height test (Phase I) and a speed test (Phase II). In both phases, two different scenarios were generated: cycling and car races (see Figure 7). All the shot types, shown in Table 1, are based on the definition of conventional shots of cycling races and are included in the shot types set defined in [28, 31]. The video duration in the height test was five seconds, while in the Phase II speed test, the video duration varied from 3 seconds to 10 seconds, in order to cover similar video content for the same shot types. Note that, in many cases, the video duration is shorter than ten seconds, as recommended by ITU BT.500 [46] for subjective study. This is justified by a recent study [32, 33] on optimal sequence length for subjective video quality assessment.

Table 1: The tested shot types and UAV/camera parameters.

| No. | Object & Background | Shot Type | Tested Parameters | |
|---|---|---|---|---|
| | | | Height (m) | Relative Speed (m/s) |
| 1 | Cycling in CountrySide | DESCENT | 1, 2, 3, 4, 5 | 4, 5, 6, 7, 10 |
| 2 | | ORBIT | 1, 2, 3, 4, 5 | -2, -3, -4, -5, -6 |
| 3 | | SSMT | 1, 2, 3, 4, 5 | Static Camera |
| 4 | | FLYBY | 1, 2, 3, 4, 5 | 2, 2.5, 3, 4, 6 |
| 5 | | CHASE | 1, 2, 3, 4, 5 | 2, 2.5, 3, 4, 5 |
| 6 | Car in Night City | DESCENT | 2, 4, 6, 8, 10 | 8, 9, 11, 15, 20 |
| 7 | | ORBIT | 2, 4, 6, 8, 10 | -5, -6, -7, -9, -12 |
| 8 | | SSMT | 2, 4, 6, 8, 10 | Static Camera |
| 9 | | FLYBY | 2, 4, 6, 8, 10 | 4, 5, 6, 8, 11 |
| 10 | | CHASE | 2, 4, 6, 8, 10 | 1.5, 1.8, 2.1, 2.7, 3.8 |

A total of 39 subjects (20 for the height test and 19 for the speed test) participated in the experiments. Each trial consisted of the participant viewing a $3s$ mid-level gray screen, before viewing a randomly chosen sequence. The subjects were then asked to rate their viewing experience (not the video quality) from 5 to 1 (5=Excellent, 4=Good, 3=Fair, 2=Poor, and 1=Bad), following a single stimulus Absolute Category Rating (ACR) methodology. After the whole test session, each participant was informally interviewed about their viewing experience and scoring criteria. When all subjective data had been collected, mean opinion scores were calculated for each test sequence by taking the average opinion score from all the participants, along with the standard error.

Figure 8 shows two examples of the subjective results from the experiments, using Mean Opinion Scores (MOS). The error bar represents the standard error, and the red points stand for the MOS of test parameters which are significantly (through a paired t-test) lower than each best case. It can be observed that the optimal heights are close to $2m$ for the DESCENT shot in the Cycling scenario, while the relative UAV speed (to the objects) for the FLYBY shot in the Racing Car scenario is close to $5.3m/s$. The optimal parameters together with other fixed UAV/camera parameters, e.g., camera sensor size, focal length and horizontal distances between UAV and objects, were integrated within the Action Executer module in the architecture. Thus, they can be used in autonomous shooting by the proposed multiple-UAV architecture, enhancing the user experience.

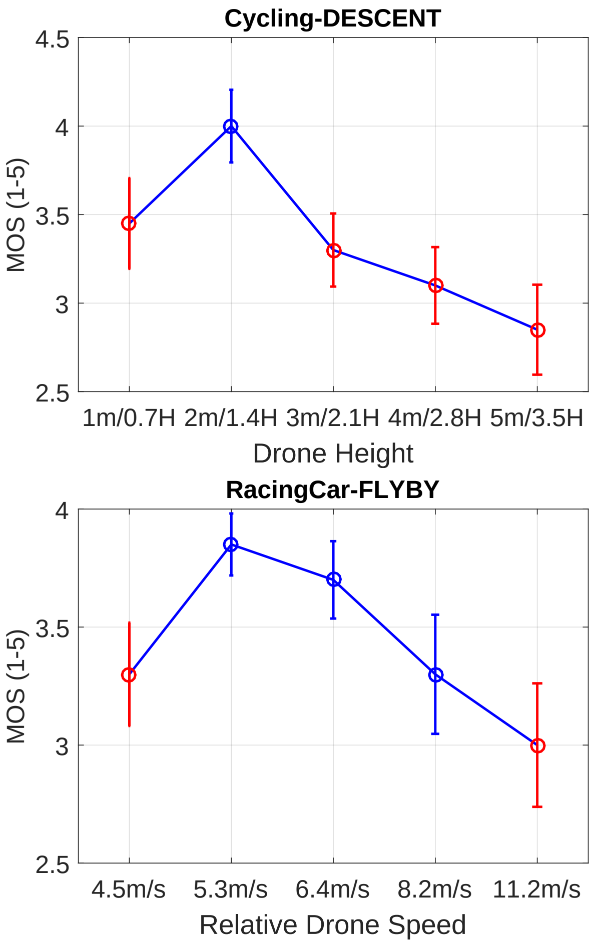

Fig. 8: Selected experimental results for the subjective study in UAV cinematography. (a) The MOS results for the DESCENT shot in the cycling scenario. (b) MOS results for the FLYBY shot in the car race scenario.

### 10.2 Empirical Evaluation for Outdoor Cinematography

Multiple integration sessions and field experiments were carried out during the MULTIDRONE project, in order to evaluate the entire architecture from both the hardware and the software point of view. The purpose of these experiments was to demonstrate the ability of our system to perform aerial media production for outdoor activities, with an actual team of filming UAVs. The evaluation of the specific, individual algorithms for cinematography mission planning and execution has been published in previous papers [1, 2, 7]. Here, a summary with a number of example experiments is included, to demonstrate the integration of the complete architecture.

The functionalities of the architecture concerning outdoor cinematography were evaluated on a set of relevant sports filming use cases. Thus, mock-up scenarios for cycling, rowing boat races and parkour activities were built and used for planning and executing cinematography missions. Summing up integration efforts by the project consortium, up to 9 weeks were devoted to physical integration throughout the last project's year, as well as 4 weeks for the field tests, including more than 40 hours of flight, split into two different campaigns in Germany and Spain. In Germany, a farm facility in Bothkamp with permits for amateur drone flight, located next to a lake, was used for cycling, rowing and parkour filming. In Spain, another outdoor site in a farm 30 $km$ away from Seville was used to emulate cycling events. All the experiments were performed with two or three UAVs in the team, and were supported by safety pilots, media experts for mission design and amateur sportsmen for the outdoor activities.

The hardware used in this empirical evaluation is as described in Section 7. Finally, only one DJI S1000+ was available, the first prototype. The rest of platforms in the UAV fleet were based on a custom frame designed by the company Hexadrone. Also, because of some issues between PX4 and Pixhawk 2.1, the final autopilot firmware was Ardupilot. Thanks to the use of the UAL component [45], this change was transparent for the rest of the software modules. The particular device used for RTK-GNSS was the Here+. The final software implementation of the proposed architecture for these evaluation experiments is available online upon request [2]. Relevant datasets are also available [3].

A parkour session was filmed in Germany, with a specific mock-up area with different obstacles. Parkour is a physical activity where runners move freely over and through any terrain using only the abilities of their body, principally through running, jumping and climbing. Figure 9 (top left) shows an example view taken from one of the UAVs during a parkour filming experiment. Figure 10 (top) depicts a scheme of an example mission. Runners moved in the *parkour zone* (green) from left to right and the Director designed a mission with 5 different shots. First, a sequence of a FLYTHROUGH shot followed by FLYBY (in blue), triggered by the `START_RACE` Event. Second, a sequence of a STATIC shot, a LATERAL and an ORBIT (in red), also triggered by the same `START_RACE` Event. This mission was planned with two UAVs, each performing one of the sequences. A complete video of the experiment is accessible at: `https://youtu.be/P_n_PfuEC2A`, and more details about this and additional missions can be found in [2]. For instance, rowing missions were also shot in Germany, in a lake where four amateur boats emulated a race for media production purposes. Figure 9 (bottom left) shows a picture of one of the UAVs taking a panoramic view of the race.

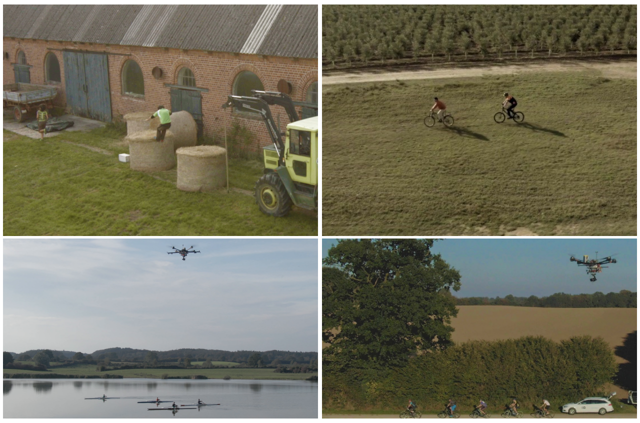

Fig. 9: Mock-up scenarios for actual field experiments. Top left, image taken from a UAV in a parkour mission. Top right, view of a cycling experiment in Seville. Bottom left, UAV filming a rowing race. Bottom right, view of a cycling experiment in Germany.

Cycling was filmed in Germany and in Spain, with amateur cyclists being tracked by the UAVs to take different types of shot. Figure 9 (top right and bottom right) depicts different views of the cycling experiments. Figure 10 (bottom) depicts a scheme of an example mission. There were two actual cyclist to be filmed, and the Director designed 3 sequences of shots in parallel, with the same starting time and duration (70 seconds) per sequence. The first sequence only included one LATERAL shot. The second sequence had two consecutive shots, a DESCENT (UAV approaches the target from behind coming closer in distance and height) and a CHASE. The third sequence had also two consecutive shots, a FLYBY (UAV starts behind the target at a lateral distance but it catches up with it to overcome it) and a STATIC (UAV stays still filming a panoramic view). Figure 10 (bottom) shows the cyclists trajectory, the shots designed by the Director (left view) and the plan executed by each of the UAVs (right view). The mission was planned with three UAVs and a sequence was assigned to each of them. A complete video of the experiment is accessible at: `https://youtu.be/nRM-TJ2njtg`. More details about this and similar cycling missions can be found in [7].

Finally, the system was demonstrated in a real regatta event in Wannsee, Berlin (Germany). A strategic spot was selected to deploy the system before the actual race. Then, two UAVs executed an autonomous preplanned mission as the rowers passed by, including STATIC and FLYBY shots. The main objective was to showcase to media end-users the possibilities of the proposed architecture and its fast deployment for covering a real sports event. Further details about the evaluation of MULTIDRONE mission planning and execution components are reported in [2, 7]. Moreover, a feature

[2] `https://aiia.csd.auth.gr/LAB_PROJECTS/MULTIDRONE/AUTH_MULTIDRONE_Software.html`

[3] `http://poseidon.csd.auth.gr/LAB_PROJECTS/MULTIDRONE/AUTH_MULTIDRONE_Dataset.html`

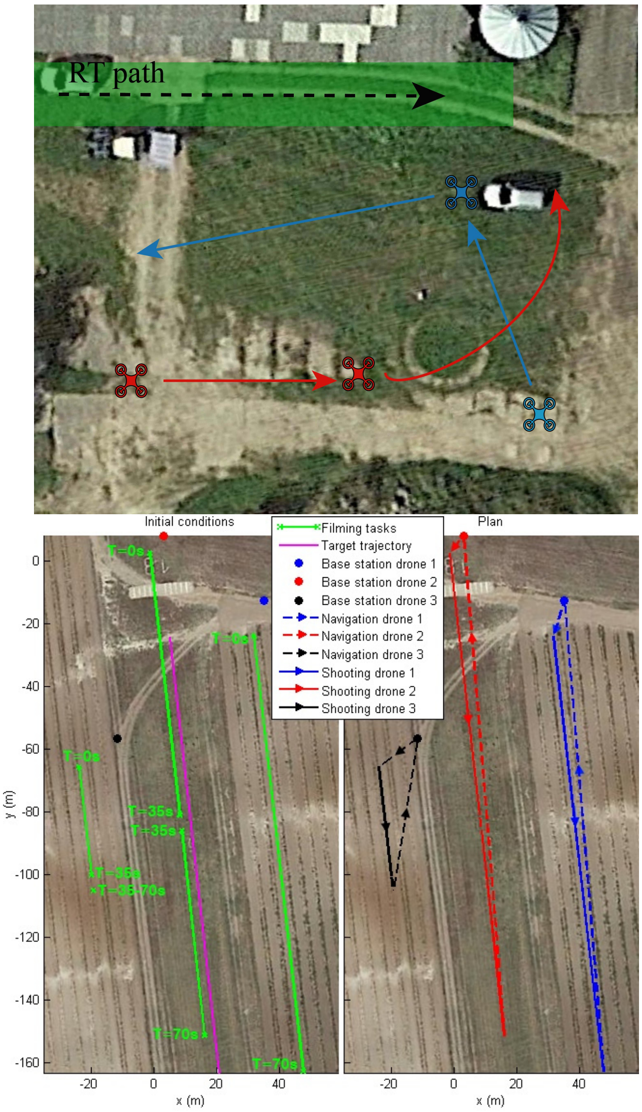

Fig. 10: Example missions in sport media production. Top, scheme of a mission with two UAVs and five shots in a parkour scenario. Bottom, scheme of a mission with three UAVs and five shots in a cycling scenario.

video about the field campaigns is accessible at `https://www.youtube.com/watch?v=iLs6Xo87j78`.

## 11 Conclusions and Future Prospects

A novel, fully integrated multiple-UAV software/hardware architecture for media production in outdoor settings has been devised. The system has been carefully designed for facilitating adaptive mission planning and control, as well as enhanced cognitive autonomy. It includes effective visual analysis modules, human-computer interfaces and suitable communication infrastructure for ensuring proper operation, platform scalability and increased system safety. The integrated system has been evaluated on mock-up and real sports event filming scenarios, while a related subjective study on UAV cinematography has also been performed by exploiting components of the proposed system.

The presented system architecture more than suffices to handle the problems posed in Section 3, since:

- it allows the Director's team to formally describe a cinematographic plan in the form of a machine-readable Shooting Mission,
- it translates the cinematographic plan into an overall, high-level mission plan for the entire fleet, that is subsequently optimally converted to lower-level individual UAV missions in an automated manner, respecting energy/battery limitations,
- it automatically executes the individual UAV missions, properly adjusting filming on-the-fly in order to ensure acquisition of the desired cinematographic shots, while concurrently obeying to safety constraints,
- it autonomously replans the overall mission on-the-fly when necessary, taking into account safety considerations, unforeseen events or possible emergencies,
- it permits optional, informed human intervention during mission execution, either for safety or for artistic reasons.

Media production using multiple, cooperating, autonomous UAVs is expected to greatly enhance the cinematic potential in outdoor live coverage. The proposed system constitutes a significant step towards this direction, in a manner that allows the Director's team to focus on the creative side of filming an event, rather than the technical details. Additionally, the system may be easily adapted to less challenging scenarios, such as filming scripted sequences (e.g., movie production).

Future technology improvements can be integrated in a straightforward manner into the system design, so as to increase the achieved levels of cognitive autonomy and platform safety. This is a main avenue for further research in the area. For instance, enhanced computational capabilities may allow richer semantic scene mapping in real time and, thus, increased safety. Moreover, higher sensor accuracy and lower communication latency may enable more precise target and UAV localization, while improved battery technologies are expected to increase UAV flight time and payload, in turn facilitating the use of better on-board sensors and allowing more complex cinematography planning.

Finally, alternative algorithms may be plugged-in into the system, replacing the current ones to potentially increase performance. For instance, decentralized multiple-UAV coordination methods could be employed instead of the priority-based trajectory planning now used. Also, once the available level of embedded computational power allows it, the visual human crowd detection and target detection/tracking neural networks could be combined into a single neural module, performing video instance segmentation on-board each UAV. Thus, overall, the proposed architecture opens wide opportunities to additional research, by permitting easy integration of new algorithms and technologies in the near future.

## Acknowledgments

The research leading to these results has received funding from the European Union's European Union Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE). This publication reflects only the author's views. The European Union is not liable for any use that may be made of the information contained therein.

The authors would like to thank Prof. Anibal Ollero (Head of GRVC, University of Seville) for his integral scientific contribution to the MULTIDRONE project and architecture.

The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Alcantara, A., Capitan, J., Cunha, R., Ollero, A.: Optimal trajectory planning for cinematography with multiple unmanned aerial vehicles. Robotics and Autonomous Systems **140**, 103778 (2021). DOI https://doi.org/10.1016/j.robot.2021.103778
2. Alcantara, A., Capitan, J., Torres-Gonzalez, A., Cunha, R., Ollero, A.: Autonomous Execution of Cinematographic Shots with Multiple Drones. IEEE Access pp. 201300–201316 (2020). DOI 10.1109/ACCESS.2020.3036239. URL https://ieeexplore.ieee.org/document/9249238/
3. Arev, I., Park, H.S., Sheikh, Y., Hodgins, J.K., Shamir, A.: Automatic editing of footage from multiple social cameras. ACM Transactions on Graphics **4**(33), 81:1–81:11 (2014)
4. Bonatti, R., Ho, C., Wang, W., Choudhury, S., Scherer, S.: Towards a robust aerial cinematography platform: Localizing and tracking moving targets in unstructured environments. In: International Conference on Intelligent Robots and Systems (IROS), pp. 229–236 (2019). DOI 10.1109/IROS40897.2019.8968163
5. Bonatti, R., Wang, W., Ho, C., Ahuja, A., Gschwindt, M., Camci, E., Kayacan, E., Choudhury, S., Scherer, S.: Autonomous aerial cinematography in unstructured environments with learned artistic decision-making. Journal of Field Robotics **37**(4), 606–641 (2020). DOI 10.1002/rob.21931
6. Bucker, A., Bonatti, R., Scherer, S.: Do you see what i see? coordinating multiple aerial cameras for robot cinematography. ArXiv e-prints (2020). URL https://arxiv.org/abs/2011.05437
7. Caraballo, L.E., Montes-Romero, A., Diaz-Bañez, J.M., Capitan, J., Torres-Gonzalez, A., Ollero, A.: Autonomous Planning for Multiple Aerial Cinematographers. In: International Conference on Intelligent Robots and Systems (IROS). Las Vegas, USA (2020)
8. Cunha, R., Malaca, M., Sampaio, V., Guerreiro, B., Nousi, P., Mademlis, I., Tefas, A., Pitas, I.: Gimbal Control for Vision-based Target Tracking. In: EUSIPCO. Satellite Workshop on Signal Processing, Computer Vision and Deep Learning for Autonomous Systems. La Coruña, Spain (2019)
9. Daniyal, F., Cavallaro, A.: Multi-camera scheduling for video production. In: European Conference on Visual Media Production (CVMP) (2011)
10. DJI: DJI products. https://www.dji.com/ (2020)
11. FreeSkies: FreeSkies CoPilot. https://freeskies.pr.co/ (2020)
12. Galvane, Q., Lino, C., Christie, M., Fleureau, J., Servant, F., Tariolle, F.l., Guillotel, P.: Directing cinematographic drones. ACM Trans. Graph. **37**(3) (2018). DOI 10.1145/3181975. URL https://doi.org/10.1145/3181975
13. Gandhi, V., Ronfard, R.: A computational framework for vertical video editing. In: Proceedings of the Eurographics Workshop on Intelligent Cinematography and Editing (WICED). Eurographics (2015)
14. Garage, A.: Skywand. https://skywand.com/ (2020)
15. Gebhardt, C., Hepp, B., Nägeli, T., Stevšić, S., Hilliges, O.: Airways: optimization-based planning of quadrotor trajectories according to high-level user goals. In: Proceedings of the Conference on Human Factors in Computing Systems (CHI). New York, USA (2016)
16. Gschwindt, M., Camci, E., Bonatti, R., Wang, W., Kayacan, E., Scherer, S.: Can a robot become a movie director? Learning artistic principles for aerial cinematography. In: IEEE IROS, pp. 1107–1114 (2019). DOI 10.1109/IROS40897.2019.8967592
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
18. Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C., Burgard, W.: OctoMap: An efficient probabilistic 3d mapping framework based on octrees. Autonomous Robots **34**(3), 189–206 (2013)
19. Huang, C., Yang, Z., Kong, Y., Chen, P., Yang, X., Cheng, K.T.: Learning to capture a film-look video with a camera drone. In: IEEE ICRA, pp. 1871–1877 (2019). DOI 10.1109/ICRA.2019.8793915

20. Joubert, N., E, J.L., Goldman, D.B., Berthouzoz, F., Roberts, M., Landay, J.A., Hanrahan, P.: Towards a drone cinematographer: guiding quadrotor cameras using visual composition principles. ArXiv e-prints (2016)
21. Kakaletsis, E., Mademlis, I., Nikolaidis, N., Pitas, I.: Bayesian fusion of multiview human crowd detections for autonomous UAV fleet safety. In: Proceedings of the European Signal Processing Conference (EUSIPCO). IEEE (2020)
22. Kakaletsis, E., Tzelepi, M., Kaplanoglou, P.I., Symeonidis, C., Nikolaidis, N., Tefas, A., Pitas, I.: Semantic map annotation through UAV video analysis using deep learning models in ROS. In: Proceedings of the International Conference on MultiMedia Modeling (MMM). Springer (2019)
23. Karakostas, I., Mademlis, I., Nikolaidis, N., Pitas, I.: UAV cinematography constraints imposed by visual target tracking. In: Proceedings of the IEEE International Conference on Image Processing (ICIP) (2018)
24. Karakostas, I., Mademlis, I., Nikolaidis, N., Pitas, I.: Shot type constraints in UAV cinematography for target tracking applications. Information Sciences **506**, 273–294 (2019)
25. Karakostas, I., Mademlis, I., Nikolaidis, N., Pitas, I.: Shot type feasibility in autonomous UAV cinematography. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2019)
26. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High-performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
27. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single-shot multibox detector. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer (2016)
28. Mademlis, I., Mygdalis, V., Nikolaidis, N., Montagnuolo, M., Negro, F., Messina, A., Pitas, I.: High-level multiple-UAV cinematography tools for covering outdoor events. IEEE Transactions on Broadcasting **65**(3), 627–635 (2019)
29. Mademlis, I., Mygdalis, V., Nikolaidis, N., Pitas, I.: Challenges in autonomous UAV cinematography: an overview. In: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME) (2018)
30. Mademlis, I., Nikolaidis, N., Tefas, A., Pitas, I., Wagner, T., Messina, A.: Autonomous unmanned aerial vehicles filming in dynamic unstructured outdoor environments. IEEE Signal Processing Magazine **36**, 147–153 (2018)
31. Mademlis, I., Nikolaidis, N., Tefas, A., Pitas, I., Wagner, T., Messina, A.: Autonomous UAV cinematography: A tutorial and a formalized shot type taxonomy. ACM Computing Surveys **52**(5), 105:1–105:33 (2019)
32. Mercer Moss, F., Wang, K., Zhang, F., Baddeley, R., Bull, D.R.: On the optimal presentation duration for subjective video quality assessment. IEEE Transactions on Circuits and Systems for Video Technology **26**(11), 1977–1987 (2016). DOI 10.1109/TCSVT.2015.2461971
33. Mercer Moss, F., Yeh, C.T., Zhang, F., Baddeley, R., Bull, D.R.: Support for reduced presentation durations in subjective video quality assessment. Signal Processing: Image Communication **48**, 38–49 (2016)
34. Messina, A., Metta, S., Montagnuolo, M., Negro, F., Mygdalis, V., Pitas, I., Capitán, J., Torres, A., Boyle, S., Bull, D.: The future of media production through multi-drones' eyes. In: International Broadcasting Convention (IBC) (2018)
35. Montes-Romero, A., Torres-González, A., Capitán, J., Montagnuolo, M., Metta, S., Negro, F., Messina, A., Ollero, A.: Director tools for autonomous media production with a team of drones. Applied Sciences **10**(4) (2020). DOI 10.3390/app10041494. URL `https://www.mdpi.com/2076-3417/10/4/1494`
36. MULTIDRONE-Consortium: D2.1: Multidrone media production requirements. https://multidrone.eu/wp-content/uploads/2017/01/MultiDrone_D2.1_Multidrone-media-production-requirements.pdf (2017)
37. Nägeli, T., Alonso-Mora, J., Domahidi, A., Rus, D., Hilliges, O.: Real-time motion planning for aerial videography with dynamic obstacle avoidance and viewpoint optimization. IEEE Robotics and Automation Letters **2**(3), 1696–1703 (2017)
38. Nägeli, T., Meier, L., Domahidi, A., Alonso-Mora, J., Hilliges, O.: Real-time planning for automated multi-view drone cinematography. ACM Transactions on Graphics **36**(4), 1–10 (2017). DOI 10.1145/3072959.3073712. URL `http://dl.acm.org/citation.cfm?doid=3072959.3073712`
39. Nousi, P., Mademlis, I., Karakostas, I., Tefas, A., Pitas, I.: Embedded UAV Real-Time Visual Object Detection and Tracking. In: Proceedings of the IEEE International Conference on Real-time Computing and Robotics (RCAR) (2019)
40. Nousi, P., Patsiouras, E., Tefas, A., Pitas, I.: Convolutional Neural Networks for visual information analysis with limited computing resources. In: Proceedings of the IEEE International Conference on Image Processing (ICIP) (2018)
41. Nousi, P., Triantafyllidou, D., Tefas, A., Pitas, I.: Re-identification framework for long-term visual object tracking based on object detection and classification. Signal Processing: Image Communication **88**, 115969 (2020)

42. Papaioannidis, C., Mademlis, I., Pitas, I.: Autonomous UAV safety by visual human crowd detection using multi-task deep neural networks. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2021)
43. Passalis, N., Tefas, A.: Deep reinforcement learning for controlling frontal person close-up shooting. Neurocomputing **335**, 37 – 47 (2019)
44. Patrona, F., Nousi, P., Mademlis, I., Tefas, A., Pitas, I.: Visual object detection for autonomous UAV cinematography. In: Proceedings of the Northern Lights Deep Learning Workshop (2020)
45. Real, F., Torres-González, A., Soria, P.R., Capitán, J., Ollero, A.: Unmanned aerial vehicle abstraction layer: An abstraction layer to operate unmanned aerial vehicles. International Journal of Advanced Robotic Systems **17**(4), 1–13 (2020). DOI 10.1177/1729881420925011
46. Recommendation ITU-R BT.500-11: Methodology for the subjective assessment of the quality of television pictures (2002)
47. Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
48. Saeed, A., Abdelkader, A., Khan, M., Neishaboori, A., Harras, K.A., Mohamed, A.: On realistic target coverage by autonomous drones. arXiv preprint arXiv:1702.03456 (2017). URL `http://arxiv.org/abs/1702.03456`
49. Skydio: Skydio products. https://www.skydio.com/ (2020)
50. Symeonidis, C., Mademlis, I., Nikolaidis, N., Pitas, I.: Improving neural Non-Maximum Suppression for object detection by exploiting interest-point detectors. In: Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP) (2019)
51. Tzelepi, M., Tefas, A.: Human crowd detection for drone flight safety using Convolutional Neural Networks. In: Proceedings of the EURASIP Signal Processing Conference (EUSIPCO). IEEE (2017)
52. Yuneec: Yuneec products. http://us.yuneec.com
53. Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., Hu, W.: Distractor-aware siamese networks for visual object tracking. In: Proceedings of the European Conference on Computer Vision (ECCV). Springer (2018)