A Multiple-UAV Software Architecture for Autonomous Media Production

Ioannis Mademlis[†], Arturo Torres-González[⋆], Jesús Capitán[⋆], Rita Cunha[‡], Bruno Guerreiro[‡$], Alberto Messina[§], Fulvio Negro[§], Cedric Le Barz[¶], Tiago Gonçalves[¶], Anastasios Tefas[†], Nikos Nikolaidis[†], Ioannis Pitas[†]

[†]Dept. of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece

[⋆]Robotics, Vision and Control Group, University of Seville, Seville, Spain

[‡]Institute for Systems and Robotics (ISR/LARSyS), Instituto Superior Tecnico, Lisbon, Portugal

[$]Dept. of Electrical and Computer Engineering, NOVA School of Science and Technology (FCT/UNL), Caparica, Portugal

[§]RAI - Centre for Research and Technological Innovation, Torino, Italy

[¶]Thales - Advanced studies department THERESIS, Palaiseau, France

***Abstract*—The use of UAVs in media production has taken off during the past few years, with increasingly more functions becoming automated. However, current solutions leave a lot to be desired with regard to autonomy and drone fleet support. This paper presents a novel, complete software architecture suited to an intelligent, multiple-UAV platform for media production/cinematography applications, covering outdoor events (e.g., sports) typically distributed over large expanses. Increased multiple drone decisional autonomy, so as to minimize production crew load, and improved multiple drone robustness/safety mechanisms (e.g., regarding communications, flight regulation compliance, crowd avoidance and emergency landing mechanisms) are supported.**

***Index Terms*—media production, UAV fleet, UAV cinematography, autonomous drones**

I. Introduction

[1]The rapid popularization of commercial, battery-powered, camera-equipped, Vertical Take-off and Landing (VTOL) Unmanned Aerial Vehicles (UAVs, or "drones") during the past five years, has already affected media production and coverage. They are expected to continue rising in popularity, for amateur and professional filmmaking alike. Single-UAV shooting with a manually controlled drone is the norm in media production today, with a director/cinematographer, a pilot and a cameraman typically required for professional filming. Initially, the director specifies the targets to be filmed, i.e., subjects or areas of interest within the scene. Then, he designs a cinematography plan in pre-production, composed of a temporally ordered sequence of target assignments, UAV/camera motion types relative to the current target (e.g., Orbit, Fly-By, etc.) and framing shot types (e.g., Close-Up, Medium Shot, etc.), which the pilot and the cameraman, acting in coordination, attempt subsequently to implement during shooting.

In such a setting, each target may only be captured from a specific viewpoint/angle and with a specific framing shot type at any given time instance, limiting the cinematographer's artistic palette. Moreover, there can only be a single target at each time, restricting the scene coverage and resulting in a more static, less immersive visual result. Finally, the "dead" time intervals required for the UAV to travel from one point to another, in order to shoot from a different angle, aim at a different target, or return to the recharging platform, impede smooth and unobstructed filming.

Fleets of multiple UAVs, composed of many cooperating drones with decisional and functional autonomy, are a viable option for overcoming the above limitations, by eliminating dead time intervals and maximizing scene coverage, since the participating drones may simultaneously view overlapping portions of space from different positions.

Although we are still far from realizing a fully autonomous platform, this paper proposes a UAV fleet approach for partially automating media production. This work is part of the MULTIDRONE European research project[2], attempting to produce an intelligent UAV fleet for media production. The system developed will be tested for filming sport events outdoors, such as freerunning, cycling or boat races. Following-up on preliminary relevant work focusing on specific areas and on surveying the domain ([1], [2], [3], [4]), the complete SW architecture is described in this paper.

II. Related Work

Current work on automating media production processes using autonomous UAVs is limited, making this a novel application for robotics. In general, the goal is to automate as many aspects as possible, while ensuring adherence to artistic and cinematographic constraints. Although a few low-hanging fruits have been grabbed, the general problem is still open and unsolved.

In commercial cinematography drones, only a few rudimentary functions are performed autonomously. Specifically, obstacle avoidance, landing, physical target following or target orbiting (for low-speed, manually pre-selected targets), as well as automatic central composition framing, i.e., continuously rotating the camera so as to always keep the pre-selected

[1]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).

[2]https://multidrone.eu/

target properly framed at the center, are the only available autonomous functions in state-of-the-art drones[3].

In research settings, many attempts consist in outputting feasible UAV trajectories that capture the intended visual content, possibly under cinematographic constraints ([5]). Additionally, in [6], the required number of drones in order to provide maximum scene coverage from appropriate viewpoints is computed. End-to-end systems able to autonomously execute single-UAV shooting missions have been developed as well, as in [7] and [8]. Such systems are capable of guiding a UAV outdoors so as to capture footage obeying to cinematographic rules, such as well-established visual composition principles and a list of canonical shots. The user implicitely specifies the UAV path and the shot types to be filmed before executing a drone mission by prescribing desired "key-frames", i.e., actual, temporally ordered example video frames of the intended shot, which are then subsequently captured autonomously during flight. The flight process is automated based on the cinematography plan, but no dynamic adaptation or active environment perception is involved. In [9], an on-line real-time planning algorithm is proposed that jointly optimizes feasible trajectories and control inputs for multiple UAVs filming a cluttered, dynamic, indoors scene with FoV/collision avoidance, by processing user-specified aesthetic objectives and high-level cinematography plans.

Several aspects of safely deploying autonomous UAV fleets for effective outdoor filming have not been explored in these works. For instance, the need to detect, localize and avoid human crowds, to adapt flight/shooting plans across large areas, to minimize the communication load and battery consumption, etc. These are the aspects emphasized by the SW architecture presented here.

III. Platform Objectives

The MULTIDRONE project aims at developing an innovative multi-drone audiovisual capture system targeting outdoor live media production, with novel contributions in the areas of a) decisional autonomy, robustness, safety and b) active perception and audiovisual shooting.

Within the first domain, MULTIDRONE aims at providing versatile planning and replanning capabilities that allow for coverage of large-scale events (both in time and space), a resourceful interface for interaction with the human operators (e.g., the Director), augmented decisional and cognitive system autonomy, improved safety functionalities such as autonomous emergency landing and autonomous vision-based crowd detection, etc.

Within the second domain, MULTIDRONE aims at developing geometric and semantic mapping functionalities that allow for defining different safety annotations (such as flight corridors, no-flight zones, or landing sites), multi-drone vision-based and GNSS-based target localization and tracking capabilities for tracking people (e.g., traceurs), crowds (e.g., viewers) or objects (e.g., boats, bicycles), multi-drone flight formation and camera controllers complying with cinematographic rules, multi-drone human-centered visual information analysis, etc.

[3]E.g., the popular DJI Phantom IV Pro

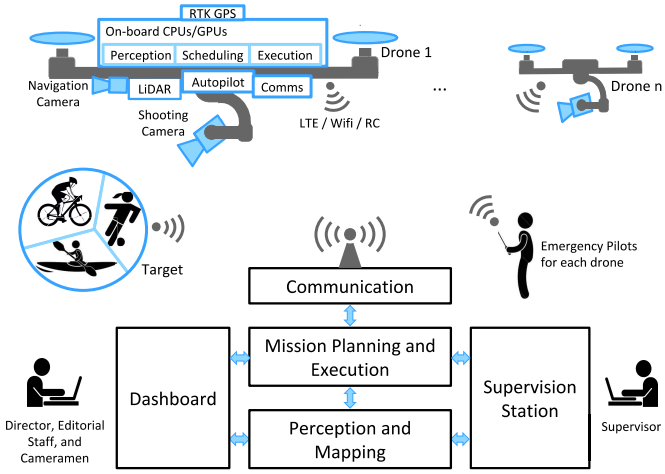

Fig. 1: Overview of the proposed multi-UAV architecture.

IV. Platform Overview

The MULTIDRONE system can be divided into "on-ground" and "on-drone" components, as depicted in Figure 1. The ground infrastructure comprises four main modules: Dashboard, Supervision Station, Mission Planning and Execution, and Communication. There are also modules for Perception and Mapping, with both on-ground and on-board components.

- *Dashboard*: it provides a GUI that enables the interaction between the Editorial Team (Director and other Editorial Staff) and the MULTIDRONE system, during both the pre-production and the production phases. The Dashboard will manage and display maps of the areas where the shooting will take place, annotated with relevant information such as no-flight zones, flight corridors, points-of-interest, landing sites, etc. During Pre-Production, the Director can create and manage the Shooting Mission, which consists in a list of Shooting Actions triggered by events. During Production, the Director uses the Dashboard to i) control the execution of the mission, by triggering events that start or stop the execution of Shooting Actions; ii) graphically monitor the execution of the Mission through the map display and the video streams from the drones' A/V cameras; and iii) introduce changes to the Shooting Mission or to specific Shooting Actions. The Dashboard will also allow for manual control of the cameras and respective gimbals.
- *Supervision Station*: it allows for a human operator, called the Supervisor, to supervise the execution of the Mission in terms of safety and security. The Supervision Station will include a GUI that displays the annotated map, video streams from the drone's navigation cameras, telemetry and status information from the drones. Through this GUI the operator can i) check and validate the safety of the

flight plan that originates from the Shooting Mission, when it is created and when changes are introduced to it; ii) monitor the mission execution, including the overall state of the drones; iii) abort the mission for security reasons; and iv) insert manually safety- and logistics-related annotations in a semantic map.

- *Mission Planning and Execution*: it comprises a collection of sub-modules that manage the planning and execution of a mission, providing the interface between the ground station and the drones. During pre-production, this module receives the Shooting Mission from the Dashboard, generates a plan for the available drones, asks the Supervisor for a security check, and sends to each drone their part of the plan. During production, it monitors the execution of the current plan by continuously receiving information from each drone. Using this, it generates and sends to each drone events that trigger the execution of Shooting Actions. As the execution progresses, replanning of the Mission may also be performed, in case of deviations between planning and execution and in case the Director introduces changes to the Shooting Mission (e.g., new Shooting Actions or modification of previous ones).
- *Perception and Mapping*: it comprises a collection of "on-ground" and "on-drone" modules that are responsible for several functionalities. These include:

  Geometric mapping: during pre-production, an exploration mission is done to generate a map using the LiDAR on-board the drones. This global geometric map is used for localization during mission execution.

  Semantic mapping: a semantic map containing geo-referenced annotations in the form of polygons or points and the corresponding labels (representing safety/logistics annotations such as landing/take-off spots and no-flight zones, as well as media-related annotations such as points of interest, etc.), is created manually during pre-production through the Supervision Station. New annotations can be added to the map during production either manually through the Supervision Station or automatically through the detection of new features on the images acquired by the drones, e.g., crowd detections and the Visual Semantic Annotator module.

  Drone localization: a localization module is implemented on-board each drone, which estimates the drone pose fusing data from the GNSS sensor (Global Satellite Navigation System), LiDAR, 2D visual analysis, and the geometric map.

  2D visual information analysis and Visual shot analysis: these modules are responsible for detecting and tracking targets of interest inside frames of the shooting video stream and providing visual control errors to be used for camera and gimbal control.

  3D target tracking (2D/3D translator and global 3D tracker): the detected 2D positions of a target inside image frames of different cameras are fused together with additional 3D measurements from other sensors (e.g., a GPS on board the target) in order to compute 3D estimations of target positions. These are used throughout the system for mission planning and execution, in addition to displaying the target's location.
- *Communication*: it consists of the modules in charge of implementing all the communication required by the system, by means of different communication links between the drones and the ground, namely LTE, WiFi or other radio links.

V. Software Architecture

As explained in Section IV, MULTIDRONE software is distributed both on-board the drones and on a Ground Station (GS). It runs on top of an Ubuntu Linux 16.04/ROS Kinetic Kame environment and depends on standard ROS message libraries, such as "std_msgs", "geometry_msgs" and "sensor_msgs", as well as on MULTIDRONE-specific messages/services for inter-module interactions. The design divides the SW modules depending on the functionality they are covering, so that each module provides a "simple" functionality and the objectives outlined in Section III are fulfilled.

A. On-Ground

In this section, the modules running on the Ground Station will be described. Some off-line tools that produce the semantic and geometric map, used by the run-time software modules, are also included.

1) Director's Dashboard: This module is a Web GUI allowing the Director's team to prescribe the cinematography plan in pre-production. Once the editorial instructions are finalised by the Director, they are translated into an XML document and sent to the Mission Controller for verification. Each relevant entity of the Director's Dashboard (e.g., Events, Missions, Shooting Actions) will be identified with a Universal Unique IDentifier (UUID), so that the planning components can associate low-level drone actions plans to editorial Shooting Actions they derive from.

During mission execution, current positions, poses, speeds etc. of all UAVs are depicted on the Dashboard. Mission configuration parameters may be updated on-the-fly by the Director's team and sent to the Mission Controller for re-validation, by sending XML fragments corresponding only to the modified entities. All entities will be linked through their UUID, which will have been sent beforehand, during the first validation step.

The Dashboard's functionalities corresponding to Director's events will be update / modification / deletion of an Event or a Mission, deletion of a mission, sending active Mission structure, stopping/aborting a Shooting Action inside a Shooting Action sequence, etc.

2) Supervision Station: The role of the Supervision Station is to reduce the workload of the Supervisor (the Supervision Station operator), allowing him/her to guarantee the good execution of multiple drone missions in terms of safety and security. It includes a GUI displaying all required information, that enables the Supervisor to have a clear overview of the situation: a) a map, where drones will be placed and on which

useful information will be overlayed, b) the video streams from the drone's navigation cameras, c) telemetry information (battery status, altitude above ground, vertical speed) and d) the drone action status, e.g., is the drone taking off, following a target, etc. The Supervision Station GUI is a thin JavaScript client, connected to a standard Web server. A Web RTSP proxy, integrated within the Web server as a service and connected to a standard RTSP server, is used for user interface display.

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, send the corresponding actions to each drone and monitors the fulfilment of the mission.

4) High-level Planner: This module computes a plan of a Shooting Mission. Once the Mission Controller receives the first Shooting Mission or decides that re-planification is needed, it will use the High-level Planner to compute the plan. This plan is later sent to each of the participating drones. It will receive the semantic map and some reference waypoints in the shooting mission in geodesic coordinates and transform them to the global Cartesian reference frame. It will also receive dynamic map annotations (crowd polygons) expressed in the global Cartesian reference frame.

5) Event Manager: This module centralizes the reception and generation of events which can trigger some actions. This module will generate an event in case of any of the drones reporting an emergency status. The Mission Controller will decide in that case whether a new plan is necessary. At the same time, the drone in emergency will execute an emergency maneuver. The rest of the system events will be related to the sport event being recorded and will trigger associated shooting actions. Each event will have an identifier and each shooting action will be triggered by the occurrence of an event with a specific identifier. Thus, the system can deal with an unbounded list of events working by means of their identifiers.

6) Global 3D Tracker: This module fuses the estimations of all targets' positions provided by the on-board visual detections and the GNSS attached to the target if available. It will be a stochastic filter that will produce an estimation of the pose of each target in the global coordinate system. It may receive targets' poses in geodesic coordinates from their on-board sensors, but it would transform them into metric before being integrated.

7) Visual Semantic Analyzer: Given a video frame sampled by the shooting camera of a specific drone at a specific moment in time, the Visual Semantic Analyzer will detect the occurrence of human crowds in the recorded scenes and will generate corresponding probability heatmaps. The input video frame is available at the ground station through a ROS message that is posted at a drone-specific ROS topic and extracted from the shooting video stream that is received on the ground.

8) Semantic Map Manager: The Semantic Map Manager provides two types of semantic annotations: 1) Static annotations in Keyhole Markup Language (KML) format that are computed before executing a mission. These are geolocalized features that augment the Geometric Map with semantic information. 2) Dynamic annotations that are derived during the execution of a mission, in the form of polygons.

The static annotations will refer to no-flight zones, geofencing limits, points of interest, landing zones, etc. Those will be originally posted to the Semantic Map Manager by the Supervision Station and the Mission Controller using geodesic coordinates in a KML format. The Semantic Map Manager will combine these annotations to create the semantic map that will be provided to interested modules (e.g., High-level Planner).

The dynamic map annotations will be generated by the Semantic Map Manager to specify areas with crowds (i.e., with ROS message data structure polygon). This information will be obtained by receiving prediction heatmaps from the Visual Semantic Analyzer and projecting that information onto the global inertial frame. The geometric map, gimbal status, camera status and drone pose will be used for this purpose.

9) Geometric Mapping: This module creates the global geometric map fusing and optimizing the maps generated by each drone. This optimization procedure is performed off-line and a priori (this is why the module does not appear in the general functional diagrams). It will be executed in pre-production as a standalone tool processing data previously logged by the drones. The exploration mission for mapping the scenario is made autonomously, given a certain area and probably some waypoints. It can also be done with manual flights, though this is not recommended for large scenarios. It will receive LiDAR data in the drone frame and drone poses to translate them into the global coordinate frame.

10) Video Streaming: Each drone sends 2 video streams: one from the A/V camera, the other one from the navigation camera. These two video streams are sent to the ground through the LTE network. Streams include H.264 RTP packets and RTCP packets. All the received streams are broadcast to an RTSP server, allowing clients to connect and select streams for display, and to a ROS node in charge of converting the H.264 RTP packets and RTCP packet to a ROS topic, where raw images are associated with their NTP timestamps. Each image-processing node will subscribe to this topic for video analysis.

B. On-Drone

This section describes the modules running on-board the drones.

1) On-board Scheduler: The On-board Scheduler receives the list of actions corresponding to the drone from the Mission Controller. Anytime the Mission Controller decides that re-planification is needed, it will compute a new plan and send new lists of actions to the drones involved. Then, the On-board Scheduler is in charge of executing them sequentially, via the Action Executer module, and monitoring the action status.

2) Action Executer: Once the On-board Scheduler receives the list of actions for the drone, it sends them sequentially to the Action Executer, which is responsible for the execution of these actions. For that, it will command the drone by means

of the interface called UAL, and the Gimbal and Camera by means of the Gimbal and Camera interfaces, respectively.

The final output of the Action Executer to command the drone movement will be a Velocity Tracking command to be issued by means of the UAV Abstraction Layer (UAL). A Drone Controller will compute those velocity commands depending on the shooting action parameters, the target position and velocity, and the drone position. For drone actions that involve a formation of drones, computation of the velocity commands will also depend on the position of the other drones, whose ID is provided in the drone action description, so that collision-free action execution is achieved.

In parallel, the Gimbal Controller computes the desired gimbal orientation such that the desired optical axis direction points towards the target, which requires knowledge of the target position and drone position contained in the drone pose messages. Alternatively, the desired gimbal orientation can be computed based on the visual control errors provided by the Visual Shot Analysis module that encodes the error between desired and current 2D positions of the target in the image frame.

3) UAV Abstraction Layer (UAL): The UAL ([10]) is the interface between the controller in the Action Executer and the autopilot. It receives velocity commands from the Action Executer and sends them to the autopilot. It also provides the pose and velocity of the drone in the global metric frame.

4) Drone Localization: This module is in charge of estimating the drone pose based on the on-board sensors available, namely GNSS positioning, LiDAR data, video streams from navigation and shooting cameras and the geometric map. The Drone Localization module will work both with geodesic coordinates and the global metric coordinate frame. The drone pose will be provided in the global metric frame and the geodesic coordinates received by the drone telemetry will be translated into this global frame to be integrated.

5) On-board 3D Target Tracker: This module estimates the 3D position of the target detected by the 2D tracking module. Basically, it will project 2D measurements on the image plane onto a 3D global system, by using camera pose. The module could exchange information with other instances on other drones to triangulate and get better 3D estimations. It will project the target positions on the image plane onto the global inertial frame.

6) 2D Visual Information Analysis: The 2D Visual Information Analysis module consists of a visual object detector and visual object tracker of the main actors (targets) of each scenario. It receives an uncompressed video frame from the shooting camera in real-time and generates 2D positions of the tracked targets as bounding boxes. Each 2D region of interest (ROI) on the image will contain attached the camera pose, the gimbal status and the drone pose at the time instance the corresponding image was taken, so that the 2D ROI can be later back-projected in 3D space. These data will be expressed in global metric coordinates. The module is initialized by a call to the Follow Target service by the Action Executer, which informs 2D Visual Information Analysis about the current target type and target ID.

7) Visual Shot Analysis: The Visual Shot Analysis module is initialized by the Set Framing Type service (called by the Action Executer), which sets cinematographic shot specifications (desired target position on frame, desired framing shot type). The module constantly receives the target 2D position from the 2D tracker and calculates the current visual control error, according to the desired shot specifications. This error is simply the deviation of the current ROI on-frame position from the desired one (in pixel coordinates), as well as the deviation of the current ROI on-frame area (as a percentage of the total video frame area) from the desired one. The error response can subsequently be used by the Action Executer to improve the quality of the shot, by controlling the gimbal orientation and camera parameters such as zoom.

VI. Conclusions

A novel, complete software architecture has been presented, that is suited to an innovative, intelligent, multiple-UAV platform for media production applications, covering outdoor events (e.g., sports) typically distributed over large expanses. Increased multiple drone decisional autonomy, as well as robustness and safety mechanisms (e.g., communication robustness/safety, embedded flight regulation compliance, enhanced crowd avoidance and emergency landing mechanisms) are supported. They are foreseen by a design that partitions functionality into processes executed on a ground station and others (more critical) that are executed on-board each drone.

References

[1] I. Mademlis, V. Mygdalis, C. Raptopoulou, N. Nikolaidis, N. Heise, T. Koch, J. Grunfeld, T. Wagner, A. Messina, F. Negro, et al., "Overview of drone cinematography for sports filming," in *European Conference on Visual Media Production (CVMP), short*, 2017.

[2] A. Torres-González, J. Capitán, R. Cunha, A. Ollero, and I. Mademlis, "A multidrone approach for autonomous cinematography planning," in *Iberian Robotics Conference (ROBOT)*, 2017.

[3] I. Mademlis, V. Mygdalis, N. Nikolaidis, and I. Pitas, "Challenges in Autonomous UAV Cinematography: An Overview," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2018.

[4] I. Mademlis, N. Nikolaidis, A. Tefas, I. Pitas, T. Wagner, and A. Messina, "Autonomous Unmanned Aerial Vehicles filming in dynamic unstructured outdoor environments," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 147–153, 2018.

[5] C. Gebhardt, B. Hepp, T. Nägeli, S. Stevšić, and O. Hilliges, "Airways: Optimization-based planning of quadrotor trajectories according to high-level user goals," in *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 2016.

[6] A. Saeed, A. Abdelkader, M. Khan, A. Neishaboori, K. A. Harras, and A. Mohamed, "On realistic target coverage by autonomous drones," *arXiv preprint arXiv:1702.03456*, 2017.

[7] N. Joubert, D. B. Goldman, F. Berthouzoz, M. Roberts, J. A. Landay, and P. Hanrahan, "Towards a drone cinematographer: Guiding quadrotor cameras using visual composition principles," *arXiv preprint arXiv:1610.01691*, 2016.

[8] Q. Galvane, J. Fleureau, F.-L. Tariolle, and P. Guillotel, "Automated cinematography with Unmanned Aerial Vehicles," in *Proceedings of the Workshop on Intelligent Camera Control, Cinematography and Editing (WICED)*, 2016.

[9] T. Nägeli, L. Meier, A. Domahidi, J. Alonso-Mora, and O. Hilliges, "Real-time planning for automated multi-view drone cinematography," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 132:1–132:10, 2017.

[10] F. Real, A. Torres-González, P. Ramón-Soria, J. Capitán, and A. Ollero, "UAL: An Abstraction Layer for Unmanned Aerial Vehicles," in *2nd International Symposium on Aerial Robotics*, 2018.