Materials Today: Proceedings 77 (2023) 848–854

Contents lists available at ScienceDirect

Materials Today: Proceedings

journal homepage: www.elsevier.com/locate/matpr

Development of an unmanned aerial vehicle for remote live streaming on web dashboard

Akshata Shendge [a], Rajendra Singh [b], Kashif I.B.H. Ansari [a,*], Kishita Pakhrani [c]

[a] Sinhgad College of Engineering, Pune, Maharashtra 411041, India
[b] IIT Palakkad, Kerala 678577, India
[c] MIT - World Peace University, Pune, Maharashtra 411014, India

ARTICLE INFO

Article history:
Available online 13 December 2022

Keywords:
ROS
Computer vision
OpenCV
Drones
Gazebo

ABSTRACT

As the prominence of autonomous aerial vehicles continues to rise globally, the ability of surveillance carried out remotely is likely to become a key aspect. In large spaces, random security checks are performed by security guards and are quite inefficient since on-foot monitoring is extremely non-productive and slow. This paper proposes a remote surveillance solution integrated into an aerial vehicle that performs random inspections and streams the live video on a web dashboard which is a faster and more reliable solution. The paper presents how to design and implement an Unmanned Aerial Vehicle (UAV) with surveillance capabilities wherein the live stream will be remotely accessible on any device connected to the same network. The main purpose of this drone is to transmit live data/video to a web dashboard and hence run inferences to analyse the data. A custom-made drone is integrated with Raspberry Pi with a Universal Serial Bus (USB) camera. An attempt to stream drone video feeds on local devices has been made in this field. The proposed framework operates in the robot operating system (ROS) integrated with the gazebo simulator and is designed to enable the software to communicate with the hardware. Two primary methodologies are utilised; simulations and experimental. The expected outcome should satisfy the need to view the drone's live video feed on a web dashboard for remote surveillance and computer vision application. At the resolution of 640×480, the latency in the live stream varied from 1 to 3 s with 10 frame rates per second. The analysis of the flight logs of the drone suggests that it meets the desired stability required for smooth video streaming. The total estimate calculated to implement this system is Rs. 70,000 including contingencies and spare parts required for experimentation. The proposed prototype appears to be a viable solution for security solutions required by multiple large space owners on the study's findings.

Selection and peer-review under responsibility of the scientific committee of the International Conference on "Innovations in Mechanical and Civil Engineering".

1. Introduction

Robotics in the words of an Oxford Dictionary is the amalgam of mechanics, electronics, and programming but truly it has an influence on all the branches of science. Over the years drones have become an intricate part of the robotics industry. The term drone is an abbreviation for Dynamically Operated Navigation Equipment which suggests that these unmanned aerial vehicles are capable of navigation and can be remotely operated. Under the category of multirotor, quadcopters are commonly used for development purposes.

Omnichen2 was the first quadcopter invented in 1920 by Etienne Omnichen [1]. It took 1000 successful flights and flew a recorded distance of 360 m. Nowadays, there is an incredible evolution in the 21st century in quadcopters. Drone technology has advanced and succeeded in recent years, from technically staffing sensitive military regions to enticing hobbyists worldwide. Hence to introduce more robust controllers, modelling, and techniques researchers are working continuously so that they can provide detailed and accurate representations of real-life quadrotors.

1.1. Problem statement

In large spaces, random security checks are performed by security guards to keep an eye on their spaces for any unauthorised

* Corresponding author.
E-mail address: kashif.ibha@gmail.com (K.I.B.H. Ansari).

https://doi.org/10.1016/j.matpr.2022.11.492
2214-7853/
Selection and peer-review under responsibility of the scientific committee of the International Conference on "Innovations in Mechanical and Civil Engineering".

access or trespassing, which makes it difficult for guards to patrol the entire space within their shift. Moreover, traditional security checks are quite inefficient since on-foot monitoring is extremely non-productive and slow. It is a very time-consuming process where security guards would have to keep walking up and down the areas and keep checking for suspicious activity. Security practices have not been able to keep pace with the evolution of crime and technologies. For example, a construction site can easily have an area of 40,000 square feet, which is too large to be constantly monitored by a human. It is estimated that to monitor such a large area, a security guard would need to walk at least 21 miles in a day. Hence, it has become mandatory to find a solution that will enable the efficient and quick patrolling or surveillance of large spaces.

2. Design

2.1. Quadcopter

Quadcopters are equipped with four motors and propellers to generate lift. In quadcopters, two motors fixed in the opposite direction rotate in a clockwise direction and the other two motors rotate in a counter clockwise direction to lift from the ground. Every fixed motor will generate the required amount of thrust and torque in rotational motion and such forces are used to move and hover the quadcopter properly. Coordination of these motors with respect to their direction of revolution ideally cancels all forces and maintains speed with the help of an electronic speed controller.

2.2. Robot operating system

The Robot Operating System (ROS) is an open-source ecosystem that uses a combination of libraries and tools to allow us to design complicated and reliable robot behaviour [4]. As a result, a primary program can be split into multiple subprocesses that run in parallel and communicate with one another. The rospack program in ROS allows you to query and analyse the code tree, search for dependencies, and discover packages by name, among other things. For convenience, rosbash is a collection of shell extensions that speeds up the system's command-line navigation.

The rospack program was created to allow developers to work on many ROS package repositories at the same time. The roots of local copies of ROS package repositories are defined by environment variables, and rospack explores the package trees as needed. The rosmake program supports recursive builds, which allow for cross-package library dependencies. ROS packages are open-ended in nature that allow a wide range of structures and purposes: some ROS packages encapsulate existing software, such as Player or OpenCV, automating builds and exporting functionality. Some packages provide nodes for usage in ROS graphs, while others offer libraries and standalone executables, as well as scripts for automating demos and tests. The packaging method is designed to break down the development of ROS-based software into tiny, manageable pieces, each of which may be maintained and developed independently by its own development team.

2.3. Embedded hardware and electronics

The selection of motors, propellers, electronic speed controllers and batteries vastly depends on the payload [5]. It is essential to take into account the compatibility and efficiency of these parts to ensure the stability of the structure. Direct Current (DC) motors are of two types: Brushless Direct Current (BLDC) and Brushed Direct Current motors. Generally, brushless DC motors are used in the case of a multirotor since BLDCs cause fewer vibrations as compared to brushed DC motors. The electronic speed controller (ESC) controls the speed of the motor and is either connected directly to the flight controller or power distribution board. Quadcopters require four propellers out of which two rotate in the clockwise direction and the other two in the anticlockwise direction. The propellers convert rotational motion into thrust and are represented in terms of the product of diameter and pitch for e.g.: $10 * 4.7$, $10 * 4.5$.

The most commonly used propulsion type in multirotor are Lithium Polymer (LiPo) batteries. This battery type has high capacity and is light in weight. The autopilot is the entity that controls the multirotor and can be interfaced with a companion computer used to extend the capabilities of the drone. The drone's power distribution board provides a maximum of 16.8 V and thus this buck converter is used to step down the voltage to supply the companion computer which operates on 5 V 3A. The required value of output voltage should be adjusted using a voltmeter before soldering it to the power distribution board. A lightweight camera was mounted for testing the live stream on the web dashboard. The complex robotic system was developed on the Holybro S500 chassis. The specifications of all key components are specified in Table 1.

3. Software

3.1. Base environment modification

The following technologies were used in the development of this drone:

- Robot Operating System - Set of software libraries, packages, and tools used for building robot applications
- Python - Python language was used to design ROS packages
- QGroundControl- A software used to communicate and control the drone

To create simulation environment of proposed system the base development/simulation versions of the environment tools were exploited:

- Ubuntu: 18.04
- Recent version of QGroundControl
- ROS: Melodic
- Version 8 of the Gazebo simulator, together with the px4 SITL (software in the loop) gazebo plugin

3.2. Autonomous mission planning

Flight Modes are of two types: manual and autonomous. Manual modes are where the user has control over the vehicle movement using the Remote Control (RC) control sticks (or joystick), while autonomous modes are fully controlled by the autopilot or

Table 1
List of components used in the quadcopter and their specifications.

Name of the components	Specification
Motors	2216 KV880
Electronic Speed Controller	BLHeli S ESC 20A
Battery	4S, 14.8 V, 5200mAh
Propeller	1045
Flight Controller	Pixhawk PX4
Companion Computer	Raspberry Pi 4
Camera	20MP | 30 fps
Buck Converter	DC-DC 3A Adjustable Buck module

the onboard companion computer and require no pilot/remote control input.

Steps for planning a mission in QGroundControl:

- Switch to the plan view
- Find the add waypoint 'plus' icon on the top left
- Click on the map to add waypoints
- Set various parameters as per desired values
- Once the mission plan is complete click on upload to send it to the vehicle

Once the mission has been uploaded, select fly view. The mission is displayed such that the progress of the mission can be tracked. On the left, if the pre-arm check is successful, the vehicle status will appear as ready to fly and the mission can be started by sliding the confirm bar at the bottom.

3.3. ROS package

A ROS package consists of a catkin-compliant package.xml file and CMkaeLists.txt where the package.xml provides *meta*-information about the package. A ROS package can be created by using the catkin_create_pkg script after sourcing the workspace. This command requires the package name and optionally a list of dependencies. After creating the package, the directory can be built by using catkin_make or catkin build.

The following packages were installed:

- Web_video_server [8]
- USB_cam

The usb_cam_node interfaces with the USB camera and provides a rostopic which is subscribed by the web_video_server package creating an instance of video encoder.

The web video server [10] listens for Hypertext Transfer Protocol (HTTP) requests on a local port. When an HTTP request is made for a video stream of a ROS image subject.

3.4. Computer vision application

With the need to view the drone's live video stream remotely, a computer vision application was implemented to address one of the safety use cases, face detection. OpenCV was used to integrate computer vision applications. The ROS package cv_camera was installed to provide a drone video feed with OpenCV inference available on the web dashboard as a rostopic. Opencv_apps ROS package was used for the many features it provides like simply running a launch file that corresponds to the desired OpenCV feature. For testing purposes, a face detection [14] module was explored. While drones can be used for numerous applications some notable surveillance applications pertain to large space areas such as warehouses, construction, agriculture etc.

4. Results

4.1. Simulations

Before carrying out any real-time testing [9], it is crucial to test the program in a simulated environment and observe the behaviour of the drone. A gazebo simulator was used to effectively test various packages and algorithms discussed in this paper. The gazebo is a robust software and is compatible with both ROS and Pixhawk.

4.2. PX4-Avoidance: Local planner

While the drone is able to live-stream video feed it is also crucial to develop the obstacle avoidance ability of the drone to navigate autonomously in an outdoor environment. For testing this in a simulated environment the local planner algorithm [6] was used and visualised in a tool called RViz (ROS visualisation). The camera node acquires the data from the camera and the MAVROS (MAVLink extendable communication node for ROS with proxy for Ground Control Station) node is the communication driver for autopilots. The vehicle state information is given by the MAVROS node. The algorithm calculates the values of the next set-points and communicates with the MAVROS node which then sends it to the autopilot. During the tests, the drone is in an off-board state and thus the autopilot directly uses these setpoints for navigation. In case of delay when receiving setpoints an error message is sent and the drone switches back to position mode. To avoid this, continuous setpoints are required to be sent hence two different threads are used to handle the algorithm and communication while keeping a track of time.

4.3. Real-time testing

Once all basic testing was carried out in the simulations, the system was tested in an open environment [7]. It is critical to follow standard safety protocols such as ensuring the arena has the least number of obstacles, high tension wires are not present, weather conditions are appropriate, etc. It was observed that the drone's live feed was available on the web dashboard. As the distance between the device and the drone increases, latency is observed in the live stream. With a resolution of 640*480, the drone was flown up to 30, 20, 15 and 10 m and it was observed that for clear visibility and computer vision applications a height below 10 m was optimum (see Figs. 1-9).

Fig. 1. Quadcopter and It's the motor configuration used in the experiment [2].

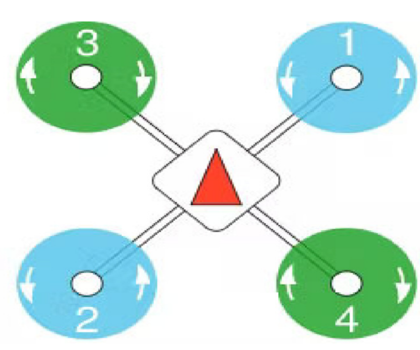


Fig. 2. Configuration of the direction of rotation for each quadcopter Motors [3].

(a) (b) (c)

Fig. 3. (a) Front view; (b) Side View; (c) Top View of the drone used.

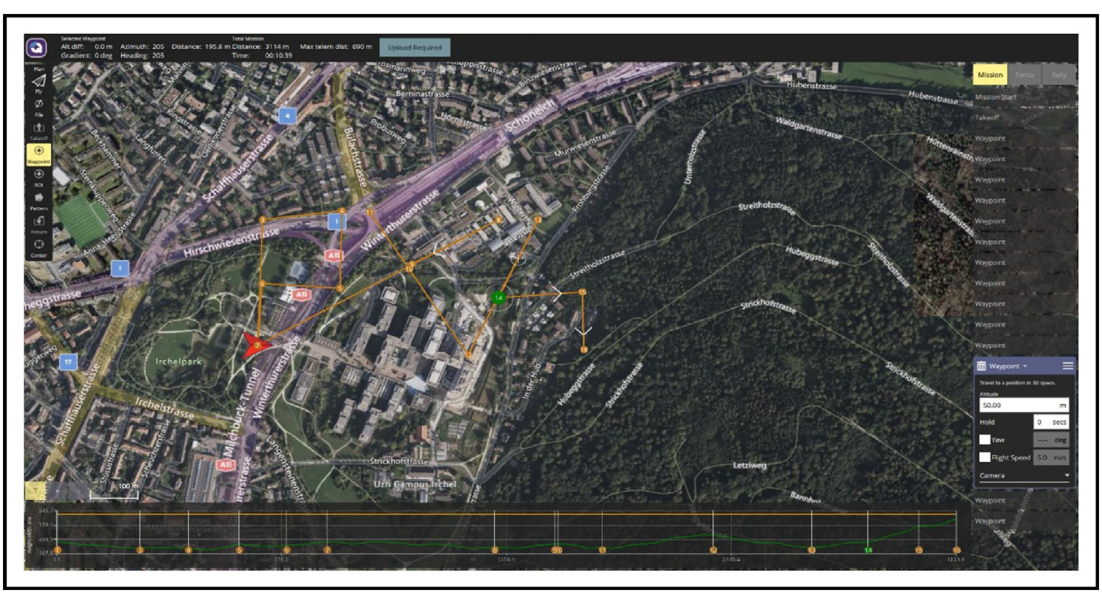


Fig. 4. GUI of QGroundControl for giving waypoint mission to the quadcopter.

5. Conclusion & future scope

A clear drawback of the current approach is the high latency (1–3 s with 10 frame rates per second) in the video feed and the limitation of the range of the drone flight. Moreover, the web dashboard is only accessible to devices connected to the same network (i.e., local network).

The proposed system does not address any security issues (as any device on a local network can access the drone video web server). In the future, the dashboard can be integrated with cloud services like Amazon Web Services (AWS) and a database to make the stream globally accessible and add functionalities such as storage or video feed and images.

Fig. 5. GUI for tracking the state of the quadcopter during the mission execution.

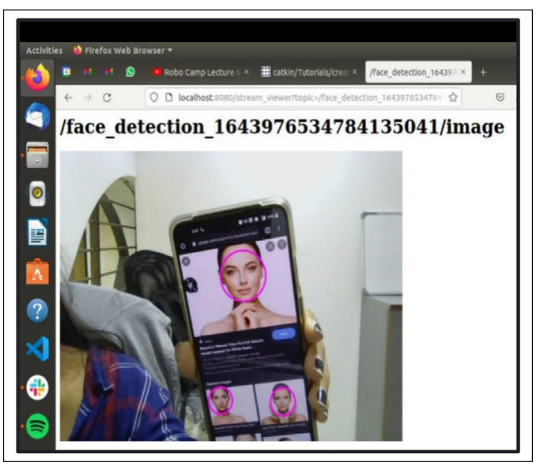


Fig. 6. Pink colour circle shows the detected face through the drone camera feed on the web dashboard. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

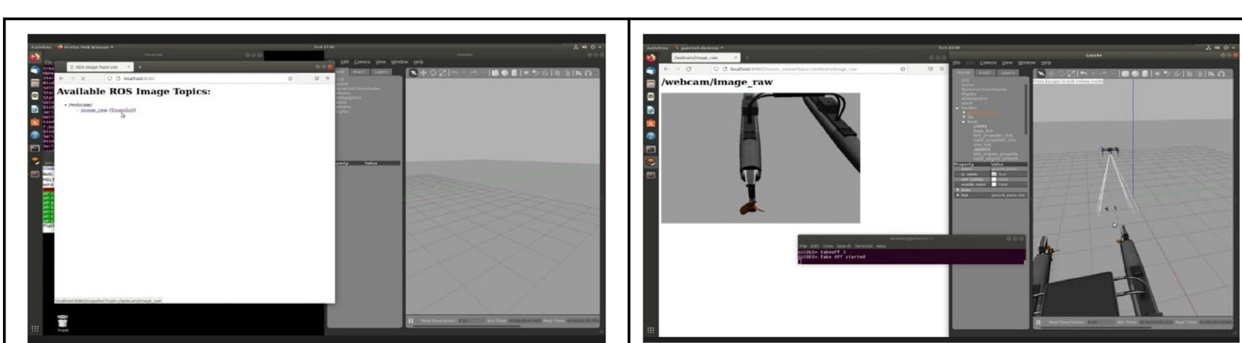
This paper proposes a UAV system that can be used for remote monitoring in large spaces where a web dashboard is used to view the video stream on a local network.

The emergence of open-source hardware and software has eased the development of robotics. The usage of ready-to-use packages of ROS and PX4 has helped in faster advancement [13].

The ROS package web_video_server was modified which helped improve the performance of the live stream (Lowered the latency with a 640*480 Resolution Stream). The design of the developer kit was customised and tuned using PID (Proportional-Integral-Derivative) tuning to achieve a stable flight and smooth stream.

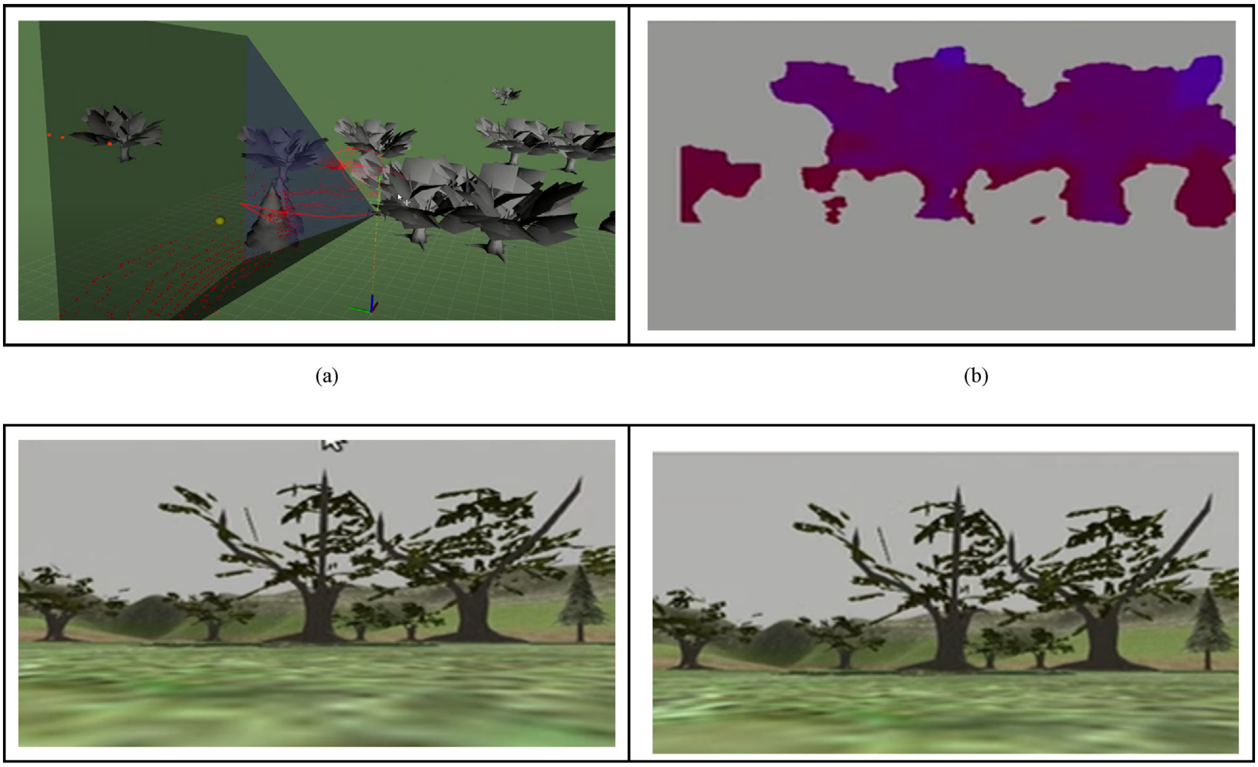
To summarise

- An aerial [12] vehicle along with the chosen ROS packages i.e., web_video_server and computer vision can be a simple solution towards surveillance [11] and security.
- Large spaces are extremely tough to monitor and administrative staff can make use of this technology for several use cases.
- Multiple applications of computer visions can be incorporated



(a) (b)

Fig. 7. (a) Available ROS Image Topics Message from Simulation; (b) Live video stream from the drone on web dashboard.



(a) (b) (c) (d)

Fig. 8. (a) Third Person View of Drone flying in Simulation (b) Disparity generated using Stereo Vision; (c) Drone's left camera view; (d) Drone's right camera view.

(a) (b)

Fig. 9. (a) Live video stream testing through quadcopter camera on the web server; (b) Third Person View of Quadcopter's Flight.

CRediT authorship contribution statement

Akshata Shendge: Conceptualization, Methodology, Software. **Rajendra Singh:** Data curation, Writing – original draft, Supervision. **Kashif I.B.H. Ansari:** Visualization, Investigation, Writing – review & editing. **Kishita Pakhrani:** Investigation, Software, Validation.

Data availability

Data will be made available on request.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A.M. Pudke, A.C. Govalkar, D.S. Lal, "Design and Development of Quadcopter with Paint Spray Mechanism", IRJET, May 2021, Volume: 08 Issue: 05, 6 Pages, Retrieved from https://www.irjet.net/archives/V8/i5/IRJET-V8I5852.pdf.

[2] AliExpress, Holybro Motor 2216 880KV , accessed on 2nd Jan, 2022 <https://www.aliexpress.com/item/4000307923780.html>.

[3] Airframes Reference, Last Updated: 22/4/2022, accessed on 2nd Jan, 2022. <https://docs.px4.io/master/en/airframes/airframe_reference.html>.

[4] P. Estefo, J. Simmonds, R. Robbes, J. Fabry, The Robot Operating System: Package reuse and community dynamics, Journal of Systems and Software 151 (2019), https://doi.org/10.1016/j.jss.2019.02.024.

[5] J.A. Benito, G. Glez-de-Rivera, J. Garrido, R. Ponticelli, Design considerations of a small UAV platform carrying medium payloads, Design of Circuits and Integrated Systems (2014) 1–6, https://doi.org/10.1109/DCIS.2014.7035583.

[6] Marín, Pablo & Hussein, Ahmed & Martín Gómez, David & de la Escalera, Arturo. Global and Local Path Planning Study in a ROS-Based Research Platform for Autonomous Vehicles. Journal of Advanced Transportation. (2018). 1-10. 10.1155/2018/6392697.

[7] V. Kangunde, R. Jamisola, E. Theophilus, A review on drones controlled in real-time, International Journal of Dynamics and Control. 9 (2021) 1–15, https://doi.org/10.1007/s40435-020-00737-5.

[8] ROS.org, web_video_server, accessed on 24th Nov, 2022 <https://wwwhttp://wiki.ros.org/web_video_server>.

[9] U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, E.T. Matson, Real-Time and Accurate Drone Detection in a Video with a Static Background, Sensors (Basel). 20 (14) (2020 Jul 10) 3856, https://doi.org/10.3390/s20143856. PMID: 32664365; PMCID: PMC7412503.

[10] C.-W. Chen, Drones as internet of video things front-end sensors: challenges and opportunities, Discover Internet of Things. 1 (2021), https://doi.org/10.1007/s43926-021-00014-7.

[11] N. Dilshad, J. Hwang, J. Song, N. Sung, Applications and Challenges in Video Surveillance via Drone: A Brief Survey, International Conference on Information and Communication Technology Convergence (ICTC) 2020 (2020) 728–732, https://doi.org/10.1109/ICTC49870.2020.9289536.

[12] U. Stankov, Đ. Vasiljević, V. Jovanović, M. Kranjac, M.D. Vujičić, C. Morar, L. Bucur, Shared Aerial Drone Videos — Prospects and Problems for Volunteered Geographic Information Research, Open Geosciences 11 (1) (2019) 462–470, https://doi.org/10.1515/geo-2019-0037.

[13] A. Abdelmaboud, The Internet of Drones: Requirements, Taxonomy, Recent Advances, and Challenges of Research Trends, Sensors 21 (2021) 5718, https://doi.org/10.3390/s21175718.

[14] Chenfan Sun, Wei Zhan, Jinhiu She, Yangyang Zhang, "Object Detection from the Video Taken by Drone via Convolutional Neural Networks", Volume 2020, https://doi.org/10.1155/2020/4013647.