Grenze International Journal of Engineering and Technology, Jan Issue

Network Quality Feedback of Drone Telemetry and Imagery Upload in the CICLOPE Wildfire Protection System

Joao F. Oliveira[1], Joel P. Almeida[2] and Luís Silva[3]
[1-3]INOV – Instituto de Engenharia de Sistemas e Computadores Inovação, Lisbon, Portugal
Email: {joao.oliveira, joel.almeida, luis.silva}@inov.pt

***Abstract*— Wildfires prevention/detection/fighting is a topic of increasing interest in recent years. Continuous network video streaming from remote surveillance towers to command centers has facilitated the early detection of wildfires. While the tower positions can be optimized with respect to optical and radio line of sight coverage using topography maps and raster simulations, network quality feedback becomes important for integrated mobile surveillance such as drones operating in remote areas and valleys. In this paper we present a visual network quality feedback solution that enables the drone operator to see immediately whether the images time-stamped with drone telemetry are being received correctly in the command center or not. This +/- icon in our custom smart controller application enables the operator to quickly become aware of network issues, adjust their position/router position and adapt the flight path accordingly when assessing fires.**

***Index Terms*— Ciclope, wildfire, network, drone, visualization and mobile.**

I. Introduction

Drone technology has been revolutionary in many applications such as precision farming, package delivery, offense/defense and wildfire characterization/fire validation. To help decision making when fighting wild fires, the drone imagery and telemetry typically needs to be integrated into a more comprehensive geographical information system (GIS) platform. Whilst the position and coverage of the static surveillance towers can be planned in advance, the network conditions of mobile assets can vary with their location in the terrain. For instance, when moving the drone smart controller too far away from the router. Knowing that the image and telemetry data of a drone are being received correctly or not in the command center becomes important for timely decisions. Simply relying visually on the available Wi-Fi network bars becomes inadequate in remote areas as it does not answer whether the images and data made it to the servers on time or at all. Pierce reports that with one Wi-Fi bar, packet delivery may be unreliable [1]. In this paper we present a network quality feedback solution that enables the drone operator to know immediately and clearly whether the drone data is being sent/received or not regardless of the number of Wi-Fi bars. Furthermore, it allows the connection to be seamlessly reestablished if physically possible respecting the Android lifecycle. In section 2 we briefly review Ciclope, an Integrated System for Remote Monitoring and Wildfire Detection. In section 3 we present Ciclope Drone the android application that controls the drone and provides the network quality feedback to the operator. In section 4 we present results and present conclusions in section 5.

Grenze ID: 01.GIJET.10.1.390

II. BACKGROUND

The initiative to create the Ciclope system originated in 1994 with investment from INOV (at the time INESC) a non-profit research institute based in Lisbon [2] in partnership with the Natural Park Peneda-Gerês (located in the north of Portugal). Later in 2004 private funding through COTEC further boosted critical infrastructure. Asset mapping of forest land and inventory mapping of traditional watch towers were carried out [3] and coverage optimization algorithms were developed (Catry et al.[4], Almeida et al.[5]). In addition a study for the field deployment of such monitoring system was carried out (Relvas et al.[6]). Since then there has been a continuous adaptation to new technologies and end user requests. For example, the transition from analogue to digital cameras, artificial intelligence for smoke column detection and ergonomic improvements in mobile tower assembly. In this section we briefly review some of the main features of the architecture, hardware and software in the respective sections.

A. Architecture

Ciclope is a monitoring system designed to work in any place with completely autonomous energy and autonomous communications system. It has a modular and scalable architecture, completely based on a IP platform (Fig. 1)

1. Unlimited number of Data Acquisition units: fixed towers, mobile terrestrial and aerial units.
2. Independent communication system: Radio P2P links (licensed or unlicensed) network, Fiber optics, public networks (3G/UMTS, ADSL, cable, …)
3. Client-Server based user operation (unlimited number of local or internet users, low bandwidth requirements allow remote mobile operation in PC or phone
4. Web access for control and maintenance

Currently the system has 139 surveillance towers, 24 control centers, 50+ remote monitoring centers, 2 drones. 44% coverage of wildland in Portugal (4 million hectares). It is one of the most reliable systems supporting decision making in forest protection.

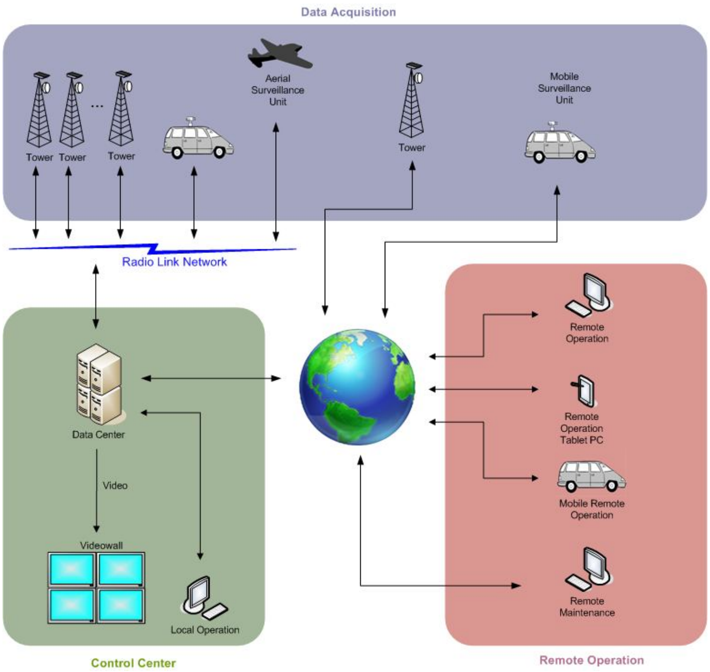

Figure 1. Ciclope - system architecture

B. Hardware

The surveillance towers either use grid energy if it is available or solar energy (Fig. 2) with a power manager system that charges the batteries and communicates their state. The cameras (visible spectrum or infrared) can be controlled real time manually in a multiuser environment with set privileges, or rotate 360º continuously (Fig. 3). Radio antennas transmit data (meteorological, video, images, etc) to management and control centers, with regional command centers (Fig. 4) and players in the fire combat operations having simultaneous access.

Figure 2. Remote data acquisition tower (autonomous energy and communications)

Figure 3. Motorized pan-tilt video camera

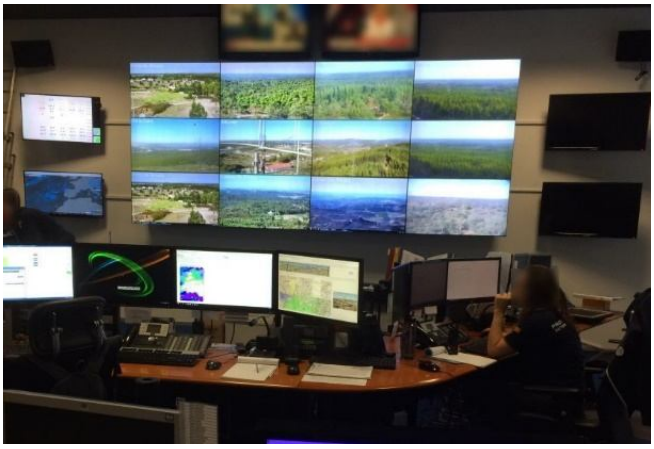

Figure 4. Command and control center with video wall

C. Software

The user interface allows different view modes. One of them provides a map area with several layers of information available: 1:25 000 cartographic maps, real time georeferenced assets, tower positions with respective adjustable pan-tilt-zoom frustums in blue (Fig.5), regional polyline limits and town names. Other types of georeferenced data can be added (e.g. hikers). In addition, a raster layer with precomputed radio coverage is assessable in light green (Fig. 6) and the current network quality status of the fixed towers radio segments can also be selected. Other views can visualize just video streams of selected cameras. The automatic fire/smoke column detection will bring up a window to the foreground and ask a user for validation. The alarm event is inserted in the shared database, regardless if a user replied (Fig. 7). Note how thermal imaging allows to see through the smoke, revealing where the fire is truly active (Fig. 8), this alarm was also automatically detected. Through the menus on the top, other functionality can be accessed, such as the hardware status of the whole system, the database of fire/smoke columns alarms and meteorological data.

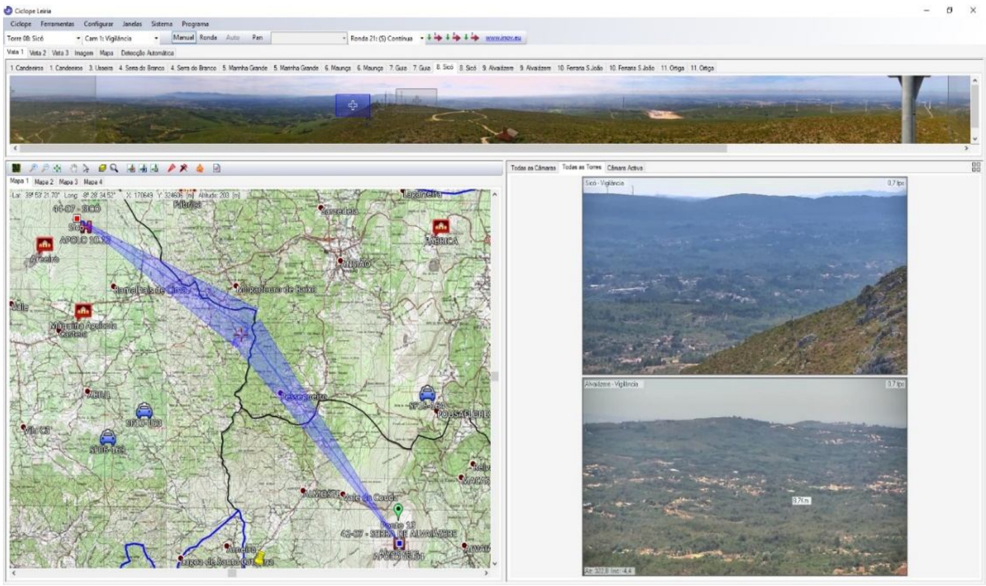

Figure 5. Ciclope decision support tool and system manager. 1:25 000 Map with georeferenced assets, tower positions with respective adjustable pan-tilt-zoom frustrums in blue and video streams, town names

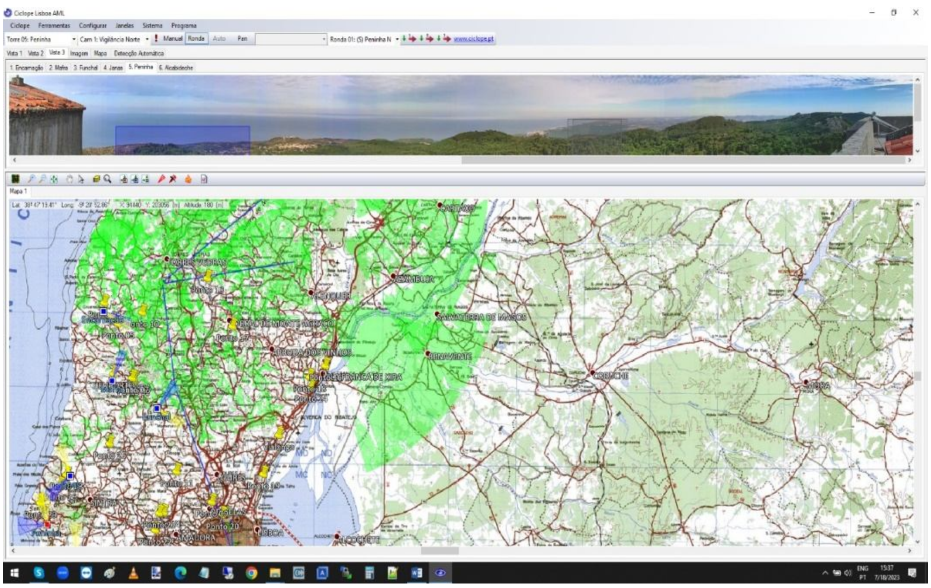

Figure 6. Ciclope fixed tower coverage and current radio network quality status visualization (blue segments: up; orange segment : unstable; red segment: down (not shown); grey: (status not known); grey-dash segments: optional segment)

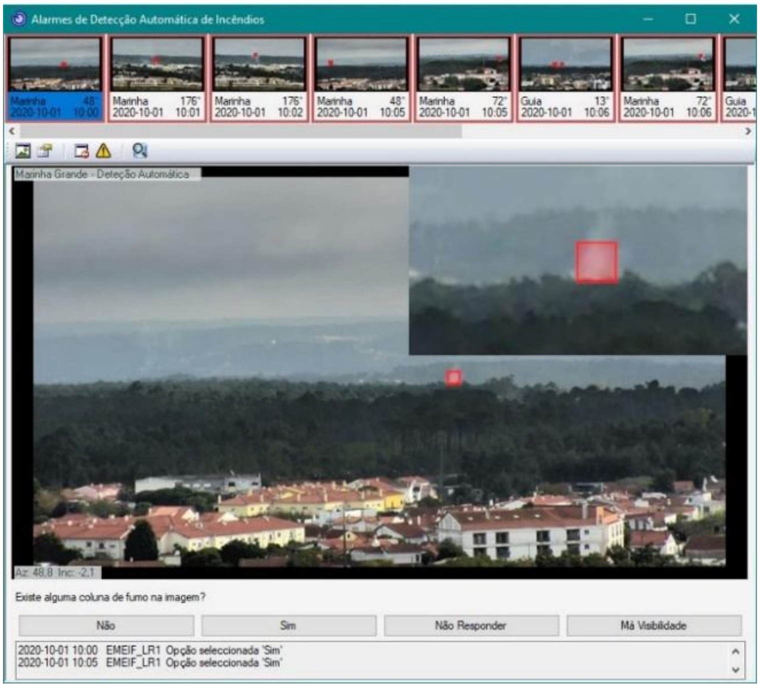

Figure 7. Automatic smoke column detection (visible sepctrum)

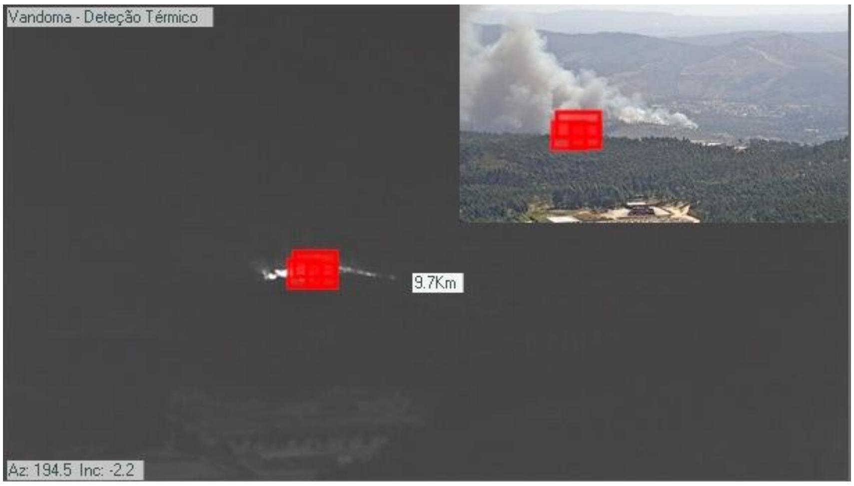

Figure 8. Automatic hot spot detection (infrared)

III. Ciclope Drone Application

Different drone swarm architectures are being studied in the context of wildfire detection/monitoring [7], for instance the combination of fixed wing drones/area coverage with quadcopters/fire validation can be most cost effective [8]. Our work focuses on more local operations such fire characterization/monitoring, studying terrain access for firefighters, network visualization and usability. This section describes the drone application and how the drone imagery and telemetry were integrated with the Ciclope decision support system. Specifically, in sub section A we describe the hardware and software platform used for development. While it is beyond the scope of this paper to describe the Android software lifecycle or how the DJI's SDKs work in detail, we provide an overview of the user interface in sub section B and the network Java exception that makes the visual feedback solution possible in sub section C.

A. Hardware and software platform

We use the DJI Mavic 2E Dual (Fig. 9) for its visible and thermal imaging capabilities and the rm500 smart controller for its ultra-bright display, HDMI resolution and processing power.

Figure 9. DJI Mavic 2E Dual and rm500 smart controller running the Android OS

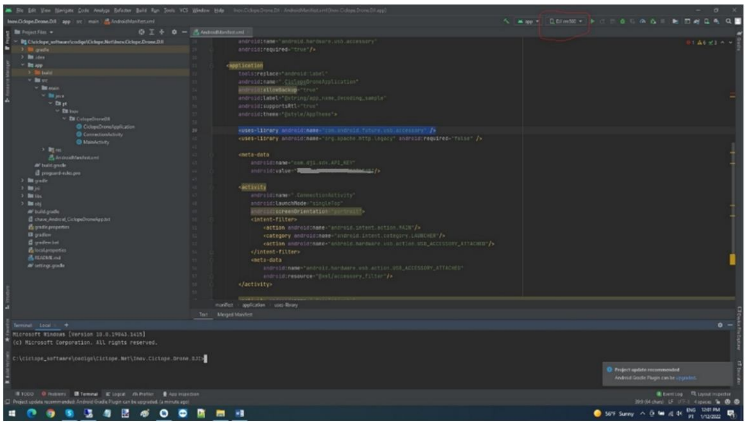

Figure 10. Android Studio Artic-Fox and the connected DJI rm500 smart controller

The Ciclope Drone App was written in Java, using Google's Android SDK, the Android Studio IDE (Fig. 10) and two DJI SDK (Mobile SDK, UX SDK 4.16). DJI makes available several SDK [9]:

- Mobile SDK (Android or iPhone) allows to access drone telemetry such as latitude and longitude
- UX SDK allows to use a library of user interface buttons for the smart controller
- ONBOARD SDK allows to create functionality internal to drone/Linux
- PAYLOAD SDK allows to create functionality for the drone accessories
- WINDOWS SDK used to create windows applications

B. User-interface

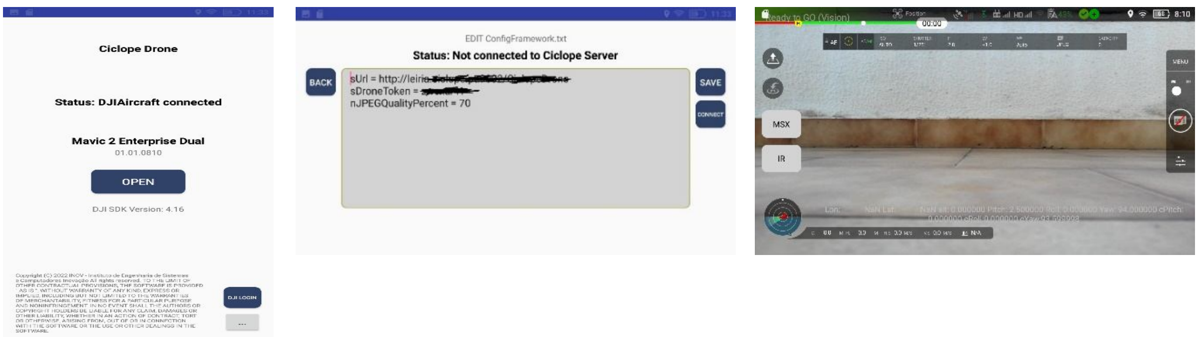

Figure 11. Ciclope Drone Android application and it's 3 screens. left) DJI SDK and aircraft connection screen middle) server configuration screen right) flight conrtol screen with network quality feedback

The application has essentially 3 main xml screens (Fig. 11) with 3 respective java activity files, the application's java file and a java class that manages and stores the server connection configuration.

activity_connection.xml (Fig. 11, left)

Is based on DJI's template examples used for registering/validating the use of their SDK with DJI servers, it confirms the connection with the specific drone, and provides an open button which goes to the main screen of the application. In our application we added a "…" button that leads to the activity_configframework screen described below.

activity_configframework.xml (Fig. 11 middle)

In this screen the user can configure the connection to the Ciclope server, url, drone token and image quality of the uploaded images. In addition, it allows to save and test the entered connection with appropriate user feedback.

activity_main.xml (Fig. 11, right)

This screen renders the images from the drone camera, provides standard flight buttons and control (ubiquitously inherited) such as takeoff, return to home, prints the drone telemetry on screen, shows the status of various signals in the top bar including the network quality button which will be described in the legend of Fig. 12.
Each of the screens needs to describe in their respective java activity files, the behavior for specific events, such as onCreate, onResume, onClick, onDestroy, onPause, onStop.
The core of the application is undertaken in the MainActivity.java file, it establishes a HTTP connection with the specified Ciclope server, acquires drone telemetry and images, assembles a SOAP message to transmit and provides visual feedback on the network and sensors.
We encountered some problems with the KSOAP2 library, so we opened an HTTP connection and configured an XML message instead. During development, the SmartSniff [10] application was particularly useful to verify that the data was been sent from the application to the server. In addition, the application SoapUI 5.6.0 [11] was also useful to identify and test wsl services of the server.
In the beginning of the development we explored the android-videostreamingdecodingsample example which allows to have access to the full resolution of the video, but we encountered some challenges in frame rate and in accessing individual frames to associate drone telemetry without creating visual synchronization artefacts. In addition, the example does not follow the DJI application model of having a ConnectionActivity at the start which separates DJI SDK registration from the main activity of the application, thus creating further challenges in integration. The present solution uses HDMI images captured in the TextureView of the smart controller. The development time for the first prototype provided to the end user on the 16th December 2021 for initial testing was approximately 3 months.
In addition to the top-bar signal feedback, the application provides an MSX button (Fig. 13) which changes the video source being displayed in the controller to edge detection performed on the visual spectrum image overlaid on the thermal IR image (Fig. 14), this mode is particular useful for helping with spatial context. Finally, the IR button switches to infrared and the Visible button to visible spectrum images. Note that the colour ramp maximum temperature value in the thermal images can be configured in advance using the DJI flight application which also is installed in the smart controller.

C.Network quality feedback

As mentioned in section II, A-3, the Ciclope system can be assessed through a PC of phone. However, for a drone operator it is crucial that they maintain line of sight with the drone or with the smart controller. Changing their focus invites accidents. In addition, the Wi-Fi signal strength does not answer whether the images with drone telemetry are being received or not in the Ciclope servers, hence we decided to incorporate a network quality feedback icon in the controller application itself. In the situation of a weak connection, we discern loss of connection going one level below SOAP, by using urlConnection:
urlConnection.setConnectTimeout(3000);
urlConnection.setReadTimeout(3000);
We then catch this specific Java event. Table I shows the code where REDE means Network. Depending on the server response, the network quality feedback is updated accordingly.

1– Drone flight mode
2– GPS signal, number of satellites that see the drone
3– Proximity sensors/automatic collision prevention (in the image legend above the front and back sensors are activated). For smoother movement, position the smart controller button to the T-tripod position 

, or to the P-position (which provides more speed than in T), instead of the S-speed position which favors agility and switches off the proximity sensors (up, down, left, right, front, back). When turning on the smart controller in the T-position it is necessary to toggle to the P-position followed to the T-position in order to activate all proximity sensors (including the lateral)
4– Signal strength of the controller to the drone
5– Signal strength of the video of the drone to the controller application*
6– Wi-Fi signal strength
7– Battery level of the drone (e.g. 90% or N/A if the drone is switched off)
8– Flight readiness/standard DJI functionality (if the drone is turned on and ready to fly a tick sign in a green circle is displayed, otherwise a red triangle with an exclamation mark is shown instead)
9– Health of the network connection to the Ciclope servers/Network Quality Feedback. When the Wi-Fi network is interrupted, the green circle with a plus sign turns immediately to a red circle with a minus sign (as shown below). In this situation the images of the drone and respective telemetry stopped being received by the Ciclope servers.
*if the video connection is bad it does not mean that the controller can't operate the drone, the connections are separate.

In addition to these 9 signals, standard DJI messages are ubiquitously displayed such as the drone's system state, e.g., magnetic interference, disconnection with the drone, restricted flight, *Ready to Go*.

Figure 12. Ciclope drone application and top bar signal legend

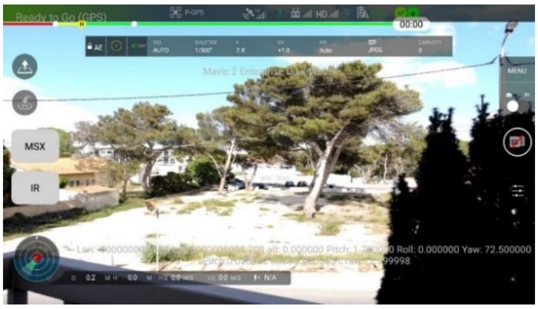

Figure 13. Ciclope drone application. User interface components and visible spectrum image

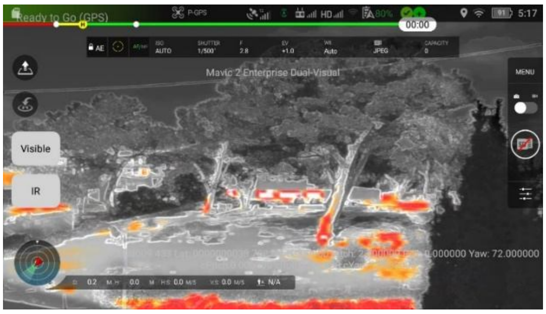

Figure 14. Ciclope drone application. DJI MSX visualization mode combining visible spectrum edge detection overlayed with the thermal image

Table I. Ciclope Drone Application, Network Exception In Java

```
public int HttpRequest()
{
          boolean REDE=false;
          boolean error=false;
          boolean TOKENVALID=true;
          BufferedReader buff_in= null;
          urlConnection= null;
          DataOutputStream wr=null;
          String sbytes=null;
             try {
                sbytes=init_xml(); // assemble string message with image and telemetry fields
                urlConnection = (HttpURLConnection) curl.openConnection();
                urlConnection.setRequestMethod("POST");
                urlConnection.setConnectTimeout(3000);
                urlConnection.setReadTimeout(3000);
                urlConnection.setRequestProperty("Content-Type", "text/xml; charset=UTF-8");
                urlConnection.setRequestProperty("SOAPAction", SOAP_ACTION);
                urlConnection.setRequestProperty("Accept-Encoding", "gzip,deflate");
                urlConnection.setDoOutput(true);
                urlConnection.setFixedLengthStreamingMode(xml.length());
                DataOutputStream(urlConnection.getOutputStream());
                wr = new DataOutputStream(urlConnection.getOutputStream());
                wr.writeBytes(xml); // send message
                int http_status = urlConnection.getResponseCode();

                if (http_status / 100 != 2) {
                   REDE = false;
                } else {
                   REDE = true;
                }
                in = urlConnection.getInputStream();

                buff_in = new BufferedReader(new InputStreamReader(in));
                String inputLine;
                StringBuffer response = new StringBuffer();
                while ((inputLine = buff_in.readLine()) != null) {
                   if(inputLine.equals(rfalse)){ TOKENVALID=false;}
                   response.append(inputLine);
                }
                if (buff_in != null) {
                   buff_in.close();
                }
                if (wr != null) {
                   wr.flush();
                   wr.close();
                }
             } catch (ClientProtocolException e) {
                Log.e("Exception:ClProtocol", e.toString());
                REDE = false;
             } catch (final java.net.SocketTimeoutException e) {
                Log.e("Exception:CONNECTIONOUT", e.toString());
                e.printStackTrace();
                REDE = false;
             } catch (IOException e) {
                Log.e("Exception:I/O", e.toString());
                e.printStackTrace();
                REDE = false;
             } catch (Exception e) {
                Log.e("Exception:exception", e.toString());
                e.printStackTrace();
                REDE = false;
             } finally {
                  if (urlConnection != null) {
                    urlConnection.disconnect();
                   }
             }
          if (REDE == false) {return 4;}
```

```
        if(sbytes.equals("zero") )
        {
           TOKENVALID=true;
        }
        if(TOKENVALID == false) {return 4;}
        return 3;
}
```

IV. Results

Fig. 15 were showing images (~2 fps) and telemetry (Lat, Lon, upwards frustum in blue) from an airborne drone integrated in the Ciclope system running on the video wall inside the white van at the image center. In addition, a deployable mobile tower with 360º pan camera also has its blue downwards/frustum rendered on the Ciclope map, heading towards it's the target yellow frustum orientation. Fig. 16 shows the thermal mode.

Figure 15. Uploaded drone visible spectrum imagery in mobile command vehicle and telemetry integration in Ciclope's map

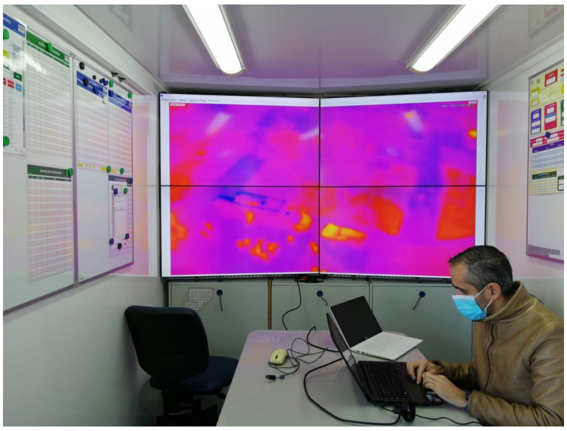

Figure 16. Uploaded drone thermal imagery in mobile command vehicle

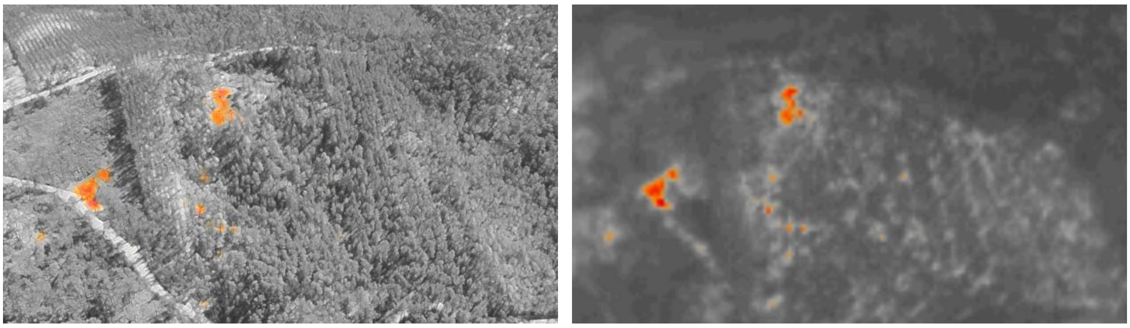

Figure 17. Hot spot detection using drone thermal imaging

V. Conclusions

Knowing that the drone images are being received or not at all times is critical when operating in remote areas. Our visual +/- network quality feedback icon solves the problem of packet delivery ambiguity present with standard Wi-Fi bars. The green circle with a plus sign continuously informs that the data is being received, turning to a red circle with a minus sign immediately when the data does not arrive.

When consulting with end users we found that the thermal capability has several co-lateral benefits in civil protection in a wider context, for example:

- Finding lost athletes through their body heat signature (GNR - Guarda Nacional Republicana end user)
- In a car accident, to establish where the heat is located before placing hands (to test the sensitivity of the thermal sensor, on request we revved the engine of the car in front the van in Fig. 15 and opened the bonnet/hood in Fig 16. (Firefighter end user)

In the context of firefighting, the use of the thermal image of the drone to detect hot spots on the ground is a process that allows to substantially reduce the number of reignitions (Cruz [12], Fig. 17). Wildland fires are fought through several fronts, characterizing physical fuel burning parameters [13] is essential for more accurate fire forecast simulations. Satellite inventory mapping amongst several applications [14] is important to feed simulations tools and provide ever faster overviews. Finally, a video documentary of the Ciclope system in action is available here [15].

Acknowledgment

The authors would like to thank Carlos Guerra (Leiria District Operation Commander, ANEPC), Ricardo Costa (Leiria sub-region 2º Commander, ANEPC, supplying Fig. 17), Paulo Santos (CIMRL – Comunidade Intermunicipal da Região de Leiria), Valério António (CIMRL), Ricardo Vieira (Leiria, GNR) for their open collaboration and active feedback, last but not least João Paulo Pereira (Leiria Sub-Chief, 2º Class, Firefighter Sapper) for his expert advice regarding DJI drone field operation, supplying Fig.17 and collaboration with the user manual of the Ciclope Drone application.

References

[1] M. Pierce, “WiFi Signal Strength: A No-Nonsense Guide”, https://www.securedgenetworks.com/blog/wifi-signal-strength, January 14th, 2021.

[2] INOV INESC Inovação, https://www.inov.pt/en/about-us/index.html, 2023.

[3] F. C. Rego, F. X. Catry, M. J. Maia, T. A. Santos, A. Gravato, I. C. Vasco, A. Esteves, F. O. Moreira, P. R. Pinto and J. Almeida, “Análise da Rede Nacional de Postos de Vigia em Portugal. (Analysis of the National Lookout Network in Portugal)”, Project Report CEABN/ADISA – INESC Inovação, December 2004. https://www.academia.edu/29095609/1_An%C3%A1lise_da_Rede_Nacional_de_Postos_de_Vigia_em_Portugal_Iniciativa_Inc%C3%AAndios_Florestais_Analysis_of_the_National_Lookout_Network_in_Portugal_.

[4] F. X. Catry, R. M. Almeida and F. C. Rego, “Production of Visibility Maps for Continental Portugal and their use in the Surveillance of Forest Fires”, Silva Lusitana 12(2), pp. 227-241, 2004.

[5] J. P. Almeida, P. Relvas, L. Silva, F. X. Catry, F. C. Rego and T. Santos, “Portuguese Lookout Towers Network Optimization Using Automatic Positioning Algorithms”, 4th International Wildland Fire Conference, Spain, 2007.

[6] P. Relvas, J. P. Almeida, F. Rego and F. X. Catry, “Study for the implementation of wildland video surveillance system in the district of Viseu”, tech report. https://www.academia.edu/478574/Estudo_para_Implementa%C3%A7%C3%A3o_de_um_Sistema_de_Videovigil%C3%A2ncia_Florestal_no_Distrito_de_Viseu, 2023.

[7] F. Saffre, H. Hildmann, H. Karvonen, T. Lind, “Monitoring and Cordoning Wildfires with Autonomous Swarm of Unmanned Aerial Vehicles”, Drones, vol. 6, no. 11, 2022.

[8] F. Saffre. “Hybrid Environmental Monitoring for Early Wildfire Detection”, 8th International Wildland Fire Conference, Oral communication/technical innovation track, (slides available in conference link below), 16th May 2023. https://www.wildfire2023.pt/program/sessions

[9] DJI, developer SDKs, https://developer.dji.com/, 2023.

[10] SmartSniff, https://smartsniff.en.softonic.com/, 2023.

[11] SoapUI, https://www.soapui.org/, 2023.

[12] L. V. Cruz, “Centro de Meios Aéreos de Alcaria entre os mais ativos de Portugal (Alcaria centre of aerial means is amongst the most active in Portugal)”, João Moderno (UEPS - Unidade de Emergência de Proteção e Socorro/ Emergency Protection and Rescue Unit, GNR Captain) and Ricardo Costa (ANEPC - Autoridade Nacional de Emergência e Protecção Civil/National Authority for Emergency and Civil Protection, Leiria sub-region 2º Commander) speaking to O Portomosense, https://oportomosense.com/centro-de-meios-aereos-de-alcaria-entre-os-mais-ativos-de-portugal/, 8th August 2023.

[13] A. Sahila, H. Boutchiche, D. X. Viegas, L. Reis and N. Zekri, "A comparative study of the combustion dynamics and flame properties of dead Mediterranean plants", International Journal of Wildland Fire, vol. 32, no. 6, pp. 979-988, https://doi.org/10.1071/WF22130, 2023.
[14] Prados, "NASA's Wildland Fire Management Program", 8th International Wildland Fire Conference, Oral communication/technical innovation track, https://appliedsciences.nasa.gov/what-we-do/wildfires, 16th May 2023.
[15] INOV, CICLOPE – Integrated video surveillance system for large areas, https://www.youtube.com/watch?v=VZIGMQ8CJ3s&list=PLrUbgdybt74LY-eUk0e0tRjKJasQoXd8F , 2023.