Real-Time Flight Recording via Mobile Devices for Autonomous and Assisted Navigation Tasks

Edison Espinosa[1(✉)], Víctor H. Andaluz[1], and Víctor Enríquez[2]

[1] Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador
{egespinosa1,vhandaluz1}@espe.edu.ec
[2] Centro de Investigación y Desarrollo de las Fuerza Aérea Ecuatoriana, Ambato, Ecuador
venriquez@fae.mil.ec

Abstract. This article presents the implementation of an application developed under the android operating system to evaluate the actions performed after a flight mission in manned aircrafts. The informatics application will be installed on an Android device that will be transported by a crew member of the aircraft. In order to obtain the required signals for the post-flight analysis, two sensors are used: Global Positioning System (GPS) and Inertial Measurement Unit (IMU), which are immersed in the Android device. The information generated at aircraft corresponding to the attitude and position is stored within a SD memory of the android device. The stored data corresponding to the position are: latitude, length, height, velocity, and hour; while the data corresponding to the attitude are Roll, Pitch, and Yaw.

Keywords: Software development · Mobile applications · Android · Debriefing · Aircraft

1 Introduction

In the military field, flight practices can be real or simulated and are carried out through flight missions that include the planning, briefing, fulfillment and debriefing phases of the mission [1]. The debriefing tasks (post-flight) are routine and mandatory once the flight practice is finished and they bring together the personnel involved in the flight mission in order to analyze everything that happened during the mission, e.g., flight path, flight behavior. the aircraft, emotional states of the pilot, skills and abilities of the pilot. The information collected with the objective that the pilots improve or maintain their performance for future flight missions [2].

In order to carry out the debriefing analysis, data obtained from the flight mission must be available that allow knowing the behavior of the personnel involved in the mission. These data are commonly stored in a file that is generated in two ways*: i) On board the aircraft,* in this case it is necessary to wait for the aircraft to complete the flight mission, in order to download the generated file and analyze the fulfillment of the

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
O. Gervasi et al. (Eds.): ICCSA 2022 Workshops, LNCS 13377, pp. 3–14, 2022.
https://doi.org/10.1007/978-3-031-10536-4_1

mission; and *ii) Remotely* for this case, a means of communication is considered that allows the monitoring of the flight mission from a remote control station. The aicraft must be permanently linked with the command station, to avoid the risk of losing flight information.

Currently there are several commercial tools, e.g., Commercial Off The Shelf (COTS) developed especially to perform virtual analysis based on military requirements using logs generated in real or simulated missions. The American company AGI has the STK Highlights tool that performs all stages of a flight from planning to debriefing [3–5]. In addition, the company Viasat has the FlightTestAnalyzer tool that allows migrating the trajectory and positioning data of the aircraft stored in a file to the Google Earth platform [3]. Finally, The Universal Flight Analysis Tool allows to simulate the flight of the aircraft by reading data from a file in which the altitude and positioning information of the aircraft is stored.

As described, this paper reports the development and implementation of a Gpsimu apk implemented in the Android PAD IP67 device, which was tested in a hexacopter and in an aircraft. The preliminary results show that the data allows the generation of information to support the Debriefing analysis in real or simulated flight missions and allows the operators to supervise and control the fulfillment of the mission in relation to the trajectory and behavior of the aircraft. In addition, this paper presents the modeling and autonomous control of an unmanned aerial vehicle (UAV). The design of a controller based on the kinematics of the UAV is presented in which the saturation of the maneuverability velocities through trigonometric functions is considered. Finally, the results obtained from the autonomous control of the UAV are presented; and the experimental results obtained from the data acquisition system for which an Aircraft Flight of the Armed Forces of Ecuador (FAE) is considered.

The paper is structured in 4 Sections including the Introduction: In Sect. 2, the developed architecture of the data acquisition system is described. In addition, the modeling and autonomous control of the UAV is presented. The tests performed are detailed in Sect. 3. Meanwhile, Sect. 4 describes the conclusions and future work.

2 Data Acquisition System Architecture

Figure 1 shows the proposed architecture of the data acquisition system that supports the execution of Debriefing tasks, considering the real-time monitoring of data obtained through the GPS (Global Positioning System) and IMU (Inertial Measuring Unit) sensors.). The information collected is analyzed with the android device scale, in order to organize the information and generate a flat file (*.csv) that is commonly stored on the computer disk or in an external device. The stored data will be analyzed with the use of commercially available tools [6].

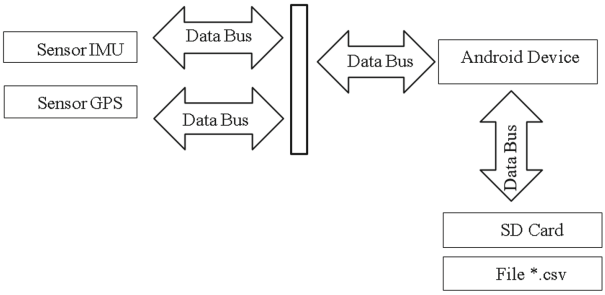
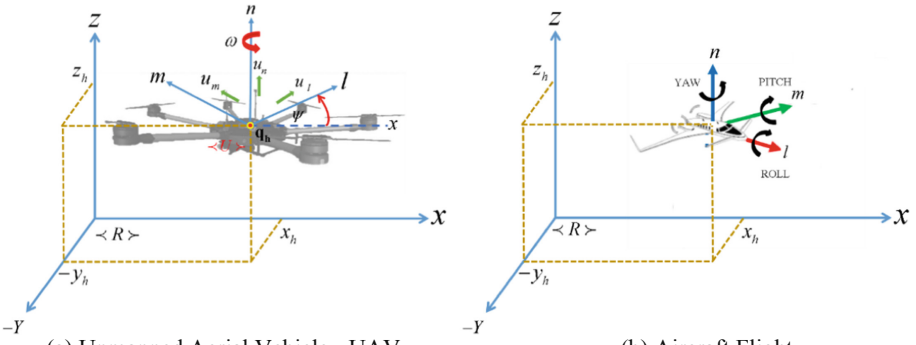


Fig. 1. Android device architecture.

The sensors that are connected to the data acquisition system can be analog or digital, coming from the GPS, IMU, and the physiological characteristics sensor. The data acquired from the GPS are latitude, longitude, height, speed, time. Meanwhile, the data acquired from the IMU are angles of roll (Roll), pitch (Pitch), yaw (Yaw). In aeronautics, three direction axes *l, m* and *n* are considered, on the mobile reference system. The *l* axis is the direction of motion (usually the axis that passes through the nose and tail of the aircraft). The *m* axis is perpendicular to the direction of movement, always in the same plane of the plane (the axis that passes through the wings). Finally, the *n* axis is considered the axis perpendicular to the previous ones, and therefore perpendicular to the plane of the plane, see Fig. 2.



(a) Unmanned Aerial Vehicle - UAV (b) Aircraft Flight

Fig. 2. Turning axes

2.1 Unmanned Aerial Vehicle Modeling and Control

To obtain the kinematic model of the UAV, the point of interest in the robot center is considered. The kinematic model of the aerial robot contains the four maneuverability velocities to achieve displacement and four velocities that represent the velocities within the frame of reference. Figure 1 represents the kinematic variables of the quadcopter on

the fixed reference frame <R>. In other words, the movement of the quadcopter over the fixed frame of the reference <R> is defined as:

$$\begin{bmatrix} \dot{\eta}_x \\ \dot{\eta}_y \\ \dot{\eta}_z \\ \dot{\eta}_\psi \end{bmatrix} = \begin{bmatrix} C_\psi & -S_\psi & 0 & 0 \\ S_\psi & C_\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ u_n \\ \omega \end{bmatrix} \tag{1}$$

Equation (1) can be described in matrix form, as follows:

$$\dot{\boldsymbol{\eta}}(t) = \mathbf{J_Q}(\mathbf{q_Q})\mathbf{u}(t) \tag{2}$$

where, $\dot{\boldsymbol{\eta}}(t) \in R^n$ represents the velocity vector of the point of interest with respect to the inertial reference system $\{\mathcal{R}\}$; $\mathbf{J_Q}(\mathbf{q_Q}) \in R^{mxn}$ is the Jacobian matrix of the UAV; and $\mathbf{u}(t) \in \Re^n$ is the maneuverability vector of the UAV with respect to $\{\mathcal{R}_\mathcal{Q}\}$with $m = n = 4$ the $\mathbf{J_Q}(\mathbf{q_Q})$ matrix is of full rank with $|\mathbf{J_Q}| = 1$.

For the UAV autonomous navigation mission $\boldsymbol{\eta}_d = [\,\eta_{xd}\ \eta_{yd}\ \eta_{zd}\,]^T$ of the UAV with a variation $\dot{\boldsymbol{\eta}}_d = [\,\dot{\eta}_{xd}\ \dot{\eta}_{yd}\ \dot{\eta}_{zd}\,]^T$ are obtained; therefore, the control error is defined as $\tilde{\boldsymbol{\eta}}(t) = \boldsymbol{\eta}_d(t) - \boldsymbol{\eta}(t)$ with its variation in time $\dot{\tilde{\boldsymbol{\eta}}} = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}}$. In order to demonstrate stability, a Lyapunov-based controller is proposed. Proposing a positive candidate function defined as, $\mathbf{V}(\tilde{\boldsymbol{\eta}}) = \frac{1}{2}\tilde{\boldsymbol{\eta}}^T\tilde{\boldsymbol{\eta}} > 0$. Taking the first derivative and replacing $\dot{\tilde{\boldsymbol{\eta}}} = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}}$ and considering that $\dot{\boldsymbol{\eta}} = \mathbf{J_Q}\mathbf{u}$ is obtained, $\dot{V}(\tilde{\boldsymbol{\eta}}) = \tilde{\boldsymbol{\eta}}^T(\tilde{\boldsymbol{\eta}}_d - \mathbf{J_Q}\mathbf{u})$. The proposed control law for the UAV is defined as,

$$\mathbf{u} = \mathbf{J_Q^{-1}}(\dot{\boldsymbol{\eta}}_d + \mathbf{Q}\tanh(\tilde{\boldsymbol{\eta}})) \tag{3}$$

where, $\mathbf{Q}$ is a positive diagonal gain matrix. By introducing (3) into the time derivative of $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}})$ it is obtained.,

$$\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}}) = -\tilde{\boldsymbol{\eta}}^T\mathbf{Q}\tanh(\tilde{\boldsymbol{\eta}}) < 0 \tag{4}$$

As described, the equilibrium point is asymptotically stable, *i.e.* $\tilde{\boldsymbol{\eta}}(t) \to 0$ asymptotically. If you relax the assumption of perfect speed tracking you have to $\tilde{\mathbf{u}} = \mathbf{u}_d - \mathbf{u}$ where (4) is defined as $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}}) = \tilde{\boldsymbol{\eta}}^T\tilde{\mathbf{u}} - \tilde{\boldsymbol{\eta}}^T\mathbf{Q}\tanh(\tilde{\boldsymbol{\eta}})$. Therefore, the error $\tilde{\boldsymbol{\eta}}(t)$ is delimited by,

$$\|\tilde{\boldsymbol{\eta}}\| \leq \frac{\|\tilde{\mathbf{u}}\|}{\zeta\lambda_{\min}(Q)};\ \text{with } 0 < \zeta < 1 \tag{5}$$

Therefore, if $\tilde{\mathbf{u}}(t) \neq 0$ the formation error $\tilde{\mathbf{h}}(t)$ is ultimately bounded by (5).

2.2 Data Acquisition System Algorithm

The algorithm of the acquisition system is implemented in an apk called Gpsimu that must be downloaded to mobile or fixed devices. This algorithm includes three blocks of code in the first lines of code enable permissions to the GPS device to

determine that the API use WiFi data or mobile data that establishes the location of the device. The API returns the location through the available location providers, including the GPS global positioning system and WFi data and mobile data cell phone using the android. Permission functions. ACCESS_COARSE_LOCATION and android.permission. ACCESS_FINE_LOCATION. The second block of code allows to establish the trajectory and the behavior of the aircraft based on obtaining the positioning, height, speed, time, attitude, Roll, Pitch, Yaw. Finally, the third block of code allows the information to be stored in a *.cvs file that will then be processed for mission analysis. The operation algorithm of the acquisition system represented in the flow diagram is shown in Fig. 3.

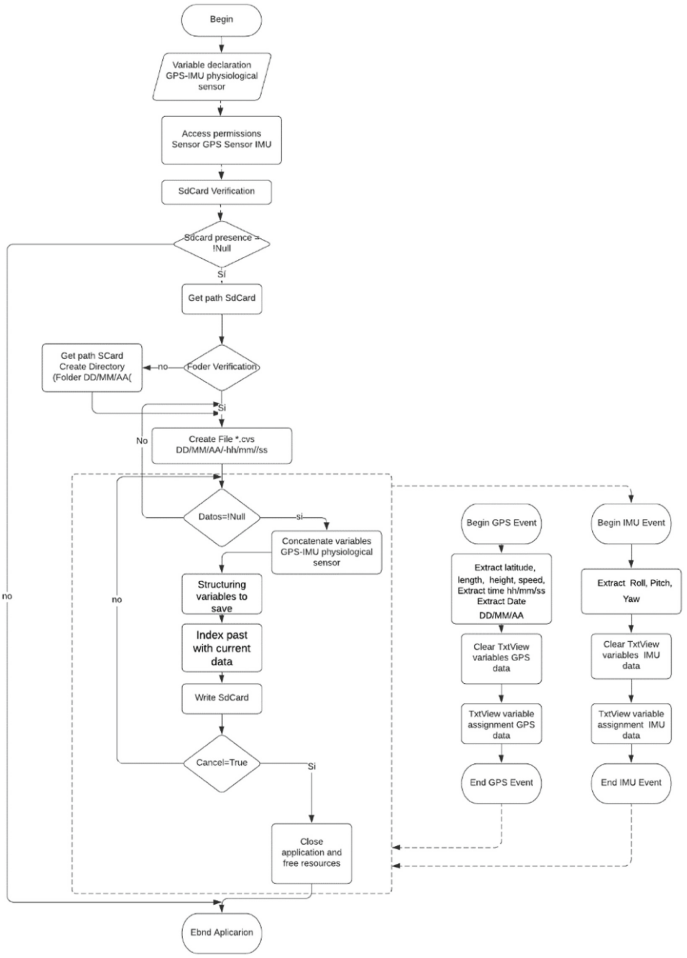


Fig. 3. Algorithm of operation of the data acquisition system.

2.3 Operation of the Data Acquisition System

The Gpsimu APK must be installed on a mobile or fixed processing device. For this work, an IP67 PAD that is placed on board the aircraft was considered, as shown in Fig. 4.



Fig. 4. PAD IP67 on board of the aircraft

The data of the signals generated by the aircraft during the mission are recorded in the IP67 PAD in a flat file in *.cvs format where the fields are separated by commas or semicolons [7, 8]. The flat file structure is shown in Table 1.

Table 1. Structure of the positioning archive file

Extension	Name	Fields
CSV	Date	Date, Hour, seconds, Latitude, Longitude, Altura, Distance (m), Velocity (km/h), Roll(deg), Pitch(deg), Yaw(deg)

3 Experimental Tests

Several experimental tests were developed in order to evaluate the performance of the proposed data acquisition system. The system tests to establish the trajectory and behavior of the aircraft were carried out first using a hexacopter and then on an FAE (Armed Forces of Ecuador) aircraft where the PAD IP67 device was placed, see Fig. 5. The test in the hexacopter it was short-lived. Whereas, on the air-craft it was long lasting. Figure 6 shows an extract of the captured data. Figure 6(a) shows an extract of the time-stress relationship for 100[s], the minimum stress is 11,09, while the maximum stress is about 12,8. Figure 6(b) shows the heart rate maintained a minimum average frequency of 65,08 and a maximum frequency of 76,79. Finally, Fig. 6(c) shows temperature during the 100[s].



(a) Unmanned aerial vehicle with six rotating propellers

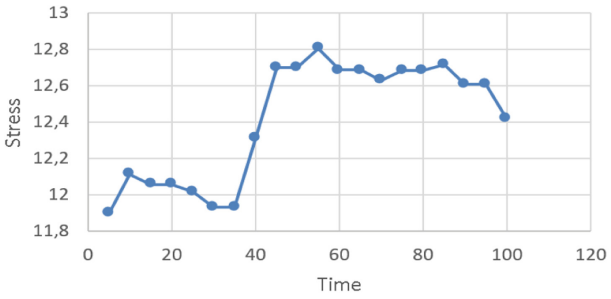


(b) Aircraft Flight

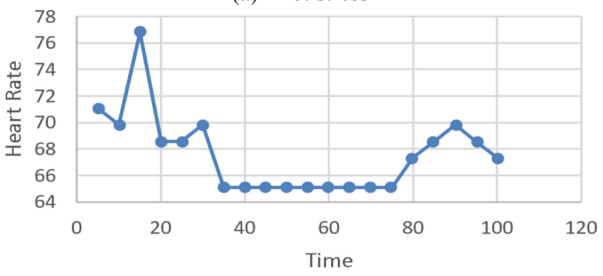
Fig. 5. Aerial vehicles for experimental tests with the IP67 PAD device

3.1 UAV Autonomous Navigation

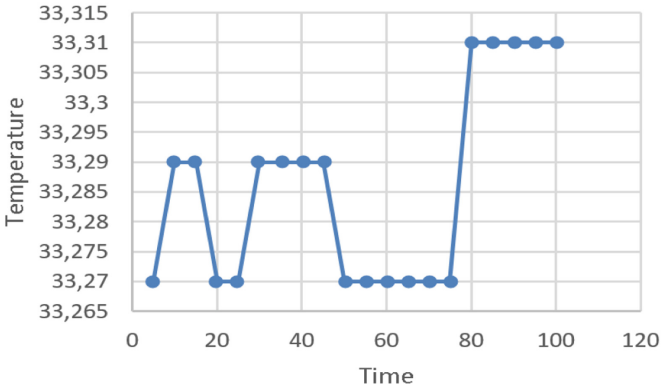
For the control of the hexacopter, an autonomous mission is considered, for which the control law proposed in Subsect. 2.1 is considered. Figure 7 presents the results of the experiment, showing the desired task and the task obtained by implementing the autonomous control through the controller proposed in the Eq. (3). Figure 8 shows the control errors, where it can be seen that the errors are finally bounded, as demonstrated in the robustness analysis in Eq. (5).



(a) Pilot stress



(b) Pilot heart rate



(c) Pilot temperature

Fig. 6. Extract of data stored in the archive.

3.2 Results of the Roll, Pitch and Yaw Attitude Variables

Aircraft navigation angles are a type of Euler angle that is used to indicate the orientation of an object in three dimensions. The angles as indicated are yaw, pitch, and roll. [9] The device used to store the three angles that give the aircraft attitude has an internally inertial sensor (IMU) which delivers the values of the angles in radians (roll, pitch and yaw). To verify the efficiency of the sensor used to acquire the aircraft's attitude parameters, a comparison pattern was used. The 3DMGX525 sensor [10] was in-stalled on the platform with the data acquisition device and the data obtained was logged and the results generated are shown in Figs. 9, 10 and 11.

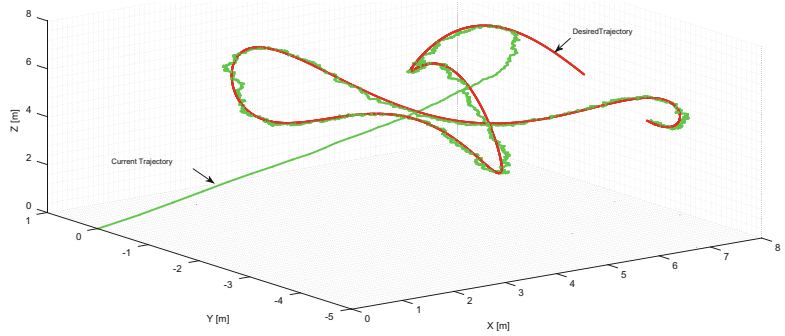


Fig. 7. Stroboscopic movement of the autonomous control of the UAV

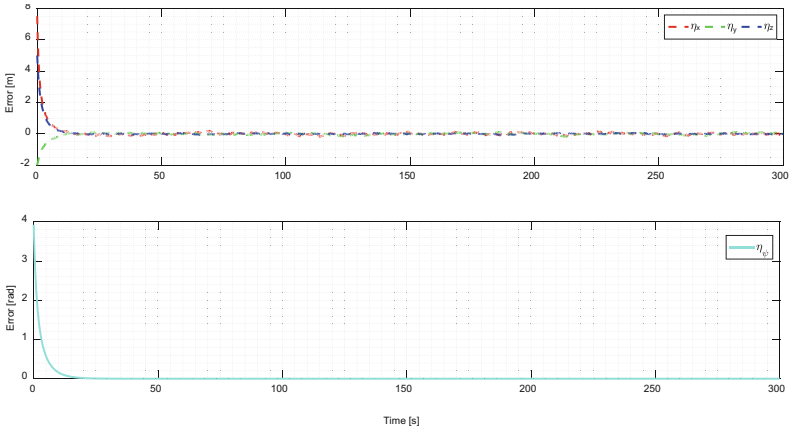
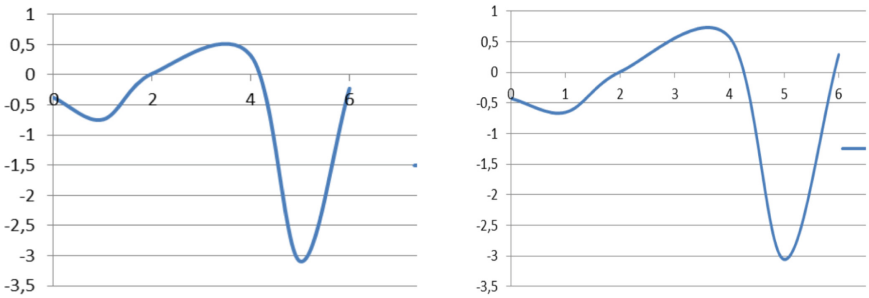
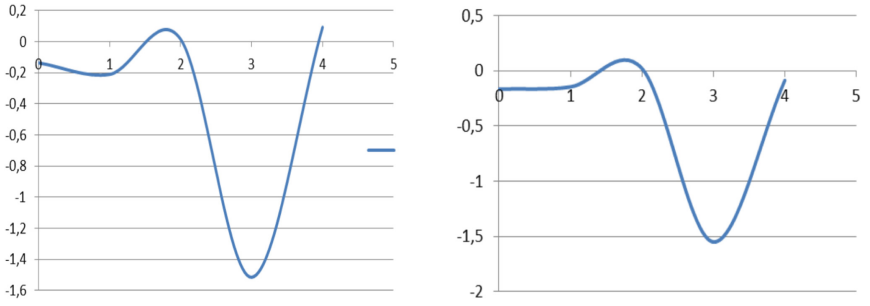


Fig. 8. UAV control errors



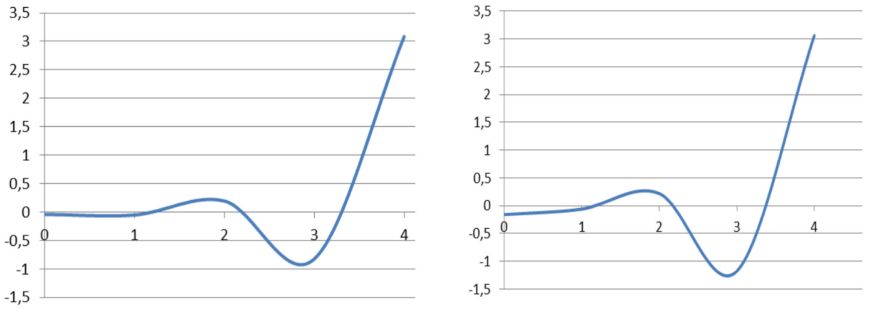
(a) 3DM-GX5-25 Sensor (b) Android device

Fig. 9. Representation of the roll angle obtained with the 3DM-GX5-25 sensor and the Android



(a) 3DM-GX5-25 Sensor (b) Android device

Fig. 10. Representation of the pitch angle obtained with the 3DM-GX5-25 sensor and the Android

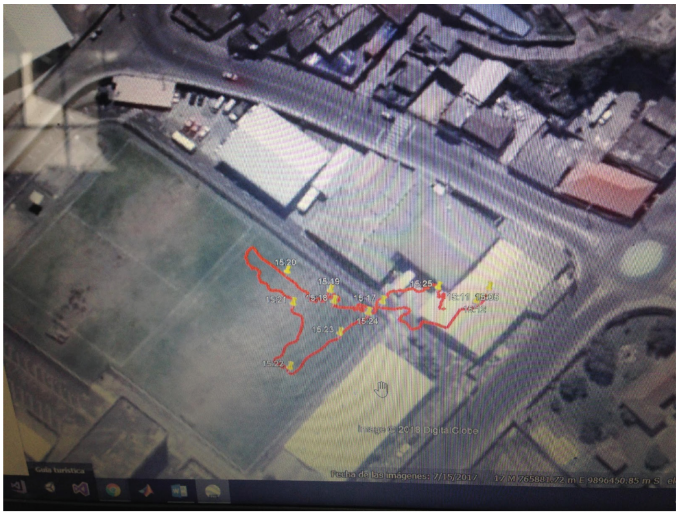


(a) 3DM-GX5-25 Sensor (b) Android device

Fig. 11. Representation of the yaw angle obtained with the 3DM-GX5-25 sensor and the Android device.

3.3 Results of the Positioning Variables

The data obtained on board the hexacopter and the King Air FAE 1141 manned aircraft that were recorded and stored using the Gpsimu apk installed in the IP67 PAD were used as input in the Tacview flight data analysis tool to generate the trajectory and behavior shown in Fig. 12.



(a) Flight Reconstruction of the Unmanned Aerial Vehicle With Six Rotating Propellers.



(a) Flight Reconstruction of the Aircraft Flight

Fig. 12. Representation of the data obtained from the log generated from the hexacopter and aircraft file

4 Conclusions

This research was aimed at solving the problem of obtaining data in real time to carry out debriefing analysis based on storing data in real time for subsequent processing and analysis. The Gpsimu apk was developed and installed in an Android PAD IP67 device and the tests were generated in a hexacopter and an aircraft. The results found when analyzing the data stored in the file with the *.csc extension is coherent, practical and

very useful to carry out the Debriefing analysis in real or simulated missions and allows the operators to supervise and control the fulfillment of the mission. Finally, as future work, it is proposed to visualize the trajectory and behavior of the aircraft in real time in the same mobile device app.

Acknowledgements. The authors would like to thank the Universidad de las Fuerzas Armadas ESPE for the financing given to research and development of the project "Sistema experto para el análisis y diagnóstico en tiempo real de una misión de vuelo"; also to ESPE Innovativa E.P.; Centro de Investigación Científica y Tecnológica del Ejercito (CICTE); Centro de Investigación y Desarrollo de las Fuerza Aérea Ecuatoriana (CIDFAE); and Grupo de Investigación GITBIO y ARSI, for the support for the development of this work.

References

1. IFIS3 (2018). http://www.ais.aviacioncivil.gob.ec/ifis3/aip/ENR%201.10
2. Mission Debriefing - Datatoys Data & Video Systems by Bad Wolf Technologies (2018). http://www.datatoys.com/industries/defense-government/mission-debriefing/
3. ViaSat Inc. Video/ISR Data Links | ViaSat (1986–2016). https://www.viasat.com/products/isr-data-links
4. Bansal, S.K.: Towards a semantic extract-transform-load (ETL) framework for big data integration. In: 2014 IEEE International Congress on Big Data, Anchorage, AK, pp. 522–529. IEEE Press, New York (2014). https://doi.org/10.1109/BigData.Congress.2014.82
5. AGI Engineering Tools STK Highlights. https://www.agi.com/products/engineeringtools
6. Tacview The Universal Flight Analysis Tool. https://www.tacview.net/
7. Pimienta, P.: Arquitecturas de aplicaciones moviles. ZENVA PTY LTD (2018). https://deideaaapp.org/arquitecturas-de-aplicaciones-moviles/
8. Mahmud, S., Hossin, M., Jahan, H., Noori, S., Bhuiyan, T.: CSV-ANNOTATE: generate annotated tables from CSV file. In: 2018 International Conference On Artificial Intelligence And Big Data (ICAIBD) (2018). https://doi.org/10.1109/icaibd.2018.8396169
9. Shafranovich, Y.: Common Format and MIME Type for Comma-Separated Values (CSV) Files. IETF RFC 4180, October 2005. http://tools.ietf.org/html/rfc4180
10. VECTORNAV, Embedde Navigation Solutions VN-200 GPS-Aided INS. https://www.vectornav.com/products/vn-200?gclid=Cj0KCQjwof3cBRD9ARIsAP8x70PiuHeXjIZ8f0MZQewQKrseEsjRKlefnnF-Rtey_2A6fHfE9jCoEoMaAleZEALw_wcB