A UAV Platform Based on a Hyperspectral Sensor for Image Capturing and On-Board Processing

PABLO HORSTRAND[ID]**, RAÚL GUERRA**[ID]**, AYTHAMI RODRÍGUEZ, MARÍA DÍAZ**[ID]**,**
SEBASTIÁN LÓPEZ[ID]**, (Senior, Member, IEEE), AND JOSÉ FCO. LÓPEZ**
Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria, 35017 Las Palmas de Gran Canaria, Spain

Corresponding author: Pablo Horstrand (phorstrand@iuma.ulpgc.es)

ABSTRACT Application-oriented solutions based on the combination of different technologies such as unmanned aerial vehicles (UAVs), advanced sensors, precise GPS, and embedded devices have led to important improvements in the field of cyber-physical systems. Agriculture, due to its economic and social impact on the global population, arises as a potential domain which could enormously benefit from this paradigm in terms of savings in time, resources and human labor, not to mention aspects related to sustainability and environment respect. This has led to a new revolution named precision agriculture (or precision farming), based on observing and measuring inter and intra-field variability in crops. A key technology in this scenario is the use of hyperspectral imaging, firstly used in satellites and later in manned aircraft, composed by hundreds of spectral bands which facilitate hidden data to be converted into useful information. In this paper, a hyperspectral flying platform is presented and the construction of the whole system is detailed. The proposed solution is based on a commercial DJI Matrice 600 drone and a Specim FX10 hyperspectral camera. The challenge in this work has been to adopt this latter device, mainly conceived for industrial applications, into a flying platform in which weight, power budget, and connectivity are paramount. Additionally, an embedded board with advanced processing capabilities has been mounted on the drone in order to control its trajectory, manage the data acquisition, and allow on-board processing, such as the evaluation of different vegetation indices (the normalized difference vegetation index, NDVI, the modified chlorophyll absorption ratio index, MCARI, and the modified soil-adjusted vegetation index, MSAVI), which are numerical and/or graphical indicators of the vegetation properties and compression, which is of crucial relevance due to the huge amounts of data captured. The whole system was successfully tested in a real scenario located on the island of Gran Canaria, Spain, where a vineyard area was inspected between May and August of the year 2018.

INDEX TERMS Unmanned aerial vehicle, hyperspectral, pushbroom sensor, vegetation index, on-board processing.

I. INTRODUCTION

Nowadays, there is an increasing interest in the use of unmanned aerial vehicles (UAVs) to collect data for inspection, surveillance and monitoring in the areas of defense, security, environmental protection and civil domains, among others. The potential of these aerial platforms in relation to others such as satellites or manned airborne platforms is that

they represent a lower-cost approach with a more flexible revisit time and a better spatial and spectral resolution, which permits a deeper and more accurate data analysis [1].

In the scientific literature, we can find several research publications in the aforementioned fields which confirm the current high demand of these aerial vehicles. Just to name some, in [2] UAVs are used to detect power lines to achieve automatic power line surveillance and inspection. [3] presents several missions in different safety, security and rescue field tests using UAVs and [4] illustrates the advantages offered

---

The associate editor coordinating the review of this manuscript and approving it for publication was Maurizio Magarini.

by the introduction of small-scale UAVs in the near future in civilian applications, concretely, in police departments, fire brigades and other homeland security organizations. However, it is in the agriculture field where remote sensing relying on UAVs has been strongly positioned as an emerging field of application [5]. In this context, the use of UAVs permits periodically monitoring the plants during their growth and until the harvest and also controlling all external conditions that may affect their health. They allow collecting periodical information of the crops using different kinds of sensors. In this scenario, several multispectral sensors are being widely used on-board UAVs for collecting spectral information that allows the generation of maps for indicating the aspects of the plant state [6]. One of the most typical examples is the normalized difference vegetation index (NDVI), which indicates the vigorousity of the plants using the information of two different spectral channels placed in the red and near infrared parts of the electromagnetic spectrum [7].

Nevertheless, there are many more indices, in addition to the NDVI index, that provide useful information for smart farming applications [8]. These indices use the spectral information corresponding to different parts of the electromagnetic spectrum. Due to this reason, the main goal of this research work is to develop an autonomous system able to carry a hyperspectral sensor for collecting information in a wide range of spectral channels. While the multispectral sensors typically carried by these UAVs collect just some spectral bands, the hyperspectral sensors are able to sense hundreds of very narrow spectral channels, providing information that may be extremely useful not only for smart farming applications, but also for other applications such as target detection, anomaly detection, and classification, among others. However, the use of hyperspectral sensors in UAVs, instead of multispectral sensors, is not exempt of drawbacks. For instance, while there are multispectral sensors that have been specifically designed for being carried by a drone and for being managed using embedded devices, finding a hyperspectral sensor that can be directly set up in UAVs is not an easy task.

In this work, a Specim FX10 VNIR (visible and near infrared) hyperspectral pushbroom sensor [9] has been successfully set up in a DJI Matrice 600 drone [10] for collecting high quality hyperspectral images. This sensor is able to provide hyperspectral data with 224 spectral bands with spectral wavelengths between 400 and 1000 nm, and a spatial resolution of 1024 hyperspectral pixels per image line. Additionally, an industrial IDS RGB camera [11] has also been mounted in the UAV with the goal of providing extra information in order to improve the spatial quality of the acquired hyperspectral data. These two sensors have been placed in a DJI Ronin MX gimbal for reducing the drone vibrations and increasing the quality of the acquired images. The system includes a Jetson TK1 NVIDIA embedded device [12] used as on-board computer for autonomously controlling the drone flight and managing the data acquisition. This board also enables the possibility of carrying out the on-board processing of the

acquired data, what highly increases the applicability of this acquisition system. In addition to the devices carried by the drone, a ground station composed by a DJI radio controller (RC) transmitter and an iPad tablet with an ad-hoc application developed for this specific task has been set up. This tablet communicates with the on-board computer through the RC transmitter in such a way that the desired acquisition mission can be easily configured by a non expert user using a custom developed iOS application.

The acquisition system developed in this work has been tested in different scenarios where a set of images have been collected over two different vineyards in the region of Tejeda, Gran Canaria, Spain. These images have been used for generating a set of different maps based on some selected vegetation indices (VIs) in order to analyze specific terrain properties. In addition, the possibility of on-board compressing the acquired hyperspectral data in real-time, in such a way that it can be efficiently transferred to a ground control station for its further processing, has been also studied. For doing so, the HyperLCA compressor [13] has been implemented using NVIDIA CUDA (compute unified device architecture) for taking advantage of the parallelism of the low power graphics processing unit (GPU) included in the Jetson TK1 NVIDIA device. The obtained results verify the achievement of real-time compression performance, also demonstrating the benefits of carrying this kind of on-board computing devices in such aerial platforms.

This manuscript is organized as follows. Section II displays the main characteristics of the different devices included in the proposed UAV acquisition system as well as in the ground station. This section also uncovers the benefits provided by each of the included devices to the whole system. Section III provides a detailed explanation on how each of the devices described in Section II has been integrated in the system. Section IV introduces the software application developed for the individual elements in the system and their interconnection, starting with the mission configuration using the developed iOS application, followed by the camera control and the flight control application running on the on-board device. Section V gives information about the real hyperspectral data acquired by the system. This data is analyzed and processed in different ways in Section VI. First, the data is calibrated in Section VI-A and then analyzed using the NDVI, the modified chlorophyll absorption ratio index (MCARI) and the modified soil-adjusted vegetation index (MSAVI) in Section VI-B, showing an example of the possible benefits that entitle carrying a hyperspectral sensor on a UAV acquisition system. Other processing capabilities are outlined in section VI-C, presenting the results of a simple classifier applied over a portion of one of the captured images. Section VI-D displays the real-time compression results obtained for the acquired data, demonstrating the benefits of carrying an on-board computing device with a relatively high computational capability. Finally, Section VII discloses the obtained conclusions and outlines further research lines.
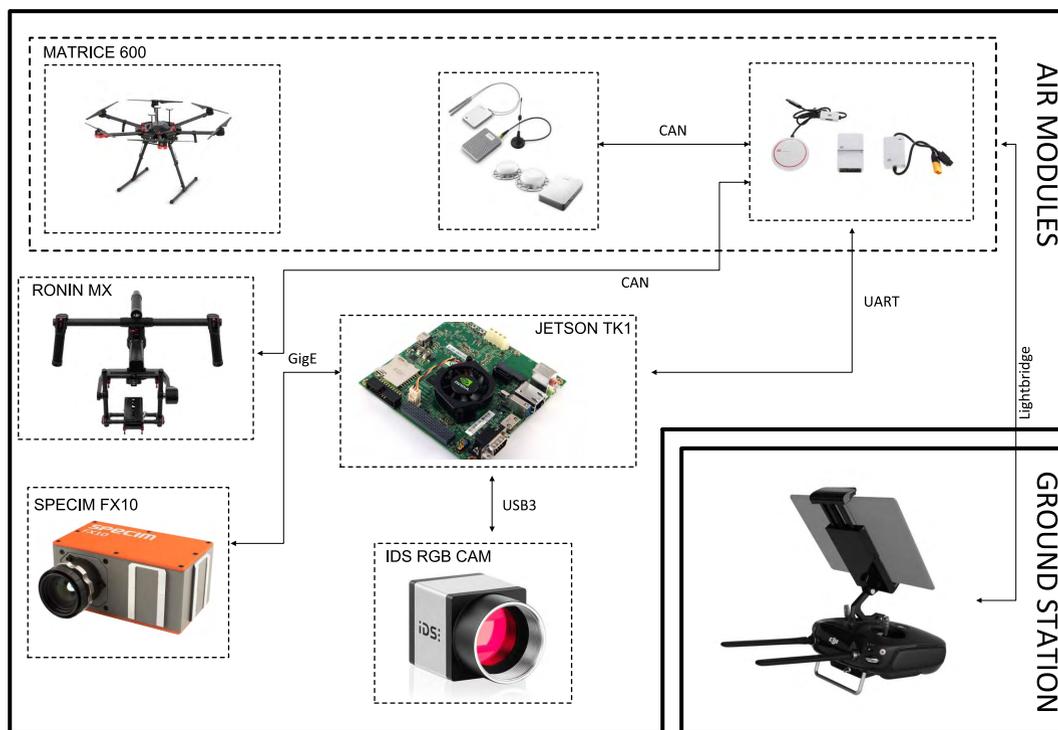
FIGURE 1. General overview of the UAV platform.

II. PLATFORM DEVICES DESCRIPTION

This section details the characteristics of the hardware elements being used in the platform, the main advantages they offer and how they are fitted into the whole system. Figure 1 shows a general overview of the platform elements and how they are connected.

A. HYPERSPECTRAL SENSOR

Once the decision of using a hyperspectral sensor over a multispectral one has been justified, the next point is to select the proper technology among the available hyperspectral cameras in the market [14]. Generally speaking, all the currently available hyperspectral cameras are based on a panchromatic sensor, lying the differences between them on the location of the filter. One of the state-of-the-art hyperspectral solutions is based on a 2D imager snapshot sensor where the hypercube is obtained in one shot as the filter is built on-chip. The Firefleye UHD 185 implements that technology, and due to its low weight is a good candidate to be mounted on a UAV [15]. Nevertheless, this type of camera has an inverse relationship between the spatial resolution and the spectral one, since the sensor dimensions do not vary. This translates, in the case of the UHD 185, in a spatial resolution of the hyperspectral cube of 50×50 pixels, not sufficient for many applications despite the up-sampling process the camera carries out with the aid of an additional sensor built in the camera.

Pushbroom hyperspectral cameras currently present a better trade off between spatial and spectral resolution, with a better range of prices available in the market due to their

popularity, hence, we have opted for such technology in this work. However, the integration of these cameras into a UAV is complex since the frames captured share no overlap, therefore much more care and attention has to be put into the acquisition and post-processing. The scientific community has set quite a lot of effort in the post-processing phase, correcting remotely sensed images both geometrically and radiometrically either by having a very precise positioning system to have the camera orientation and position for every frame [16] or by having incorporated an RGB snapshot sensor [17] that through image processing is able to make a 3D reconstruction and obtain the hyperspectral camera position in each captured frame. As it will be further detailed, one of the novelties of this work is on the acquisition system, on one hand making the whole process completely automatic and user friendly, and on the other hand on fine controlling the speed and the positioning of the platform, improving the quality of the captured data and reducing the post-processing efforts, or even eliminating them for some specific applications where the user demands a prompt result.

The Specim FX series hyperspectral pushbroom cameras have been specifically designed for industrial applications to enhance quality control processes. The target application will determine which devices should be mounted in the UAV. For the particular case of precision agriculture, our option has been to acquire the FX10 as most of the biochemical and biophysical attributes of the crops are obtained in the VNIR range. The FX10 camera (shown in Figure 1) is a hyperspectral imaging instrument, mainly designed for

industrial and laboratory use. It works as a push-broom hyperspectral scanner, collecting hyperspectral data in the VNIR (400 to 1000 nm) region through single fore optics.

However, this is the first time that an industrial hyperspectral camera like this is installed in a DJI Matrice 600, and the challenge has been to do it in a way in which communications, synchronization and control are optimized in order to extract the best of it. Under well controlled conditions, such as in a laboratory, the integration is straightforward. However, if the camera is to be included in an UAV, aspects such as vibrations and wind could affect the quality of the images. This is the reason for using a high quality gimbal as well as an industrial RGB camera, which is used for extracting the exact position and rotation of each of the images and permits future corrections in the hyperspectral lines obtained at the same time. Table 1 shows the main characteristics of the hyperspectral sensor.

TABLE 1. Specim FX10 main characteristics.

Spectral Range	400 - 1000 nm
Spectral Bands	224
Spatial Sampling	1024 px
Spectral FWHM	5.5 nm
Frame Per Seconds	330 FPS full frame 9900 FPS with 1 band selected
FOV (α)	38°
Camera SNR (Peak)	600:1
Camera Interface	GigE Vision
Dimensions	150 x 85 x 71 mm
Weight	1.26 Kg

B. THE DJI MATRICE 600 DRONE

The DJI Matrice 600 is an industrial drone, equipped with the latest generation of flight controllers from DJI, the A3 controller. This module can be connected to an external board through the UART port. The programming of the board is supported by the Onboard SDK. This enables developers to implement a wide variety of custom mission programs for different applications. Table 2 presents a summary of the main characteristics of the Matrice 600. The drone fulfills all the requirements for our application in terms of maximum flight speed, clearly above the flight speed range performed in these missions that oscillates between 2 and 5 m/s, and in terms of the payload (6 Kg), which is above the total payload of the system. The drone hovering time with the selected battery type goes from 35 minutes (no payload) to 16 minutes (6 Kg payload). In our particular case, having an approximate payload of 4.5 Kg which makes an absolute take-off weight of 13.5 Kg, the hovering time is around 20 minutes.

TABLE 2. Matrice 600 main characteristics.

General	Max take off weight	15.1 Kg
	Max speed (without wind)	18 m/s
	Max angular velocity	Pitch: 300°/s, Yaw: 150°/s
	Number of Batteries	6
	Weight (with batteries)	9.1 Kg
Battery	Model	TB47S
	Type	LiPo 6S
	Capacity	4500 mAh
	Weight	595 g
Motor	Model	DJI 6010
	kV	130 rpm/V
	Weight	230 g
Propeller	Model	DJI 2170
	Diameter × Thread Pitch	533 × 178 mm
	Weight	58 g

1) PRECISION GPS GNSS SYSTEM: DJI RTK

Flying platform positioning and heading are very important for an accurate flight control, and useful as well for post-processing the captured data. That is why data are not only used during the flight for control but also stored into a file. In this work, a precise device such as the DJI RTK [18] has been acquired in order to have an accurate positioning of the system. The device can deliver 1 centimeter of horizontal accuracy and 2 centimeters of vertical accuracy.

The main drawback of the device is its sampling frequency, which can not be set higher than 50 Hz. This means that for some flights there will be frames not positioned and hence, an ulterior interpolation is required. This limitation comes from the communication between the flight controller A3 and the Jetson TK1.

2) CAMERA STABILIZATION SYSTEM: DJI RONIN-MX

In order to damp as much as possible flight vibrations mainly produced by the wind, the camera has been mounted on a Gimbal, more concretely the Ronin Mx [19] from DJI, in order to ease its integration in the flying platform. This gimbal is mainly intended for filming cameras, that is why a great deal of effort has been spent to mechanically adapt the FX10 to the Ronin-Mx as it is explained in the next section.

The device incorporates a gyroscope which in this case has also been used to get the camera orientation, roll, pitch and yaw values. However, again the drawback is the maximum sampling frequency of 50Hz, when reading the values through the onboard SDK, as it is the case, and an accuracy of 0.1 degrees in all directions. Since having an accurate orientation of every captured hyperspectral frame is quite critical for geometric correction, a second system based on an industrial RGB camera has been installed and it is explained hereafter.

TABLE 3. Jetson TK1 features.

GPU	NVIDIA kepler GK20 with 192 SM3.2 CUDA cores (up to 326 GFLOPS)
CPU	NVIDIA 2.32GHz ARM quad-core CPU with Cortex A15 battery saving shadow core
DRAM	2GB DDR3L 933MHz EMC x16 using 64-bit data width
Storage	16GB fast eMMC 4.51 (routed to SDMMC4)
Interfaces	Mini-PCIe, USB 3.0, USB 2.0, HDMI, RS232, Ethernet, SATA, JTAG, EXPANSION I/O, ...

TABLE 4. Industrial IDS RGB camera features.

Sensor Model	IDS UI-3140CP
Sensor Type	CMOS Color
Resolution	1280×1024 Pixel
Optical Sensor Class	1/2"
Max. Frame Rate	224
Exposure time (minimum - maximum)	0.035ms - 434ms
Optics	3.5mm HR
Sensor Interface	USB 3.0

C. ON-BOARD EMBEDDED PLATFORM: JETSON TK1

The development kit NVIDIA Jetson TK1 is a super computer with a GPU based on the Kepler architecture that includes all the basic functions to develop embedded applications and provides a NVIDIA CUDA platform with all the necessary tools to accelerate developments of high computational loads. The characteristics of this embedded system are shown in Table 3. Flight control, camera capture, communication and processing algorithms have been implemented in the Jetson TK1 within this work. There are more powerful devices available in the market in terms of computation, but the Jetson TK1 presents a good trade-off between size and weight, and the number of available computational resources. Moreover it has the required interfaces to build up this platform. It comes with Gigabit Ethernet port RJ45, which is being used to connect the hyperspectral camera, a USB 3.0 port to connect the RGB camera with a high data transfer as well, a USB 2.0 port to make the serial connection with the A3 controller through a USB-TTL device and finally, a SATA interface that allows the integration of an external SSD for a very fast data transfer to memory, which is a key point in this type of applications handling an enormous amount of information.

D. INDUSTRIAL RGB CAMERA

An industrial RGB camera with a very high frame rate has also been incorporated into the system to assist hyperspectral frame registration and later correction. The sensor comes with a USB 3.0 interface and a SDK that allows its integration in real-time system platforms, being able to do parameter setup and image capture autonomously. These characteristics make this sensor optimal for this application. Table 4 shows the main characteristics of the sensor and its optics. The sensor has a maximum resolution of 1280×1024 pixels, but since we are using the images for registration and the exact same area is going to be captured several times, the user is given the possibility to define an image spatial binning in both the horizontal and vertical directions, separately, thus potentially reducing the amount of information that the system has to handle during the flight mission.

III. PLATFORM DEVICES INTEGRATION

As it was mentioned before, this is the first work carried out to integrate a Specim FX10 into a Matrice 600, which is mainly intended and adapted for filming cameras. For this reason, quite a lot of efforts had to be taken in terms of the design of mechanical pieces and electrical connections, in order to create a proper functioning system.

Figure 2 shows a 3D layout of the mechanical elements designed and created with a 3D Ultimaker 3 Extended printer to fit the gimbal and replace some of the existing structural components. The following design considerations have been taken into account to replace these components:

- Camera degrees of freedom for position adjustment. It is critical that the system is properly balanced in off mode, so when the gimbal is turned on, the motors are not continuously correcting the deviations and hence, consuming extra energy, plus the risk of having vibrations. The designed components allow therefore to adjust the camera position in the x direction (flying direction), y direction (pointing right of the flight direction), and z direction (vertical axis).
- The camera is by default pointing downwards. Since the system is designed to scan fields beneath the drone, by default it makes sense to have the sensor already targeting the scanning surface.
- Weight optimization. It is important to keep the system as light as possible, so the components design provides a good trade off between weight and resistance to hold everything tight.

As it is seen in Figure 2, mainly the bottom and top original plates of the Ronin Mx have been replaced by new custom ones, keeping the original vertical rods that close up the whole structure. Since the original bottom plate contains the electrical ports and the gyroscope, the new custom made one has been developed to be able to contain these elements as well, which are critical for the functioning of the system. The custom designed plate to hold the camera grabbing it from the top also has the holes in place to fit the on-board system, i.e, the Jetson TK1 board. On top of this board
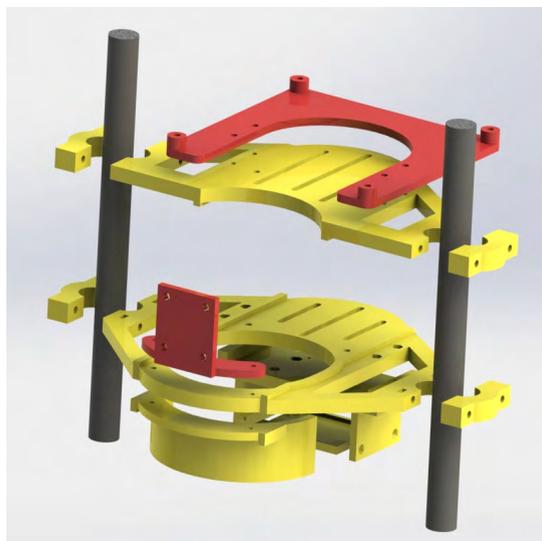
(a)
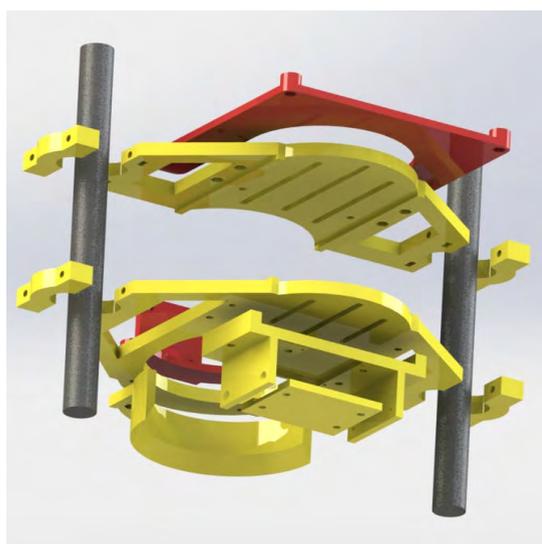
(b)

FIGURE 2. Gimbal construction elements breakout. (a) Front view. (b) Rear view.

another element has been designed and fixed into the same holes as the Jetson TK1, to bear the SSD board where images are stored after being captured.

The second aspect of the design integration are the electrical connections and communication links. The FX10 camera has to be supplied with an input voltage within the range $12\ V \pm 10\%$, and the Jetson TK1 board spans the limits into a slightly broader range $12\ V \pm 15\%$. The Ronin Mx provides two output connections out of the 4S Lipo 1580 mAh battery mounted in it and a voltage regulator that supplies at $13\ V$. This value has been measured for reassurance throughout the battery discharge cycle and it is kept constant. Since the $13\ V$ fulfills both camera and board requirements it is just a matter of making the physical connection through a cable. The output ports in the Ronin Mx are Dtap female, the input port in the Jetson TK1 is a standard 2.1mm DC

barrel plug and the input port in the Specim FX10 is a Fisher-type S1031-Z012-130+.

The communication cable between the FX10 and the Jetson TK1 is a standard Cat 6a ethernet cable with a male RJ45 connector on the board end and a male M12 connector on the camera end. Finally the serial connection between the Jetson TK1 and the A3 controller from the Matrice 600 has been implemented with a USB-TTL converter connected to the board and the jumpers connected to the A3. The reason why a USB-TTL solution was used instead of the serial port available in the GPIOs of the Jetson TK1 has been mainly the voltage level difference from both. The Jetson TK1 serial output voltage level on the GPIOs is $1.8\ V$ and the one on the A3 works at a standard value of $3.3\ V$.

FIGURE 3. UAV flying platform.

Figure 3 shows the system totally assembled and flying during one of the flight campaigns performed the 24th of July 2018 in Tejeda, Gran Canaria, Spain.

IV. APPLICATION FOR CONTROLLING THE SYSTEM

In this section a detailed description of the developed application for controlling the system is presented. The implemented software makes use of different SDKs for controlling the individual components. The main applications involved in the whole process are enumerated below:

- End-user application running on an iOS device, programmed in Objective C and based on the DJI Mobile SDK [20].
- Flight control application running on the Jetson TK1, programmed in C++ and based on the DJI Onboard SDK [21].
- FX10 camera control application running on the Jetson TK1, programmed in C++ and based on the eBUS SDK from Pleora [22].
- IDS camera control application running on the Jetson TK1, programmed in C++ and based on the IDS uEye SDK [23].

A. END-USER APPLICATION

The end-user application purpose is triple. On one side, it receives the basic inputs from the user, thanks to a graphical interface, to perform the first flight mission calculation. After these calculations are completed, the mission waypoints
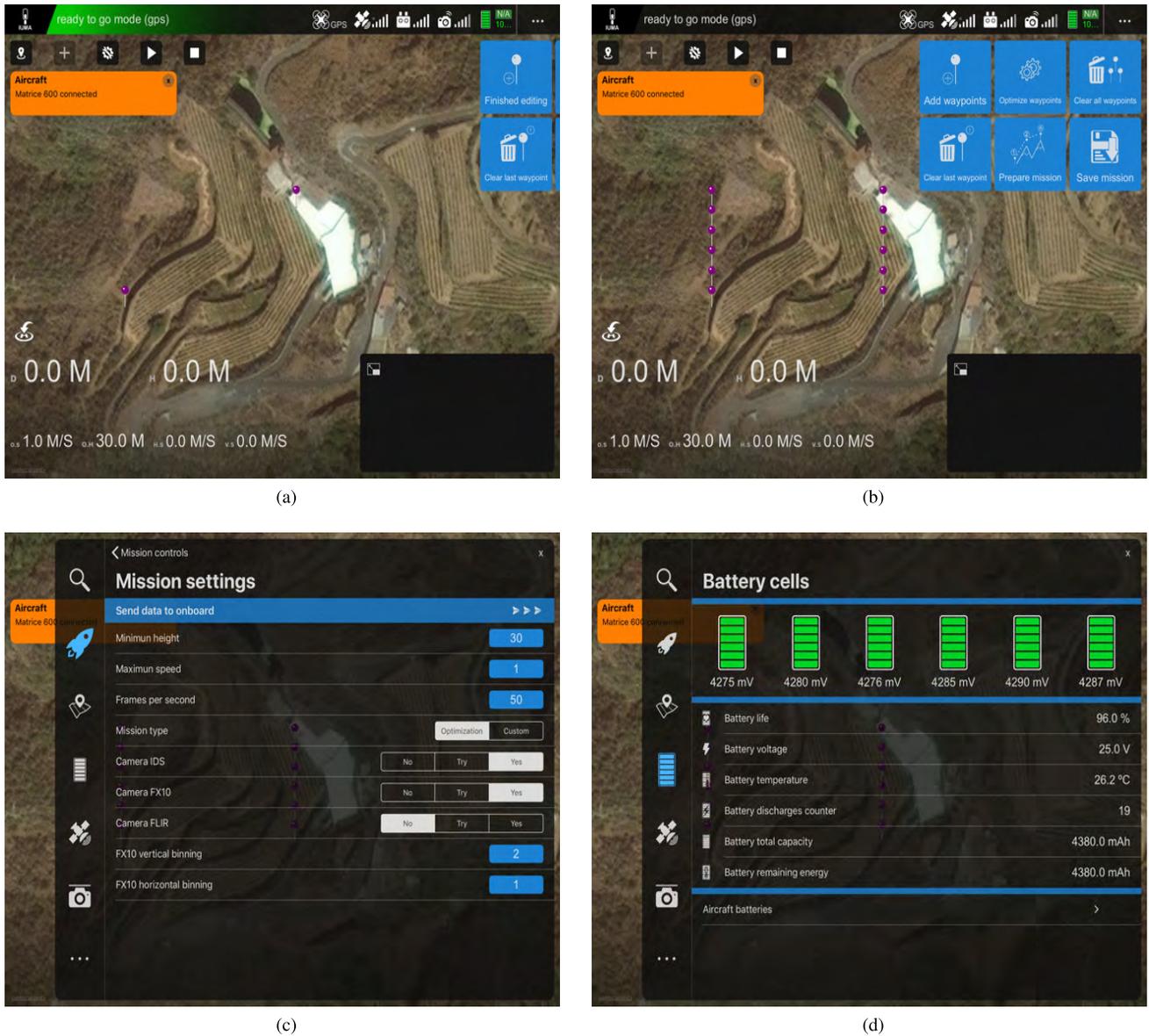
FIGURE 4. iOS application screenshots. (a) Input initial coordinates. (b) Waypoint layout. (c) Mission parameter setup. (d) System parameter overview.

are presented on the map, defining the flight swathes. Finally, during the flight, the end-user application communicates with the on-board application to provide valuable information to the user regarding its progress. Figure 4 shows a few snapshots taken from the end-user application. In Figure 4a, the user is defining the scouting area by selecting two corners in the diagonal of the rectangle. Figure 4b lays out the mission waypoints after having applied all user defined parameters. Figure 4c shows the user mission parameters window and Figure 4d indicates the status of some of the system variables, in this particular case, the charging state of the batteries.

1) USER INPUTS

The first step is to define the area to be inspected by the drone. The application allows the user to provide this area in two different ways: by selecting the two corners in the diagonal of the rectangle (see Figure 4a); or by directly selecting all the corners of an irregular area, in which case the implemented code will calculate the rectangle with minimum size including that area.

Next, the user defines some required information to be able to perform the calculations and start the mission. For instance, the sensors to be enabled and capturing during the mission, the camera sensor binnings and two out of the three main mission parameters:

- Relative height, h, from the ground
- Speed
- Frames per second, *FPS*

These parameters are linearly dependent, so that the third one not defined by the user is obtained based on the
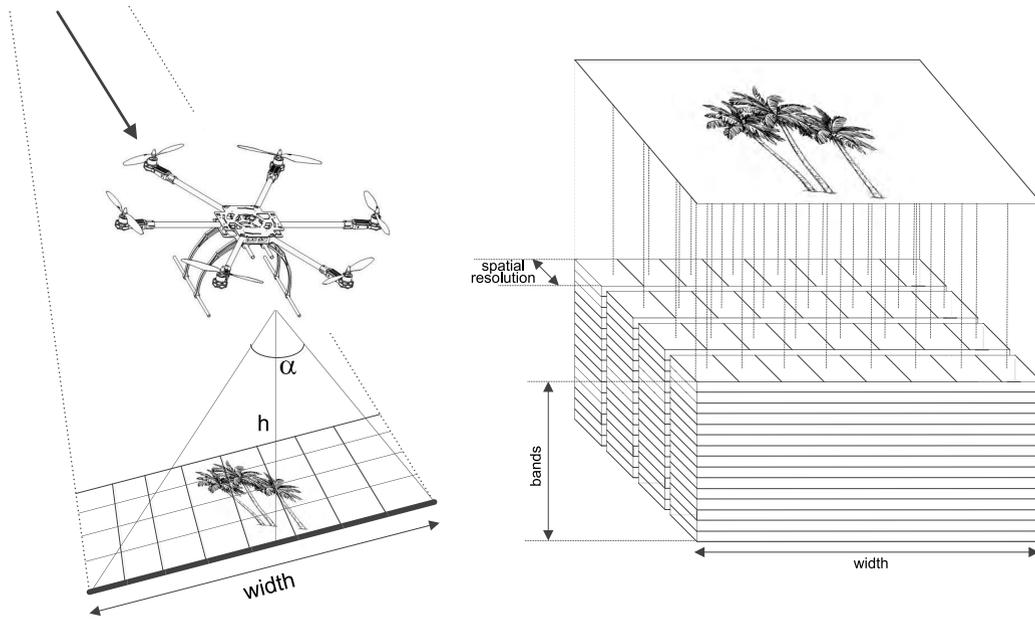
FIGURE 5. UAV flight schematics.

following relationships:

$$resolution = \frac{spatial\_sampling}{2 * h * \tan \frac{\alpha}{2}} \quad (1)$$

$$speed = resolution \times FPS \quad (2)$$

Equation 1 defines the resolution based on the camera and flight parameters, such as the camera *spatial_sampling* and field of view α, and the flight height h. The *spatial_sampling* depends on the hyperspectral camera spatial binning. The default value for the binning is 1, which means that 1024 pixels will be captured per frame.

Figure 5 shows the flight parameter representation.

2) WAYPOINT LAYOUT

After the user has defined all inputs and the remaining parameters have been obtained by the application, it is time to lay down the mission waypoints (Figure 4b), for which again the resolution obtained in Equation (1) is essential together with the overlap between different swathes. The calculations performed by the application are explained in the following four steps:

1) First, the rectangle sides are obtained in meters based on the GPS coordinates of the corners, and the shortest distance is selected to lay down the waypoints in between those corners. In this way, each individual flight swath has a bigger length and therefore, the flying time is reduced.

2) Second, the image width in meters is obtained by multiplying the resolution by the spatial sampling of the camera.

3) Third, the image width is corrected by the user input overlap: $width' = width * overlap$

4) Finally, the number of waypoints to be added in between the corner waypoints is calculated.

$$N_{waypoints} = (\frac{side\_distance}{width'} - 1) \quad (3)$$

where *side_distance* is the distance in meters of the shortest rectangle side. It is obvious that this calculation not always provides an integer number, so it is rounded up to the next integer value, giving as a result a larger overlap between images than what was initially defined.

Based on all the mentioned inputs, the application defines a rectangle and the intermediate waypoints to cover the area fulfilling the user requirements. It does also take into account the remaining battery percentage and in case the calculated remaining flight time is less than the estimated time to complete the mission, it warns the user and suggests to increase the altitude or the speed.

Once the calculation is consolidated, it is sent together with the defined user input parameters to the application running on the Jetson TK1, making use of the built-in functions in both the Mobile SDK and Onboard SDK from DJI and a specific protocol running on top of those SDKs defined for this purpose.

B. FLIGHT CONTROL APPLICATION

As it has already been mentioned, the application running on the Jetson TK1 uses the Onboard SDK from DJI for the purpose of implementing the drone flight control and the communication with the device on the ground. The application follows the steps described next.

1) The first step of the on-board application, after the board has successfully initialized, is to start the listening task implemented with a callback function that is constantly listening messages from the mobile application and implements the mentioned protocol in order to receive the mission setup parameters.

2) Once all parameters and required information have been successfully transferred and after the user has hit the *start* button, the listening task is closed and the main task together with the camera capture tasks are started and run in parallel. The main task, responsible for the drone flight control, is also in charge of collecting telemetry data.

   a) The take-off command is issued, followed by the drone elevating itself up to 2 meters above the ground, and hovering in that position.

   b) Then the function Z-movement is called with the relative height above the ground given by the user as an input.

   c) At this point, when the UAV has reached the user defined height, the RGB sensor is calibrated in order to obtain good quality images. For this purpose, images are being automatically captured while adjusting the sensitivity and exposure time until a defined pixel average value range has been reached.

   d) Afterwards the program enters into the waypoint loop, consisting in first orientating towards the next waypoint, calling the yaw control function.

   e) Once the drone is oriented, it starts moving towards the next waypoint. It must be pointed out that a relative coordinate system is being used, with distances defined in meters and with the system reference fixed at the drone starting position. Since the waypoint coordinates are defined in UTC, they have to be converted to the new system. Another point to be taken into account is the fact that for the hyperspectral capturing process it is crutial that images are captured at a constant speed.

   f) The waypoint application is running inside a control loop revisited every 5*ms* that is constantly checking whether the next waypoint coordinates have already been reached. Some measures against overshoot have also been implemented in order to make the drone stop as closest to the waypoint as possible.

   g) If it is the last point, the drone moves back to its initial position. Otherwise steps d to f must be repeated until the last waypoint is reached.

   h) Once the drone is located back in its initial coordinates (where the system reference axes were located), Z-movement function is called again to descend and finally the drone lands, completing the whole mission in a fully automatic fashion.

3) After the flight is over, the board gathers all data and copies them to the SSD disk, closes up the flight control application and restarts again the listening function, so the user can perform another mission or simply shut down the system.

It is important to highlight that after initialization, each task in the board runs in parallel in a different thread and all running threads are executed in the four cores available at the board CPU in an optimal manner to allow a smooth system overall functioning. The drone flight control is in charge of implementing the synchronization between itself and the camera capture tasks. After the drone has reached the first waypoint of an even swath the main task will trigger the capturing start of all camera sensors which internally are in charge of performing the capture and saving it in memory. When the drone reaches the next waypoint (end of the swath) the main task issues a stop command to the camera tasks. Additionally, the main task will store telemetry data in memory during those swathes, synchronized with the camera tasks. In this way only in the desired sectors data is being captured and the whole system performance and memory space is optimized. Finally once the drone has reached the end of the mission the main task will issue the command to stop the capturing process and close the camera tasks.

The entire application has been implemented in a modular way, in order to allow future inclusion of additional sensors. This means that the camera tasks have been developed sharing a common interface, so the main task interaction with any camera already mounted and/or to be mounted on the drone is exactly the same. This interface includes parameters such as frame rate, exposure time, horizontal/spatial and vertical/spectral binning.

An important remark in the described process is the use of more than one thread inside the camera tasks, just to avoid that once the next capture cycle has arrived and the process being executed in the previous thread has not yet finished, there is no system delay and images are properly stored in memory. A memory buffer with a capacity of up to 32 frames has been defined so that each thread can handle a different captured frame. The images are all saved with an index that identifies when it was captured so there is no misalignment once the whole mosaic is to be reconstructed.

For inter-task communication and synchronization purposes, a shared memory has been created allowing the exchange of parameters between the processes being executed.

V. DATA ACQUIRED BY THE DEVELOPED UAV SYSTEM

The purpose of the construction of the proposed platform is to use it in the precision agriculture domain. Being more specific, the goal is to perform several flight campaigns over a vineyard in the island of Gran Canaria. Figure 6 shows the exact location of the vineyard in a village of the island called Tejeda, and the two Google Maps pictures of the

(a)

(b)

(c)

(d)

FIGURE 6. Flight location. (a) Canary islands overview, (b) Gran Canaria overview highlighting the area of Tejeda where the scouted terrains are located, (c) terrain 1, and (d) terrain 2.

terrains under analysis. The exact coordinates of Terrain 1 are 27°59'35.6''N 15°36'25.6''W and the coordinates of Terrain 2 are 27°59'15.2''N 15°35'51.9''W.

Results of one of the flight campaigns performed over Terrain 1 are shown in Figure 7. This figure displays a false RGB representation extracted from the hyperspectral data acquired for each swath. The flight was performed at a height of 45 *m* over the ground, a speed of 4.5 *m/s* and the hyperspectral camera capturing frames at 150 *FPS*. The flight mission consisted of 12 waypoints which provides 6 swathes as it is represented on Figure 7. The number of frames per swath is between 4100 and 4200, what results in approximately 1.9 GBytes per swath. At this height, the ground sampling distance in line and across line is 3 *cm*, what gives a total of 125 *m* length and 31 *m* width coverage per swath. The mission took approximately 12 minutes to complete from take-off until land.

Results of one of the flight campaigns performed over Terrain 2 are shown in Figure 8. This figure displays a false RGB representation extracted from the hyperspectral data acquired for each swath. The flight was performed at a height of 45 *m* over the ground, a speed of 6 *m/s* and the hyperspectral camera capturing at 200 *FPS*. The flight mission consisted of 10 waypoints which provides 5 swathes as it is represented on Figure 8. The number of frames per swath is between 4900 and 5000, resulting in approximately 2.3 GBytes per swath. At this height, the ground sampling distance in line and across line is 3 *cm*, what gives a total of 150 *m* length and 31 *m* width coverage per swath. The mission took approximately 9 minutes to complete from take-off until land.

A. FLIGHT CONTROL ACCURACY
One of the main goals of the proposed platform is to fine control the flight in order to capture high quality hyperspectral images and reduce the post-processing efforts that have to be carried out performing geometric and radiometric corrections.

The designed software controls position and speed on every loop that runs every 5 *ms* to obtain a stable flight at an almost constant speed, which is very important for the captured samples. The trajectory is constantly adjusted as well, keeping the drone on track throughout the whole mission. The system minimizes overshot in position, when reaching a waypoint, and in angle, when rotating. This has been achieved by defining extra control parameters that have been carefully selected through several tests, both in the provided DJI simulation environment and in real flights.

Figure 9 shows telemetry data captured in the flight campaign performed over Terrain 2. Figure 9a shows the ideal trajectory, in red, defined by the iOS application based on the area selected by the user, and the real one, in green. GPS longitude and latitude coordinate values have been converted to meters. In the representation, the initial point of the first sector has been set as the origin and the X axis corresponds to the north in the real world. The largest deviation, as it can be visually appreciated, occurs in sector 2 producing a maximum tangential distance between the ideal and real trajectories of approximately 2 meters. In Figure 9b the altitude variation

**FIGURE 7.** Captured swathes of terrain 1.
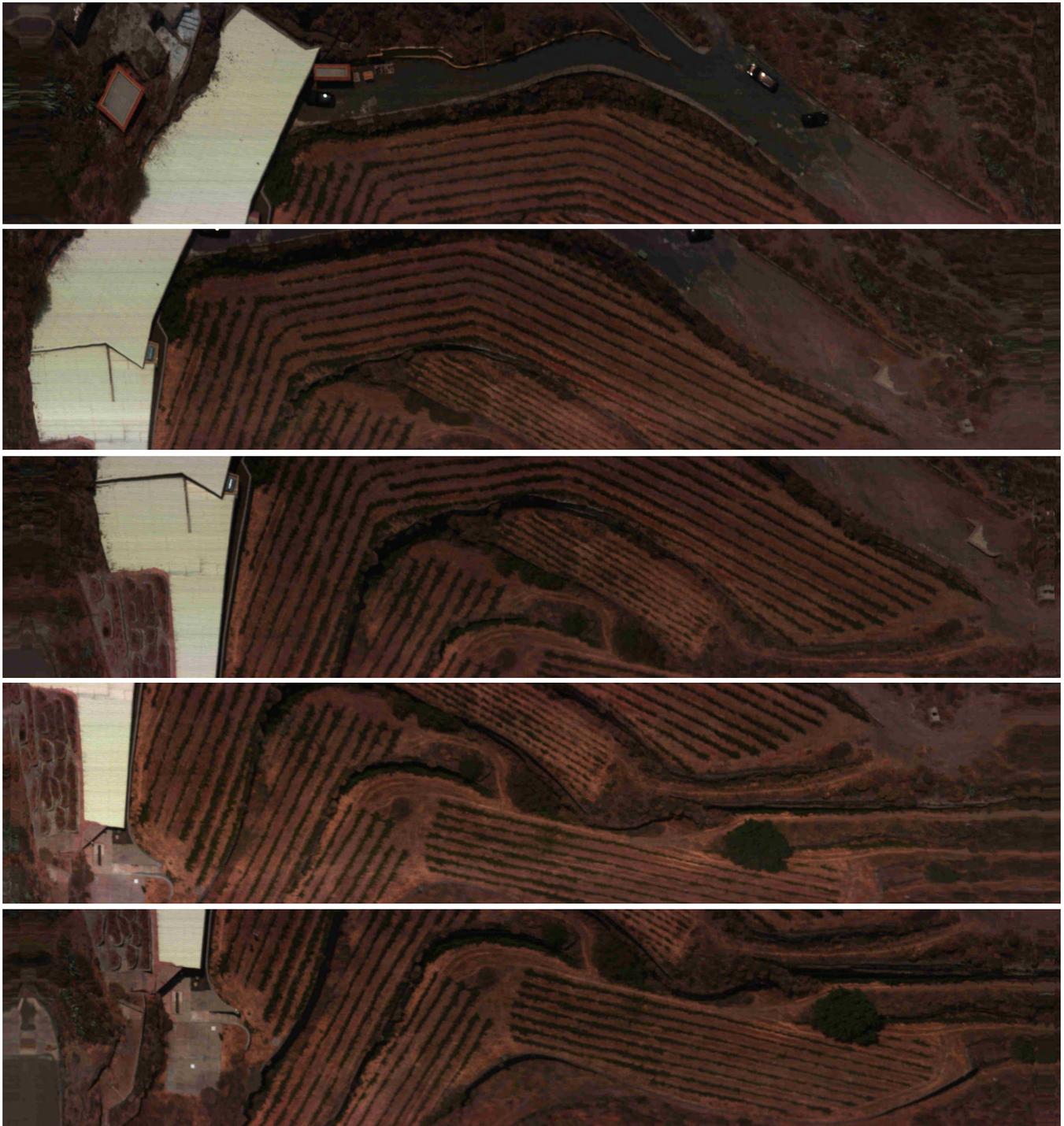
FIGURE 8. Captured swathes of terrain 2.

throughout the first sector of the flight mission is plotted. Altitude is measured over sea level, so in order to obtain the difference with the ground, the initial position altitude of the drone, 1227.5 m, is substracted.

Figure 9c shows the gimbal roll, pitch and yaw angles, again over sector 1 of the same mission. As it is seen, the yaw angle starts with a considerable deviation due to the abrupt rotation the drone does when it reaches the waypoint and

orientates itself to the next waypoint. This deviation is slowly corrected by the software until it reaches an offset, left on purpose to compensate the slight misalignment of the camera relative to the drone horizontal line. The absolute speed of the drone over sector 1 of the flight mission is plotted in Figure 9d. It takes the platform a couple of seconds to achieve the set speed by the user and then it is kept constant until the next waypoint is reached.
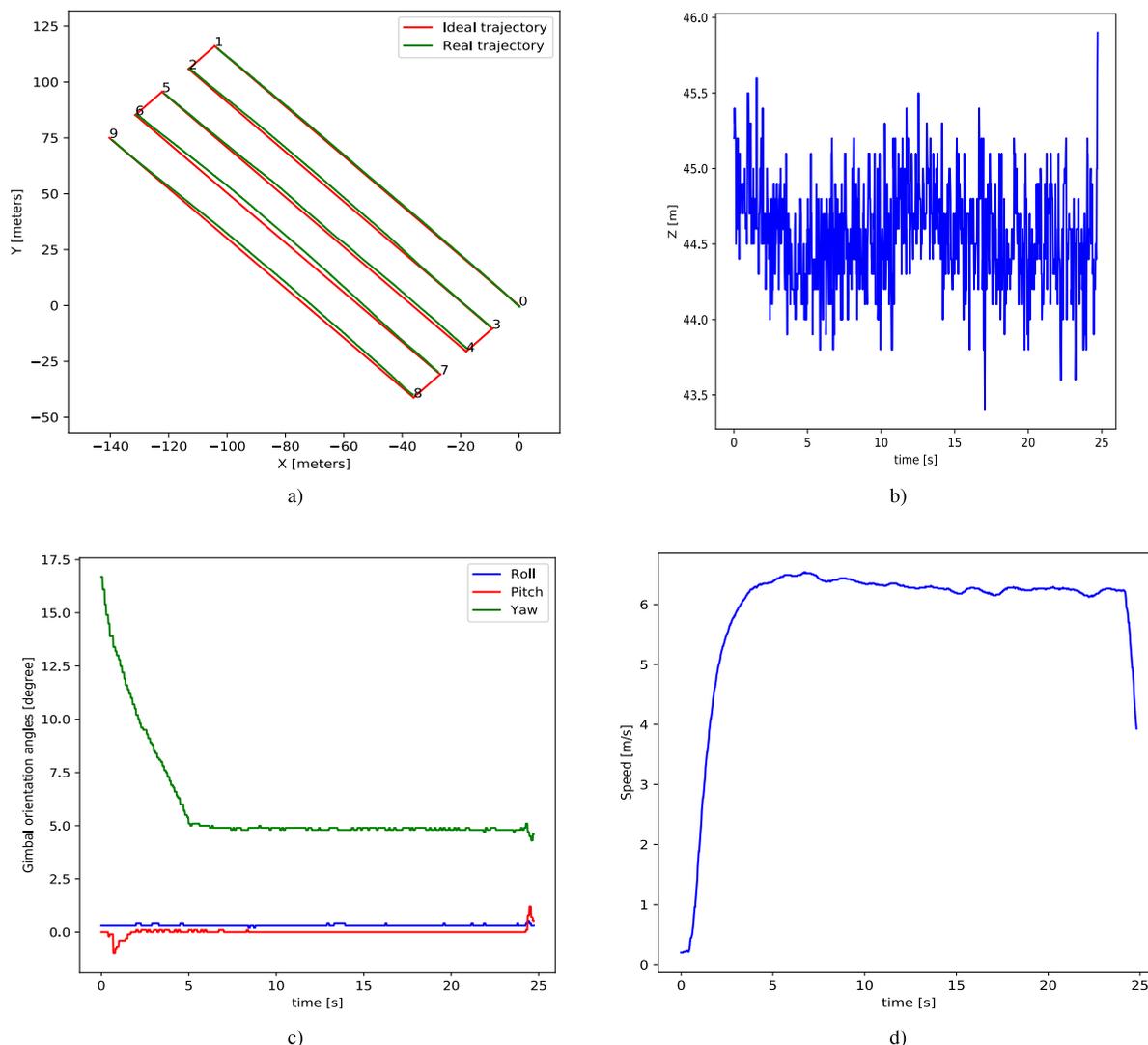
FIGURE 9. Flight data for mission 2. (a) Drone trajectory. (b) Altitude above ground variation over sector 1. (c) Gimbal orientation angle variations over sector 1. (d) Absolute speed variation over sector 1.

The results presented on Figures 9c and 9d suggest that the first couple of captured seconds in the sector should be cropped out since they are going to be more distorted than desired. However, the rest of the flight looks very stable and therefore the quality of the images is high. Those images were captured in good weather conditions with relatively low wind speed. This is of course not always the case and that is the reason why the platform incorporates an industrial RGB camera for image registration and to improve accuracy. The system does also capture telemetry data as the one shown in Figure 9, being the main drawback the low sampling rate compared with acquisition rate of the camera in terms of frames per second (*FPS*). In this particular case, while the camera captures images at 200 *FPS*, the telemetry system is only able to obtain data at 50 *Hz*.

VI. HSI DATA PROCESSING

This section highlights the main benefits of having a hyperspectral sensor on-board a flying platform. Similar platforms

based on such sensors have already detailed some possible processing applications that can be carried out with the captured data [24]–[26]. The novelty of this work is the capability of the system to do on-board processing in the Jetson TK1 and provide the user some results while the flight is still ongoing or has recently ended.

A. IMAGE CALIBRATION

The raw captured frames by the sensor are a measurement of the sensed light per sensor pixel, each pixel value expanding from 0 to 4096 considering the camera pixel depth. However, sensor response is not uniform across the covered spectral range. The consequence is that, even if the same amount of radiance is hitting all the pixels of the sensor, the digital value measured in each pixel will be different, especially for different wavelengths. Additionally, the illumination conditions may not be uniform across the covered spectral range. These facts make not possible to directly use the raw images, which are affected by the sensor response,

Spatial dimension. Pixels 1 to 1024

Spectral dimension. 400-1000 nm.
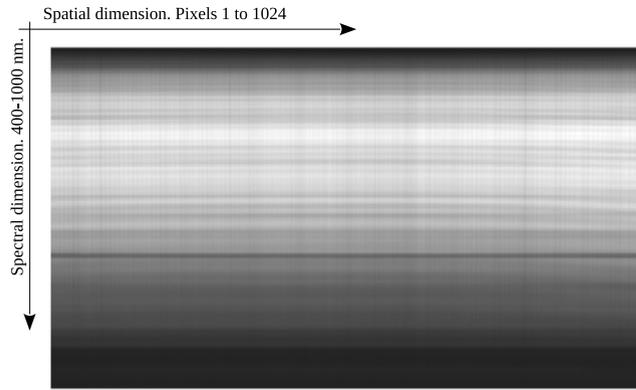
FIGURE 10. Spectral response of the Specim FX10 hyperspectral camera.

for the subsequent hyperspectral imaging applications. For instance, Figure 10 displays a captured hyperspectral frame over a certified Zenith Polymer white panel that reflects more than 99% of the incident radiation in all the VNIR range [27]. As it can be observed in this image, the sensor varies with the spectral wavelength. Additionally, the response of different pixels that measure the same spectral wavelength also varies, what typically results in stripping noise [28].

In order to solve these issues, the captured images are converted to reflectance values, in such a way that each image value is scaled between 0 and 1, representing the percentage of incident radiation that the scanned object reflects at each specific wavelength. The procedure for doing so is explained next. Prior to the mission flight, an image of a Zenith Polymer white calibration panel which is certified to reflect more than 99% of the incident radiation in the VNIR spectral range is acquired from the ground, as the one shown in Figure 10, just using the gimbal with the camera pointing downwards at a distance of around 1 meter. It is important to make sure that the camera line of pixels is entirely sensing the white panel and no pixel is left out. Up to 50 samples at the exact same frame rate as the one used during the flight were taken and then averaged to have a final reference that is used for calibration. A dark sample is also required for calibration. This dark reference collects the minimum values that the sensor measures when no radiance is hitting it. In order to obtain the dark reference, the camera lens is completely closed. Again, 50 samples are taken and then averaged.

$$\text{reflectance} = \frac{\text{sensed\_bitarray} - \text{dark\_refence}}{\text{white\_reference} - \text{dark\_reference}} \quad (4)$$

Equation 4 shows how the raw data is calibrated for obtaining its corresponding reflectance values. Figure 11 shows an example of how a set of real hyperspectral signatures collected by the described acquisition system are calibrated using the described procedure. In particular, Figure 11a shows the average value of the white and dark references across all the sensor pixels. Figure 11b shows a portion of one real hyperspectral image collected by the system developed in this work, from which some pixels have been selected, corresponding to vegetation, soil and shadows, marked in the

image in green, blue and red colors, respectively. The raw and calibrated spectral signatures of these pixels are displayed in Figures 11c and 11d. As it can be observed in these graphs, the raw spectral signatures are strongly affected by the sensor spectral response.

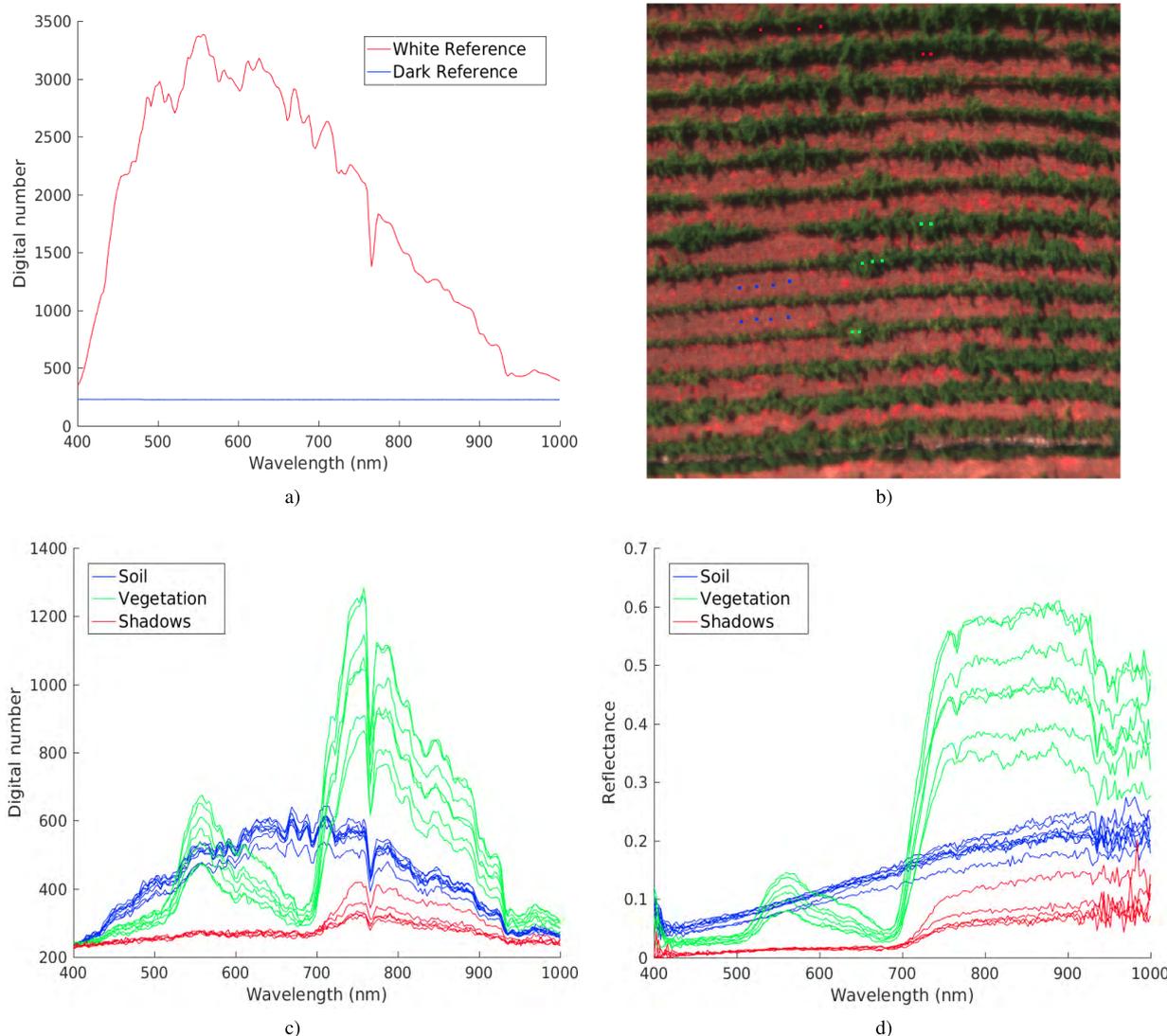### B. VEGETATION INDICES CALCULATION

One of the direct results that can be obtained with the captured images is the generation of a set of VIs able to provide information about the status of the crop [29]. This is performed by combining or transforming two or more spectral bands designed to enhance the contribution of vegetation properties, allowing reliable spatial and temporal inter-comparisons of terrestrial photosysthetic activity and canopy structural variations. There are multispectral sensors specifically developed for smart farming applications that efficiently collect images in some of the most widely used spectral channels for calculating VIs, such as the Rededge multispectral camera from Micasense [30]. Nevertheless, the number of spectral channels collected by this kind of camera is strongly limited and so is the amount of VIs that can be correctly calculated. In this sense, the fact of carrying a hyperspectral scanner provides the advantage of being able to calculate any VI whose involved bands are within the VNIR spectral range. Additionally, any other kind of index [31], not necessarily oriented to smart farming applications, can also be calculated, thus increasing the overall applicability of the developed acquisition system.

As an example, some very well known indices, whose formulas are detailed in Table 5, have been calculated for a real hyperspectral image of 1024×1024 pixels that corresponds to a portion of the first swath in the hyperspectral data acquired over Terrain 1. These indices are graphically displayed in Figure 12. Concretely, the well-known NDVI [32], which quantifies vegetation by measuring the difference between near-infrared (which vegetation strongly reflects) and red light (which vegetation absorbs), has been measured and displayed in Figure 12b. Additionally, the Modified Soil-adjusted Vegetation Index (MSAVI) [33] shown in Figure 12c and the Modified Chlorophyll Absorption Ratio Index (MCARI) [33] shown in Figure 12c have been calculated. The first one focus on automatically adjusting the NDVI when applied to areas with a high degree of exposed soil surface. The second one measures the relative abundance of chlorophyll.

In the VIs shown in Figure 12, already four different wavelengths are involved in the calculation. Many other indices can be obtained as indicated by [33], involving several other spectral wavelengths, and hence the need for a sensor that captures more than just a few bands.

C. OTHER RESULTS

In addition to the calculation of different indices, the hyperspectral data collected by the system developed in this work is potentially useful for many other applications that may benefit from being able to capture hyperspectral data

FIGURE 11. Pixel calibration for the real acquired hyperspectral data. (a) White and dark signatures. (b) Selected pixels. (c) Raw signatures. (d) Calibrated signatures.

**TABLE 5.** Vegetation indices used in this manuscript.

VI	Equation
NDVI	$(R_{800} - R_{670})/(R_{800} + R_{670})$
MSAVI	$\frac{1}{2}\left[2 * R_{800} + 1 - \sqrt{(2 * R_{800} + 1)^2 - 8 * (R_{800} - R_{670})}\right]$
MCARI	$\left[(R_{700} - R_{670}) - 0.2 * (R_{700} - R_{550})\right] * (R_{700}/R_{670})$

from an UAV. Some of them are focused on mineralogy applications [34], [35], security and/or surveillance applications [36] or target detection and/or tracking applications [37]. Tasks such as classification, spectral unmixing, anomaly detection or clustering are usually employed in this kind of hyperspectral imaging applications [38]–[40]. Just to provide a brief example of the results that can be obtained when applying these processes to the hyperspectral data that can be acquired by the proposed system, the spectral sig-

natures displayed in Figure 11d, that correspond to vegetation, soil and shadows, have been used to train a Support Vector Machine (SVM) classifier. In particular, a 1vs1 linear kernel has been employed. The trained SVM model has then been used for classifying the 1024 × 1024 image portion previously described, obtaining the classification results shown in Figure 13b. Blue represents the pixels classified as soil, green is used for the vegetation and red for the shadows.

D. ON-BOARD REAL-TIME DATA PROCESSING

The possibility of using the GPU included in the on-board PC for accelerating the execution of different processes represents an important advantage with respect to other UAV systems. This GPU is the GK20A, based on a Kepler architecture, with 195 CUDA Cores and a maximum of 2048 resident threads per multiprocessor. As it can be intuited, these
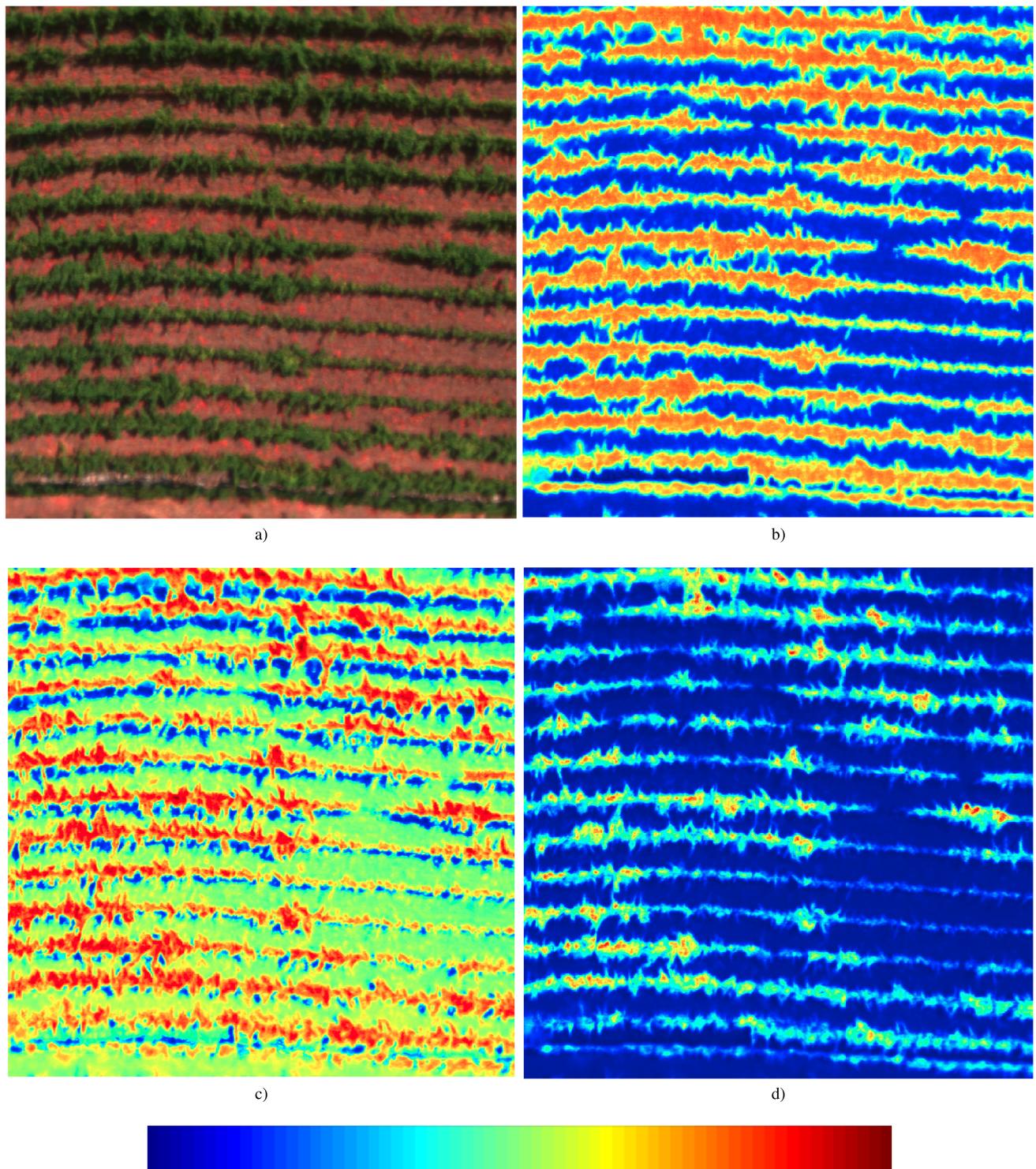
FIGURE 12. Calculated indices for a portion of the real hyperspectral image collected by the described system over terrain 1, first swath, of size 1024×1024 pixels. Lowest values are represented in blue and highest values in red. (a) RGB representation. (b) NDVI. (c) MSAVI. (d) MCARI.

characteristics fit very well our problem at hand, since we have 1024 hyperspectral pixels per captured line, which is within the limit of maximum concurrent threads that can be launched, and also a multiple of 32, which is the warp size and also the number of execution task packets.

In particular, we have exploited this advantage in two different ways. On one side, for obtaining results in real-time regarding the different VIs that can be directly sent to the user for visual inspection, and, on the other side, for real-time compression of the hyperspectral data in such a way that it

a)   b)
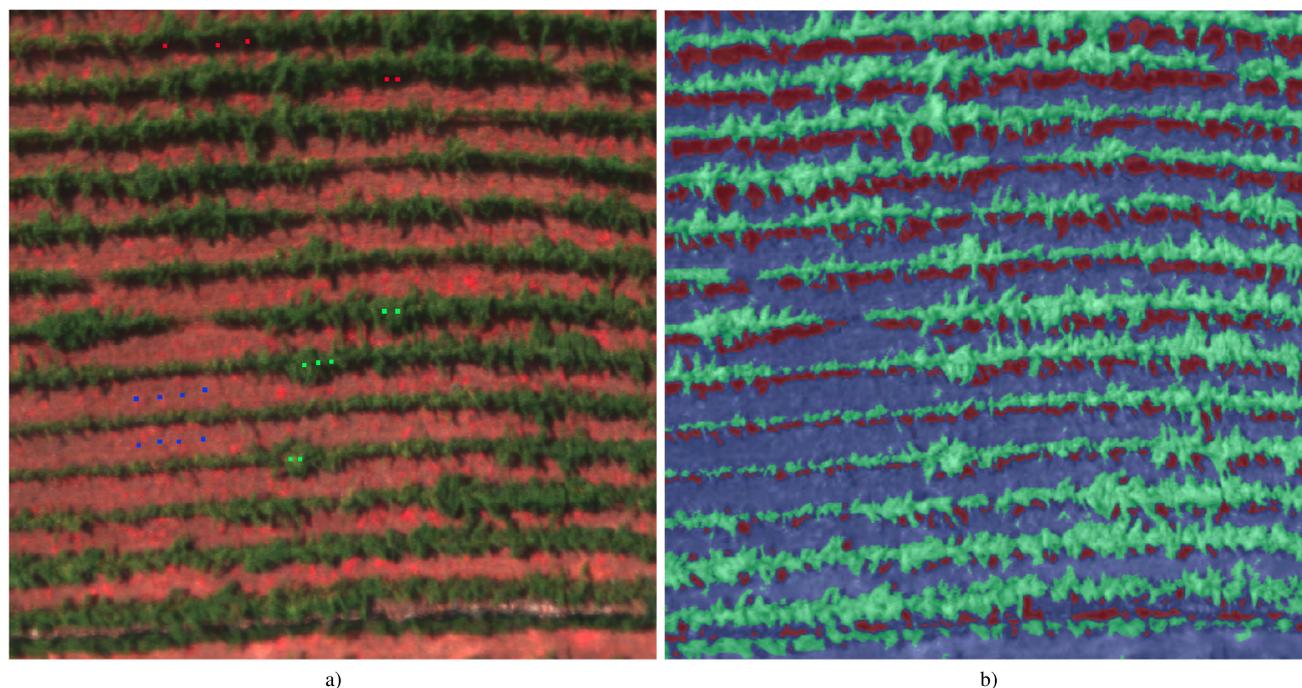
FIGURE 13. Hyperspectral imaging classification example using a 1vs1 linear SVM and the real hyperspectral data collected by the described UAV system. The classification map shown in (b) represents the three elements using the colors: Blue for the soil, green for the vegetation, and red for the shadows. (a) Pixels used for training the SVM classifier. (b) Classification map.

can be easily transferred to the ground station for its further processing.

1) VEGETATION INDEX MAPS CALCULATION IN REAL-TIME

The acceleration of the calculation of different VIs by using the aforementioned GPU is a relatively straightforward process. This process consists in independently applying the corresponding vegetation index equation to each of the 1024 hyperspectral pixels per captured line. Additionally, since the RAM memory of the Jetson TK1 is actually common to the board CPU (host) and the GPU (device), it can be used very efficiently as unified memory in CUDA. By doing so, the data transfers between host and device are avoided, reducing the required computational time.

Additionally, in order to obtain correct vegetation index values, the spectral bands of the pixel involved in the calculation of the vegetation index need to be calibrated using the white and dark references, as described in the previous section. This process is also independently applied to each of the 1024 hyperspectral pixels per captured line in the same manner as the vegetation index calculation.

2) REAL-TIME HYPERSPECTRAL DATA COMPRESSSION

In addition to the vegetation index calculation, more processes can be applied to the acquired hyperspectral data that may be of interest for different hyperspectral imaging applications, such as classification, target detection or anomaly detection. In general, the complexity of these processes, as well as the complexity of the involved mathematical

algorithm, is high. In order to provide a possible solution that allows the execution of this kind of processes for obtaining real-time or near-real time results, we have decided to compress the acquired hyperspectral frames on-board in real-time, in such a way that they can be rapidly transferred to a ground station in a streaming fashion for their further processing. For such purpose, the HyperLCA compressor [13] has also been implemented in the Jetson TK1, taking advantage of its GPU and the CUDA programming model for achieving real-time compression results.

As described in Section III, the acquisition data rate of the Specim FX10 hyperspectral camera is up to 100 Mbytes per second, what results in almost 6 Gbytes per minute. Accordingly, a 10 minutes flight could result in more than 50 GBytes. Due to this reason, the size of the acquired data has to be drastically decreased for being able to rapidly transfer it, specially if real-time transmission is desired. In particular, the hyperspectral images described in Section V were collected at 150 and 200 FPS using 2 bytes per pixel and band, producing 65.6 and 87.5 Mbytes per second for the data acquired over the Terrain 1 and 2, respectively. Their size is about 1.9 Gbytes per swath for the images of Terrain 1, and 2.3 Gbytes per swath for the images of Terrain 2.

The selected HyperLCA compressor is a lossy compressor specifically designed for independently compressing each of the hyperspectral frames collected by a pushbroom scanner in a fast manner. Additionally, it guarantees high compression ratios at a reasonable low computational complexity while keeping the image information that is potentially more useful

for the subsequent hyperspectral imaging applications. In particular, its computational complexity decreases for higher compression ratios (smaller compressed images), allowing to compress the acquired data faster when higher compression ratios are desired. As it can be noticed, these characteristics fit very well with the needs of the described system. This compressor has been implemented in the Jetson TK1 making used of custom developed kernels and the CUDA programming model, for taking advantage of the inherent parallelism present in the operations of the algorithm. Additionally, extra parallelization strategies were employed in order to pipeline the execution of the different stages of the compressor, namely spectral transform, preprocessing, mapping, and coding.

The developed implementation of the HyperLCA compressor in the Jetson TK1 has proven to achieve real-time compression for acquisition rates higher than 300 FPS for all the tested configurations of the compressor. In particular, this compressor has 3 main input parameters, the minimum desired compression ratio, CR, the block size, BS, which corresponds to the number of hyperspectral pixels per frame, and the amount of bits used for representing the output projection vectors obtained after the transformation stage of the compressor, N_{bits}. In this work, BS has been fixed to 1024 since this is the size of the hyperspectral frames captured by the FX10 camera. The CR parameter has been set to 12, 16 and 20, which guaranties that the compressed hyperspectral frames are at least 12, 16 and 20 times smaller than the original ones. The N_{bits} has been set to 12 and 8. The results obtained for each configuration are displayed in Table 6.

TABLE 6. Compression performance of the HyperLCA implementation developed for the Jetson TK1, where *BS* is the block size, N_{bits} the number of bits used to represent the output projection vector, and *CR* the compression ratio applied.

Input parameters			Maximum compression frame rate (FPS)
BS	N_{bits}	CR	
1024	12	12	495
		16	603
		20	697
1024	8	12	386
		16	473
		20	565

VII. CONCLUSION

Presently there is no doubt about the importance of including advanced sensors in many of the most common devices developed by the industry. When combined with aerial vehicles to transport these devices, an enormous interest is drawn by many potential users that see a huge amount of applications where such platforms can contribute. However, it is not an easy task to adapt industrial sensors to flying platforms, mainly due to the challenges that represent critical aspects such as weight, size, power budget and connectivity. In this work, problems are detected and solutions are highlighted in the process of validating a hyperspectral flying platform since its conception. The drone includes a hyperspectral Specim FX10 camera, a precision GPS, a controller and an embedded board to interact with a modular flight control application. 3D pieces have been constructed in order to adapt these devices to the drone in an optimal manner. As a result, a solution is proposed for the particular use case of precision agriculture, based on capturing 224 spectral bands at up to 300 frames-per-second in the VNIR range and processing different vegetation indices, such as the well-known NDVI, MSAVI and MCARI, to reveal features of vineyard areas. Furthermore, a low complexity lossy compressor was included on-board together with parallelization strategies adopted to transfer data to a ground station, allowing to perform more complex tasks. The experience gained in this research facilitates the inclusion of other advanced hyperspectral sensors in the SWIR range, uncovering innovative opportunities in other promising applications such as surface mining, remote sensing, environmental contamination, and in general, any field which involves aerial inspection.

REFERENCES

[1] J. A. J. Berni, P. J. Zarco-Tejada, L. Suarez, and E. Fereres, "Thermal and narrow band multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 722–738, Mar. 2009.

[2] Z. Li, Y. Liu, R. Walker, R. Hayward, and J. Zhang, "Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform," *Mach. Vis. Appl.*, vol. 21, no. 5, pp. 677–686, Aug. 2010.

[3] A. Birk, B. Wiggerich, H. Bülow, M. Pfingsthorn, and S. Schwertfeger, "Safety, security, and rescue missions with an unmanned aerial vehicle (UAV)," *J. Intell. Robotic Syst.*, vol. 64, no. 1, pp. 57–76, 2011.

[4] K. Daniel and C. Wietfeld, "Using public network infrastructures for uav remote sensing in civilian security operations," 2011.

[5] T. Adão *et al.*, "Hyperspectral imaging: A review on UAV-based sensors, data processing and applications for agriculture and forestry," *Remote Sens.*, vol. 9, no. 11, p. 1110, 2017.

[6] E. R. Hunt, Jr., and C. S. T. Daughtry, "What good are unmanned aircraft systems for agricultural remote sensing and precision agriculture?" *Int. J. Remote Sens.*, vol. 39, nos. 15–16, pp. 5345–5376, 2018. doi: 10.1080/01431161.2017.1410300.

[7] J. Huang, H. Wang, Q. Dai, and D. Han, "Analysis of NDVI data for crop identification and yield estimation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 11, pp. 4374–4384, Nov. 2014.

[8] C. Rey-Caramés, M. P. Diago, M. P. Martín, A. Lobo, and J. Tardaguila, "Using RPAS multi-spectral imagery to characterise vigour, leaf development, yield components and berry composition variability within a vineyard," *Remote Sens.*, vol. 7, no. 11, pp. 14458–14481, 2015. [Online]. Available: http://www.mdpi.com/2072-4292/7/11/14458

[9] Specim. *Specim FX1 Series Hyperspectral Cameras.* Accessed: May 4, 2019. [Online]. Available: http://www.specim.fi/fx/

[10] DJI. *MATRICE 600 PRO.* Accessed: May 4, 2019.[Online]. Available: https://www.dji.com/matrice600

[11] *IDS Sensor UI-3140CP.* Accessed: May 4, 2019. [Online]. Available: https://en.ids-imaging.com/store/ui-3140cp-rev-2.html

[12] NVIDIA. *Jetson TK-1 Developer Kit.* Accessed: May 4, 2019. [Online]. Available: https://www.nvidia.com/object/jetson-tk1-embedded-dev-kit.html

[13] R. Guerra, Y. Barrios, M. Díaz, L. Santos, S. López, and R. Sarmiento, "A new algorithm for the on-board compression of hyperspectral images," *Remote Sens.*, vol. 10, no. 3, p. 428, 2018.

[14] H. Aasen, E. Honkavaara, A. Lucieer, and P. Zarco-Tejada, "Quantitative remote sensing at ultra-high resolution with UAV spectroscopy: A review of sensor technology, measurement procedures, and data correction workflows," *Remote Sens.*, vol. 10, no. 7, p. 1091, 2018. [Online]. Available: https://www.mdpi.com/2072-4292/10/7/1091

[15] J. Yue *et al.*, "Estimation of winter wheat above-ground biomass using unmanned aerial vehicle-based snapshot hyperspectral sensor and crop height improved models," *Remote Sens.*, vol. 9, no. 7, p. 708, 2017. [Online]. Available: https://www.mdpi.com/2072-4292/9/7/708

[16] R. Hruska, J. Mitchell, M. Anderson, and N. F. Glenn, "Radiometric and geometric analysis of hyperspectral imagery acquired from an unmanned aerial vehicle," *Remote Sens.*, vol. 4, no. 9, pp. 2736–2752, 2012. [Online]. Available: http://www.mdpi.com/2072-4292/4/9/2736

[17] A. Habib, Y. Han, W. Xiong, F. He, Z. Zhang, and M. Crawford, "Automated ortho-rectification of UAV-based hyperspectral data over an agricultural field using frame RGB imagery," *Remote Sens.*, vol. 8, no. 10, p. 796, 2016. [Online]. Available: http://www.mdpi.com/2072-4292/4/9/2736

[18] DJI. *RTK GNSS System*. Accessed: May 4, 2019. [Online]. Available: https://www.dji.com/d-rtk/info

[19] DJI. *Gimbal Ronin-Mx*. Accessed: May 4, 2019. [Online]. Available: https://www.dji.com/ronin-mx/info

[20] DJI. *DJI Mobile SDK*. Accessed: May 4, 2019. [Online]. Available: https://developer.dji.com/mobile-sdk/

[21] DJI. *DJI Onboard SDK*. Accessed: May 4, 2019. [Online]. Available: https://developer.dji.com/onboard-sdk/

[22] Pleora. *Pleora eBUS SDK*. Accessed: May 4, 2019. [Online]. Available: https://www.pleora.com/products/ebus-sdk/

[23] IDS. *IDS uEye SDK*. Accessed: May 4, 2019. [Online]. Available: https://en.ids-imaging.com/ueye-software-archive.html

[24] J. Suomalainen *et al.*, "A lightweight hyperspectral mapping system and photogrammetric processing chain for unmanned aerial vehicles," *Remote Sens.*, vol. 6, no. 11, pp. 11013–11030, 2014. [Online]. Available: http://www.mdpi.com/2072-4292/6/11/11013

[25] M. Kanning, I. Kühling, D. Trautz, and T. Jarmer, "High-resolution UAV-based hyperspectral imagery for LAI and chlorophyll estimations from wheat for yield prediction," *Remote Sens.*, vol. 10, no. 12, p. 2000, 2018. [Online]. Available: http://www.mdpi.com/2072-4292/10/12/2000

[26] D. Turner, A. Lucieer, M. McCabe, S. Parkes, and I. Clarke, "Pushbroom hyperspectral imaging from an unmanned aircraft system (UAS)—Geometric processing workflow and accuracy assessment," *ISPRS–Int. Arch. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. XLII-2/W6, pp. 379–384, Aug. 2017.

[27] SphereOptics. *Zenith Polymer Diffusers*. Accessed: May 4, 2019. [Online]. Available: https://sphereoptics.de/en/product/zenith-polymer-diffusers/?c=79

[28] C. Rogass *et al.*, "Reduction of uncorrelated striping noise—Applications for hyperspectral pushbroom acquisitions," *Remote Sens.*, vol. 6, no. 11, pp. 11082–11106, 2014. [Online]. Available: http://www.mdpi.com/2072-4292/6/11/11082

[29] P. S. Thenkabail, R. B. Smith, and E. D. Pauw, "Hyperspectral vegetation indices and their relationships with agricultural crop characteristics," *Remote Sens. Environ.*, vol. 71, no. 2, pp. 158–182, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S003442579900067X

[30] Micasense. *Micasense Rededge-MX Camera*. Accessed: May 4, 2019. [Online]. Available: https://www.micasense.com/rededge-mx

[31] *Vegetation Index Database*. Accessed: May 4, 2019. [Online]. Available: https://www.indexdatabase.de/

[32] J. W. Rouse, R. H. Haas, J. A. Schell, and D. W. Deering, "Monitoring the vernal advancements and retrogradation of natural vegetation," NASA, Washington, DC, USA, Tech. Rep. E75-10354, NASA-CR-144661, RSC-1978-4, 1974.

[33] D. Haboudane, J. R. Miller, E. Pattey, P. J. Zarco-Tejada, and I. B. Strachan, "Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture," *Remote Sens. Environ.*, vol. 90, no. 3, pp. 337–352, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0034425704000264

[34] B. Martini, E. Silver, W. Pickles, and P. Cocks, "Hyperspectral mineral mapping in support of geothermal exploration: Examples from Long Valley Caldera, CA and Dixie Valley, NV, USA," Lawrence Livermore National Lab., Livermore, CA, USA, Tech. Rep. UCRL-PROC-202710, 2004.

[35] N. Yokoya, J. C.-W. Chan, and K. Segl, "Potential of resolution-enhanced hyperspectral data for mineral mapping using simulated EnMAP and sentinel-2 images," *Remote Sens.*, vol. 8, no. 3, p. 172, 2016. [Online]. Available: http://www.mdpi.com/2072-4292/8/3/172

[36] H. Ren and C.-I. Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1232–1249, Oct. 2003.

[37] D. Manolakis, E. Truslow, M. Pieper, T. Cooley, and M. Brueggeman, "Detection algorithms in hyperspectral imaging systems: An overview of practical algorithms," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 24–33, Jan. 2014.

[38] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, Mar. 2018.

[39] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[40] S. Matteoli, M. Diani, and G. Corsini, "A tutorial overview of anomaly detection in hyperspectral images," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 5–28, Jul. 2010.

PABLO HORSTRAND was born in Las Palmas de Gran Canaria, Spain, in 1986. He received the bachelor's degree in industrial engineering and electronics and control engineering and the master's degree in telecommunication technologies from the University of Las Palmas de Gran Canaria (ULPGC), Spain, in 2010 and 2011, respectively, where he is currently pursuing the Ph.D. degree in telecomunication technologies. He was with ABB Switzerland, the Minerals and Printing Department, the Drives Department, and last in the Traction Department, Aargau, Switzerland, from 2011 to 2017. In 2018, he was with the Royal Military Academy, Belgium, as part of the Ph.D. degree, where he was involved in research with the Signal and Image Centre Department.

RAÚL GUERRA was born in Las Palmas de Gran Canaria, Spain, in 1988. He received the Industrial Engineer degree from the University of Las Palmas de Gran Canaria (ULPGC), in 2012, the master's degree in telecommunications technologies from the Institute of Applied Microelectronics, IUMA, in 2013, and the Ph.D. degree in telecommunications technologies from the ULPGC, in 2017.

He was a Researcher with the Configurable Computing Lab, Virginia Tech University, in 2016. He was supported by the institute to do the Ph.D. research with the Integrated System Design Division. His current research interests include the parallelization of algorithms for multispectral and hyperspectral images processing and hardware implementation.

AYTHAMI RODRÍGUEZ was born in Las Palmas de Gran Canaria, Spain, in 1994. He received the degree in automatic and electronic engineering, and the master's degree in telecommunication technologies from the University of Las Palmas de Gran Canaria (ULPGC), Spain, in 2016 and 2017, respectively.

MARÍA DÍAZ was born in Spain, in 1990. She received the Industrial Engineering degree from the University of Las Palmas de Gran Canaria, Spain, in 2014, and the master's degree in system and control engineering from the Universidad Complutense de Madrid and the Universidad Nacional de Educación a Distancia (UNED), in 2017. She is currently pursuing the Ph.D. degree with the University of Las Palmas de Gran Canaria, where she is developing research activities with the Integrated Systems Design Division, Institute for Applied Microelectronics (IUMA). In 2017, she has conducted a research stay in the GIPSA-Lab, University of Grenoble Alpes, France. Her research interests include image and video processing, the development of highly parallelized algorithms for hyperspectral images processing, and hardware implementation.

SEBASTIÁN LÓPEZ (M'80–SM'15) was born in Las Palmas de Gran Canaria, Spain, in 1978. He received the Electronic Engineering degree from the University of La Laguna, San Cristobal de La Laguna, Spain, in 2001, and the Ph.D. degree in electronic engineering from the University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, in 2006.

He is currently an Associate Professor with the University of Las Palmas de Gran Canaria, where he is currently involved in research activities with the Integrated Systems Design Division, Institute for Applied Microelectronics. He has coauthored over 120 papers in international journals and conferences. His research interests include real-time hyperspectral imaging, reconfigurable architectures, high-performance computing systems, and image and video processing and compression.

Dr. López was a recipient of regional and national awards during the electronic engineering degree. Furthermore, he has acted as one of the Program Chairs of the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), in 2014, and the SPIE Conference of High Performance Computing in Remote Sensing, from 2015 to 2018. He was an Associate Editor of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, from 2008 to 2013. He is an Associate Editor of the *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *Remote Sensing* (MDPI), and the *Mathematical Problems in Engineering* journal. Moreover, he has been the Guest Editor of different special issues in JCR journals related with the research interests. He also serves as an active Reviewer for different JCR journals and as a Program Committee Member of a variety of reputed international conferences.

JOSÉ FCO. LÓPEZ received the M.S. degree in physics (specializing in electronics) from the University of Seville, Spain, where he was involved in the fields of high speed integrated systems. He has conducted the investigations with the Institute for Applied Microelectronics (IUMA), where he has been the Deputy Director, since 2009. He is currently a Lecturer with the School of Telecommunication and Electronics Engineering, ULPGC. He was with Thomson Composants Microondes, Orsay, France, in 1992. In 1995, he was with the Center for Broadband Telecommunications, Technical University of Denmark (DTU), Lyngby, Denmark. He was a Visiting Researcher with Edith Cowan University (ECU), Perth, WA, Australia, in 1996, 1997, 1999, and 2000. His current research interests include the fields of image processing, UAVs, and hyperspectral technology and their applications. He has been actively enrolled in over 40 research projects supported by the European Community, Spanish Government, and international private industries located in Europe, USA, and Australia. He has written around 140 papers in national and international journals and conferences.

● ● ●