IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 25 August 2023, accepted 20 September 2023, date of publication 26 September 2023, date of current version 3 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3319453

RESEARCH ARTICLE

Ground Level LiDAR as a Contributing Indicator in an Environmental Protection Application

CIPRIAN M. COMAN[1,2], BOGDAN C. TOMA[1], MIHAI A. CONSTANTIN[1], AND ADRIANA FLORESCU[2], (Senior Member, IEEE)

[1]Tesagon International SRL, 100029 Ploieşti, Romania

[2]Faculty of Electronics, Telecommunications and Information Technology, Politehnica University of Bucharest, 060042 Bucharest, Romania

Corresponding author: Ciprian M. Coman (ciprian@tesagon.com)

This work was supported in part by the European Commission through the Asociación de Empresas Tecnológicas Innovalia under HORIZON 2020 RESEARCH AND INNOVATION FRAMEWORK PROGRAMME under Grant DIH-World, and in part by Tesagon International SRL through the project ''Monitoring Environmental Parameters and Human Influence in Forest Areas (MEPHIFA).''

ABSTRACT The value attributed to the forests in developed countries has shifted from commercial wood production towards carbon reduction, habitat for wildlife, scenery and recreation. With the increasing perceived benefits came more concern and effort towards better forest management. There are many research and development initiatives aimed at effectively using the latest technologies to monitor forest environment, forest wildlife, forest incidents such as fires, illegal logging, poaching etc. This paper presents an expert system that we designed, developed and tested for forest environmental protection applications. The system includes remote sensing hubs and a central cloud server with several capabilities that are briefly presented. We detail a new approach of using Laser Imaging Detection and Ranging (LiDAR) sensors placed at ground level to acquire useful data from forest environments. Implementation details and considerations are presented together with test results obtained in a forest environment.

INDEX TERMS LiDAR, wireless sensor network, IoT, forest monitoring, real-time sensing.

I. INTRODUCTION

Forest exploitation is an activity as old as tools development in human history. With the increase of the population across the globe, sustainability and responsible logging have become more and more important. However, the forest areas are very large and difficult to monitor and protect from fraudulent activities. Recent technological advances present new opportunities for effective environmental monitoring and protection applications.

Sustainable forestry, or sustainable forest management, is the practice of managing forests to meet current needs and desires of society regarding forest resources, without compromising the availability of these for future generations. New understanding of forests benefits in highly developed countries, such as atmospheric carbon reduction, water yield, habitat for wildlife, scenery, recreation, and other values, have greater societal interest than commercial wood production [1]. At the same time, deforestation continues at alarming rates in some areas on the globe [2].

To protect forests from various threats such as fires or illegal deforestation, various detection systems such as infrared thermal cameras, LiDAR and Synthetic Aperture Radar (SAR), sensors have been implemented globally [3]. An early detection solution with a surface fire detection system based on a Radio Acoustic Sound System (RAAS) is a more recent approach for forest fire detecting using IoT, as presented in [4] and [5]. The widespread problem of illegal logging in Romania's forests motivated us to work on a technical solution, using existing technology in an innovative, integrated system. We designed, developed and tested an expert system for forest environmental protection applications. While the system includes several sensor types, in this paper we focus on our new approach of using LiDAR sensors placed at ground level to acquire useful data from forest environments and use it together with signals from the other sensors to identify undesirable situations in the forest. Using a distance mapping sensor was identified as a more feasible alternative

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

to using a video camera with infrared and image processing in the case of application with limitations such as compute power, available energy and network bandwidth. Another option considered is the use of ultrasonic sensors, having the advantages of being cheaper and faster, but at the same time having disadvantages such as lower precision and accuracy, shorter range and higher power consumption. In the case of our application, ultrasonic also suffers from the low reflective properties of the wood.

The current paper is organized as follows. Section II continues with a comprehensive review of the state of the art. Section III presents the expert system prototype that we designed and developed to monitor activities in forest areas, and details the sensor hub hardware and software, as well as the central hub architecture and implementation in the cloud. In section IV we present the tests carried out in a real forest environment, the problems encountered and applied solutions. Section V concludes the paper with discussion and suggestions for future research.

II. RELATED WORK

In [6], the authors have developed a type of wireless sensor network to prevent illegal deforestation using the evaluation of the acoustic signal and the network principles of node communication. Their main objectives were low energy consumption and system reliability but this approach has been studied only at the conceptual level, not being implemented and tested under real operating conditions [7].

One solution for detecting illegal deforestation is called Forest Guardian [8]. This system detects the sound produced by deforestation activities using acoustic signal evaluation and network node communication principles. The architecture of the system is modular, which means that the system functionality can be extended by adding new modules, but it mainly used a set of sensors to record environmental sounds. Forest Guardian instantly sends a notification (e-mail, SMS, etc.) to those responsible so that they can intervene to stop the illegal deforestation in the case of detection of deforestation activities in the monitored area. All areas in danger can be accessed on a map via a web application.

An implementation of the Forest Guardian system is Rainforest Connection, or RFCx as presented in [9]. It uses mobile phones as network nodes and solar panels to charge the battery. The nodes are individually mounted on the shafts and their position is set to allow superposition for detection and communication purposes.

Another method to protect forest from threats is presented in [10]. The authors propose a method to identify areas suspected of forest fires occurrence. The proposed method uses artificial intelligence, namely the neurofuzzy interface, for classification in two classes, forest fire and non-fire. Furthermore, cloud computing can be used for processing time critical data [11].

Another forest fire detection system is composed of a Raspberry Pi processing unit and three sensors: motion sensor, temperature sensor, and a gas sensor [12]. The motion sensor is used to prevent the destruction of the system from wildlife. The other two environmental sensors are configured to detect temperature value and gas concentration.

A different approach to forest monitoring is based on using arial vehicles. Small drones are increasingly being used by timber companies and government forestry agencies for applications such as tree crown/gap mapping, forest stand mapping, volume estimation, wind blow assessment, pest monitoring, and harvest planning. Additionally, conservation non-governmental organizations (NGOs) and staff of protected areas worldwide are becoming interested in using small drones for conservation-related tasks, such as: surveillance of wildlife, monitoring of land-use change and illegal activities within reserves such as poaching and illegal game hunting and forest fire prevention [13], [14], [15], [16], [17], [18].

Drone systems are disputed when it comes to forest protection. In the following paragraphs we present the advantages and disadvantages. Among the advantages we can list the followings:

- extremely high spatial resolution: The altitude at which small drones can be operated is usually in the range of 50-300m, an altitude that allows the acquisition of images with extremely high spatial resolution. At this spatial resolution, the forest can be easily monitored in detail. At this level of detail, the processes of loss, degradation and regrowth of forests could be accurately detected and monitored [19]. It is possible to identify individual tree species, detect pests and identify different stages of forest regeneration or degradation, all of which are fundamental to assessing the health of forests. The use of small drones would also allow detailed mapping over much larger areas than ground surveys.

- insensitivity to cloud cover: The fact that small drones fly at a height of about 50-100m is a significant advantage over conventional remote sensing platforms from high altitudes.

- relatively low price of drone imagery: The cost of acquiring, operating, and maintaining a small drone is much lower than the cost of deploying manned aircraft missions or acquiring images from any of the available high-resolution space satellites (e.g., IKONOS, QuickBird, RapidEye). For instance, [20] used a self-made conservation drone for tropical forest monitoring at an estimated cost of US$2,000, and are currently developing and testing cheaper models [18].

- enhanced monitoring of illegal activities: illegal deforestation could be monitored with these systems not only by monitoring the change in forest cover with time series photographs, but also by locating extraction routes and regularly monitoring the boundaries in real time using videography [21]. Fire, illegal deforestation and wildlife poaching could also be monitored in a timely manner [20].

Listed below are a number of disadvantages of the UAV approach to forest monitoring:

- Sensitivity to atmospheric conditions: The wind speed should not be higher than 15-25km/h, for a good image accuracy.

- Short flight endurance: small drones have a fairly small battery size, so flight time may not be enough for larger forests.

- Possibility of collisions: in the case of small drones, collisions can occur if the flight input coordinates are entered incorrectly or if something gets in their way, because they are not usually equipped with warning or evasion systems [21].

- Potential problems for repairs and maintenance: The drone can only be repaired by experts, which can be a significant problem if accidents occur or any component fails or something is lost or stolen.

- Safety & security issues: The safety of drone operators can be threatened if the exploitation of small drones takes place in dangerous areas such as community forests where illegal logging and farming, poaching, illegal drug production or military activities.

- Ethical issues: A major ethical concern is the possibility of violating confidentiality and the requirements for free, prior and informed consent (FPIC). These issues appear in all cases of surveillance of people, their property, resources and activities [22]. The use of surveillance drones without acceptable transparency and commonly agreed rules of engagement could lead to serious conflicts among community members. They can be charged with breach of privacy and espionage.

To communicate information from sensors spread over a forest area, low power radio or GSM are viable solutions. Because of huge coverage distance, the GSM infrastructure is an alternative to transmit or receive data from remote devices (sensor, actuator, complex device). An application created to monitor a remote photovoltaic system using a GSM and GPRS connection in presented in [23]. With technologies such as 3G, mobile phone technologies are able to provide internet access ubiquitously. The adoption of mobile internet technologies is galvanized by several perceived characteristics, such as advantages, ease of use, cost, quality, security and social influence [24]. Another study on the adoption of mobile technology in business demonstrates the use of the fit-viability framework in assessing mobile technologies for a variety of business use cases [25].

More recently, a new technology that has been developed and that has become very popular is based on wireless sensor networks (WSN). The analysis of sounds using classification algorithms to differentiate between vehicles, chainsaws and background forest noise is presented in [26]. The authors implemented the SeaForest prototype unit, with an overall architecture based on the Raspberry Pi platform connected to the ultrasonic sensor. Users of the mobile application receive notifications on the phone if suspicious noises are detected in the forest. SeaForest unit also detects presumptive fire when carbon dioxide concentration increases with temperature. To detect the sound of the chainsaw, acoustic sensors capture a wide range of ambient sounds that are analyzed through a frequency spectrum to identify artificial sounds. Through the application, the user can also view a series of parameters, for example temperature in a certain time interval. The prototype can be improved in several ways such as adding a solar panel, converter and battery.

The use of the solar panel with a Lead-acid battery is common, but there are also other options to consider. Supercapacitors have become more popular in recent years and the technology has improved. A study on replacing classic batteries with supercapacitors generated promising results [27]. The author analyzed several operating schematics (inverter and SEPIC) of buck-boost type converters that can be used together with supercapacitors.

An up-to-date review of the integration of IoT in the forest, including the recent emerging trends can be found in [28]. The review emphasizes the significance of integrating the internet modules in the forest environment for obtaining the real-time data based on the forest environmental parameters, as well as for alerting forest administrators about incidents such as fires.

A recently published technique improvement for the device-free localization (DFL) could prove useful in locating remote devices in dense forest areas with poor Global Positioning System (GPS) reception [29]. The proposed method is lightweight and can be implemented on resource-constrained devices, and also it can be incrementally updated to track environmental changes.

Other wireless sensing and detection applications include using a hierarchical sensing framework to fuse multiple features from different levels of abstraction in order to improve the robustness of the recognition system [30]. The proposed method was evaluated on a large dataset of handwriting gestures and showed improved performance over state-of-the-art methods.

The continuous and growing interest in Information and Communications Technology (ICT) adoption in business applications is moving towards integrating several services into web user interfaces dedicated to particular business processes involving humans. The increasing liberalization and fierce competition of the world economy has demanded the SMEs to continuously look for ways to improve their competitive ability. The SMEs should not underestimate their capability to compete in a larger market and that ICT adoption and utilization can act as a strategic tool to help them to achieve just that [31]. The environmental protection application presented in this article takes into account these trends and provides easy to use web interface dedicate to the process of monitoring forest plantations and exploitations.

III. EXPERIMENTAL MODEL

We designed an expert system in order to monitor activities in forest areas. The system has two main components: remote sensing hubs and a central cloud. There can be several sensing hub devices spread over an area of interest, each connected wirelessly via the internet to the central hub. Each device can be outfitted with corresponding sensors depending on its position, inside or near the forest. The central server is responsible for authenticating and authorizing devices to send

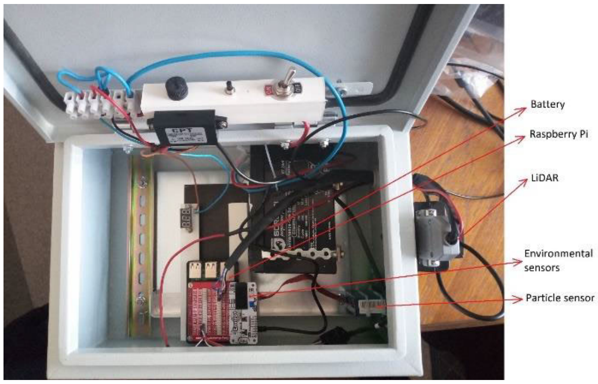

FIGURE 1. Sensing hub experimental model.

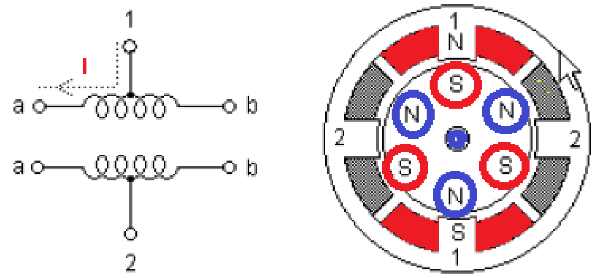

FIGURE 2. Stepper motor structure.

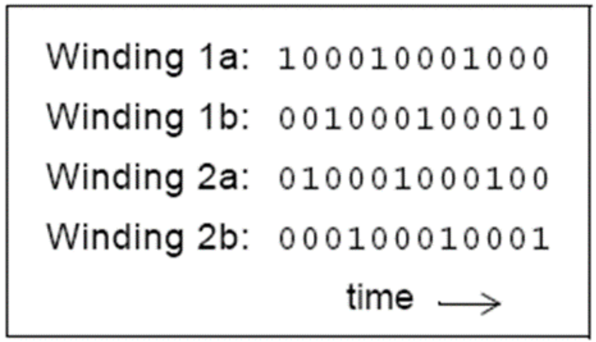

FIGURE 3. Control of SM with full steps.

data, for receiving and storing data, for making the data available to users via a web interface and email alerts.

The sensing hub experimental device uses a Raspberry Pi single board computer with several extension modules and sensors attached to it. An image of the model is presented in Fig. 1. In this paper we are going to focus on the LiDAR sensor placed near the forest, or in a clearing, with the LiDAR pointing at the forest edge.

A. SENSING HUB HARDWARE

There are a variety of LiDAR sensors available on the market, with significant differences in capabilities and prices. For the purpose of scanning the edge of a forest from the ground, we require a long-range sensor that can rotate around its vertical axis. The result was achieved with a static point LiDAR mounted on a platform rotated by a Stepper Motor (SM).

The stepper motor was chosen for the mechanical drive because of its many advantages. SMs have several characteristics that make them very suitable for use in such applications. Among these features are:

- no brushes - SMs have no brushes. The brushes and mechanical commutation of conventional motors are major disadvantages and are directly responsible for the occurrence of unwanted and even dangerous arcing in some environments;

- are not load dependent - SMs will operate at a fixed speed regardless of the load, provided the load is no greater than the maximum allowable;

- open loop positioning - SMs rotate step by step. As long as the motor is running at the specified torque, the rotor position is known at all times without the need for feedback;

- holding torque: SMs are capable of holding the rotor stationary;

- excellent response: starting, stopping and reversing

SM are modified synchronous electric machines whose control gears are fed by an m-phase system of virtually rectangular voltage pulses, the rotor being run without auxiliary drive gears. The pulsed voltage applied to the motor phases causes a discrete distribution of the magnetic field in the machine's interphase and, as a result, the rotor motion consists of successive elementary angular displacements.

Unlike classical synchronous motors, stepper motors enter synchronism without slipping and braking is achieved without getting out of synchronism. Because of this they ensure in the operating range short starts, stops and reversals without loss of information or step omissions.

In this case the motor chosen is of the unipolar type. These types of motors have 2 coils on the stator, each with an external and a middle terminal available either independently or connected together and only one terminal available externally. Such a motor has the structure shown in Fig. 2.

The middle socket is connected to the voltage source and the ends of the bushings are connected to the ground in succession.

In permanent magnet stepper motors, operation is based on the attraction of the north and south poles of the rotor to the stator poles. Thus, in these motors, the direction of the current determines which rotor pole will be attracted to which stator pole. The direction of current in unipolar motors depends on which half of the winding is energized.

Physically, the halves of the same winding are wound in parallel, so that, depending on which half the current flows through, the winding behaves as a north or south pole. In Fig. 2, the motor has winding 1 distributed between two diametrically opposite stator poles and winding 2 between the other two diametrically opposite poles. The rotor is a 6-pole permanent magnet.

In Fig. 4, the motion resolution is doubled by an appropriate command, compared to the standard situation in Fig. 3.

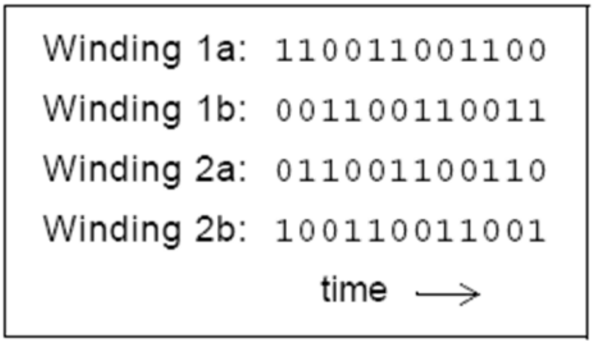

FIGURE 4. Control of SM with half steps.

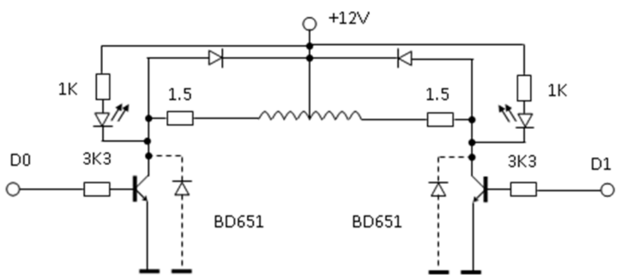

FIGURE 5. Example of control circuit for unipolar SM (one winding only).

To obtain higher angular resolutions, the motor must have more poles. Stepper motors with permanent magnets have been built with up to 100 poles for higher resolution. When the number of rotor poles is large, the stator poles are splined to allow them to interact with a large number of rotor poles.

An example of a control circuit for unipolar SM is shown in Fig. 5.

Specialized power Integrated Circuits (IC) such as ULN 2003 can be used instead of traditional transistors. The transient regime and the reduction of the time constant of the SM are aspects that can be analyzed according to the requirements of the SM use. The occurrence of transient operating modes is determined by the change in frequency of the control pulses, when starting, braking, reversing or changing the rotational speed, as well as the change in the resistive torque. During transient modes, the rotor must follow the sequence of control pulses without step loss, i.e., at the end of the transient mode, the angular position of the rotor must be at the point of stable equilibrium of the synchronized static torque characteristic corresponding to the envelope to which the control pulse is applied. However, during the transient regime, the rotor may remain behind, provided that it remains in the dynamic stability zone.

Although the commands to the General-Purpose Input Output (GPIO) port of the Raspberry PI module are sent in order (D0, followed by D1, D2 and D3 respectively), the motor windings are commanded in order 1 - 3 - 2 - 4.

The role of the freewheeling diodes (FD) is to provide a discharge path for the energy stored in the windings during their current flow, thus avoiding the transient mode that can destroy the control transistors.

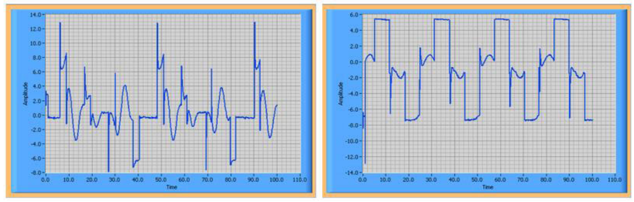

FIGURE 6. Current through a SM wrap, without FD (left) and with FD (right).

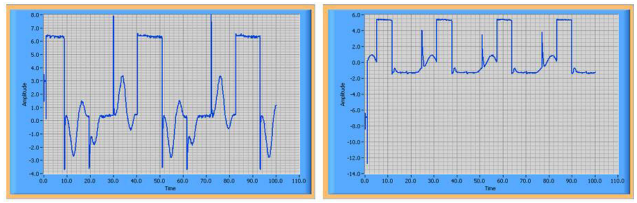

FIGURE 7. Voltage at the terminals of a winding, without FD (left) and with FD (right).

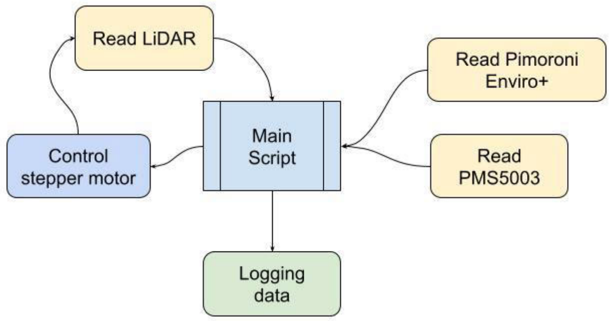

FIGURE 8. Software scripts diagram.

The effect of these diodes can be seen in the Fig. 6 and Fig. 7, where we present the qualitative representation of the current and voltage waveforms, in case of operation with and without free-running diodes.

In the case of half-step operation, the motor power is higher because it is given by the result of the interaction with two stator poles, which can be seen from the acquired signal, when 2 out of 4 intervals are active. In full-step operation, the motor power is lower than in the previous case, as only one of the 4 intervals is active.

B. SENSING HUB SOFTWARE

The Python code for the Raspberry Pi was split into multiple script files, each file dealing with one aspect of the data gathering process, as shown in Fig. 8:

- Getting data from the LiDAR sensor
- Controlling the motor
- Getting data from the particulates PMS5003 sensor
- Getting data from the Pimoroni Enviro+ sensor
- Logging the data
- Main script

1) GETTING DATA FROM THE LiDAR SENSOR

The process of getting the readings from the LiDAR sensor was straightforward and involved reading from the serial port

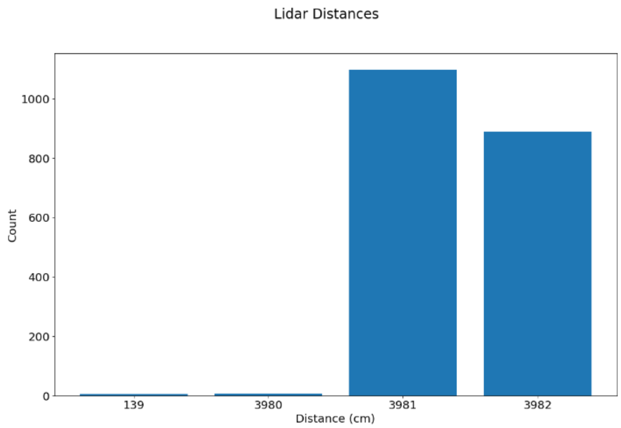

FIGURE 9. Measured LiDAR distances.

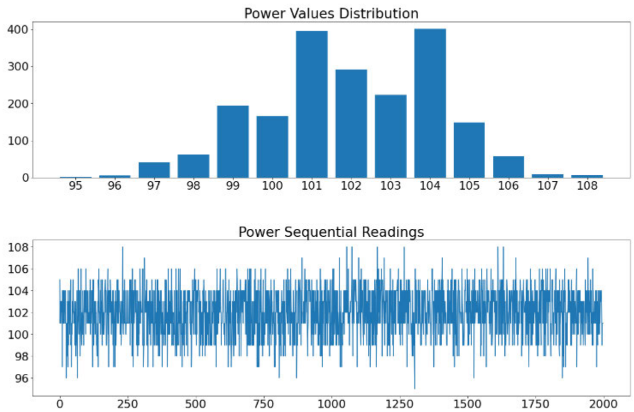

FIGURE 10. Measured LiDAR power.

sequences of 9 bytes from which, the third and the fourth were storing the value for the distance while the fifth and the sixth were storing the value for the power.

We've analyzed the distances returned from the LiDAR sensor while taking 2000 sequential readings while probing an object situated at around 40 m from the sensor, as seen in Fig. 9. From the 2000 readings, 5 of them returned an erroneous value of 139 cm while the rest of them were around the 3981 cm value. In order to remove possible outliers while measuring the distance to a point, the implemented Python script is taking 20 consecutive readings and from the returned values, the script returns the median value.

The analysis of the power values returned while taking 2000 sequential readings showed a different variation distribution, as seen in Fig. 10. With a standard deviation of 2.19 and a variance of 4.7, we can see that the value for power oscillates in a ± 7 interval around the correct value. To have stable readings throughout multiple runs, the script is taking for each point 20 consecutive readings and from the returned values, it selects the median value.

2) CONTROLLING THE MOTOR

The motor can be moved one step to the left or to the right by using a predefined sequence of signals sent to the four GPIO pins, in our case the GPIO12, GPIO16, GPIO20 and GPIO21. To move to the left, the sequence used was: 1000, 1100, 0100, 0110, 0010, 0011, 0001, 1001. A pause of 0.01 seconds was used between each step in the sequence. The sequence used to move in the opposite direction was: 0001, 0011, 0010, 0110, 0100, 1100, 1000, 1001.

To get multiple readings, the motor moves a number of steps to the right and at each step, it uses the LiDAR script to get the distance and the power of the probed point. At the end of the scanning process, the motor moves the same number of steps in the left direction.

3) GETTING DATA FROM THE PARTICULATES PMS5003 SENSOR

To get data from the particulate sensor, the script uses the pms5003 Python library provided by Pimoroni [32]. The sensor returns the concentration for three types of particles: PM1.0 (ultrafine particles with a diameter of 1 micron or lower), PM2.5 (combustion particles, organic compounds or metals with a diameter of 2.5 microns or lower) and PM10 (dust, pollen or mold spore particles with a diameter of 10 microns or lower).

4) GETTING DATA FROM THE PIMORONI ENVIRO+ SENSOR

The Pimoroni Enviro+ has multiple environmental sensors [33], that can be easily accessed using the following Python libraries:

- enviroplus – provided by Pimoroni is used to get the NH3, oxidizers and reducers
- ltr559 – used to get the values from the optical sensor that returns the light intensity
- bme280 – used to get the temperature, pressure and humidity sensor

In the case of the temperature sensor, because the Enviro+ board is very close to the Raspberry Pi board, the returned temperature is not accurate and needs to be compensated taking into consideration the temperature of the CPU. We calculate an average for the CPU temperature by getting multiple readings and by calculating a mean value. From the raw temperature value, we subtract a percentage of the difference between the raw sensor value and the mean CPU temperature. We used a 22% value based on experimental tests. The adjusted temperature value is just an approximation of the real value but for our application, where the temperature is not an essential sensor, it fulfils its intended function of getting an estimation of the environment temperature.

5) LOGGING THE DATA

Throughout multiple runs of getting the sensor's data and sending it to the server, it might be possible that the GSM signal drops or the server becomes unavailable. In this case, in order to not lose data, the script logs information in a file that is used as a records queue. In the next runs, if the connection is still not available, the data is added to the same file. The moment the connection is re-established, all the queued records are sent to the server and then removed from that log file. The concept applies to individual records in the file to account for the situation when connection is lost while sending logged data. In this way, we don't lose any records that were carefully collected by the device.

Another functionality implemented using the logging script is that of saving all the records ever collected to a separate file that can be used as a backup solution if anything happens to the records saved in the database.

The files generated by the logging script are JSON encoded strings of the records and can be manually imported into the database using the MEPHIFA web app described in the following section.

6) CONSTANTS

A separate file is used to store various constants used throughout the Python scripts and can be used to quickly change essential parameters. The following constants are stored in this script:

- the device code (e.g., 5821c0d5-e81e-3545-83b7-5e4e0689c234) that is used when sending data to the server
- the server address where the data is sent
- the name of the records queue and backup files
- the motor wait interval between each GPIO sequence
- the number of LiDAR points recorded
- the number of sequential readings for different sensors (LiDAR, particulates or enviro+)
- the temperature correction factor

7) MAIN SCRIPT

The main script connects all the other scripts and gathers the data from all the sensors and sends it to the server. The cron scheduler was used to repeat the execution of the main script at a set interval (10 minutes).

C. CENTRAL SERVER BACKEND SOFTWARE STACK

The web app developed as a dashboard to interact with the system will be built using an open-source Linux, Apache, MySQL, PHP (LAMP) stack because this offers a highly customizable, independent platform that is secure, scalable and allows fast development. For PHP we took into consideration multiple frameworks that can be used to provide a basic structure that can speed up the development process. The frameworks we have analyzed are the following:

Laravel is the most popular open-source PHP framework that offers easy configuration and customization of common application tasks as: authorization, encryption, hashing, password reset, data migration and routing. Because of the MVC (Model View Controller) architecture used it is highly flexible and scalable for any type of applications. The Laravel Artisan Console command line tool speeds up development by automating repetitive tasks and generating boilerplate code. Laravel is fast, focuses on simplicity, is very easy to use and has a large community of developers and extensive documentation.

Symfony is easy to install and configure on most platforms, making it perfect choice for developing complex enterprise level applications. Some of its key features include reusable PHP components for faster development, database engine independence, stability and it follows best practices and design patterns. It is more complex compared with Laravel, targets advanced developers thus being harder to start with while also being a little slower than the other frameworks because of its feature-rich nature.

CodeIgniter is a lightweight PHP framework that is easy to install and configure. If provides error handling, MVC architecture, inbuilt security tools and excellent documentation. Unfortunately, due to irregular releases, the platform isn't that good at offering high-level security.

CakePHP is an open-source PHP framework that follows MVC architecture, is simple to install and uses a minimal configuration. 'Convention over configuration' (COC) is a software engineering concept that powers this framework but can be a bit restrictive due to its strict following of conventions. The framework provides features like caching, authentication, validation, internationalization and a simple to use inbuilt ORM (Object Relational Mapping).

Phalcon is a high-performance PHP framework that uses MVC architecture and is delivered as a web server extension written in C. It is easy to install, highly configurable and provides high execution speed, asset management, good security and caching. It is loosely coupled and allows developers to create their own directory structure. The Volt templating engine is extremely fast and comes with helper classes to create views easily. The community size is not comparable with the other more popular frameworks and the updates and patches are irregular and this may not be suitable for applications that need high levels of security.

Yii is a simple to use, high-performance and component-based PHP framework that is suitable for all kinds of web applications: portals, CMS, e-commerce, projects, forums, etc. It is a very extensible and flexible platform that allows customization at a fine-grained level. The documentation is comprehensive and the massive community offers a great support. The disadvantage of Yii is that it has a steep learning curve and requires greater attention during development as one mistake in code can cause issues in the entire application

Upon this analysis, Laravel is the framework that is the most suitable for this project if we consider the scale of the application, the security needs, the development time and the available resources.

D. CENTRAL SERVER FRONTEND SOFTWARE STACK

The MEPHIFA web app was built to provide an API endpoint that allows the remote devices to send the gathered data to the server. On the other hand, it allows registered users to quickly inspect the data gathered from multiple devices.

The front page of the app allows the user to log in to the app or, for the new users, to register for a new account using a valid email address that is required in order to finalize the registration process as seen in Fig. 11.

Fig. 12 shows the access levels available to registered users: registered, active or admin. The users that have

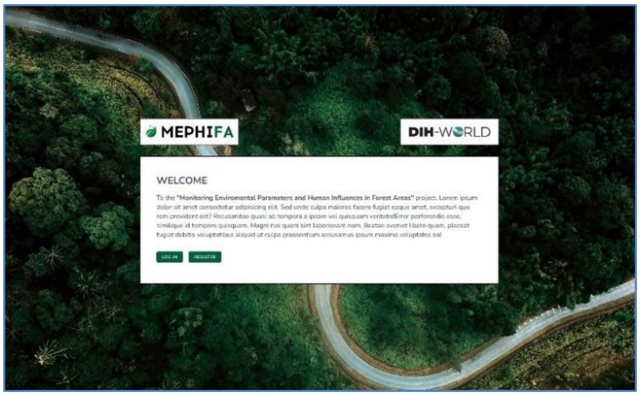

FIGURE 11. MEPHIFA web application.

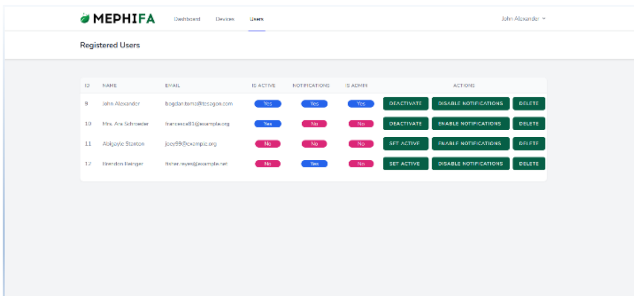

FIGURE 12. User management.

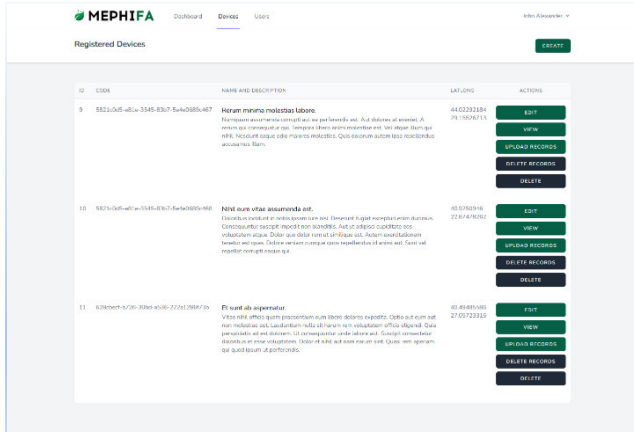

FIGURE 13. Device management.

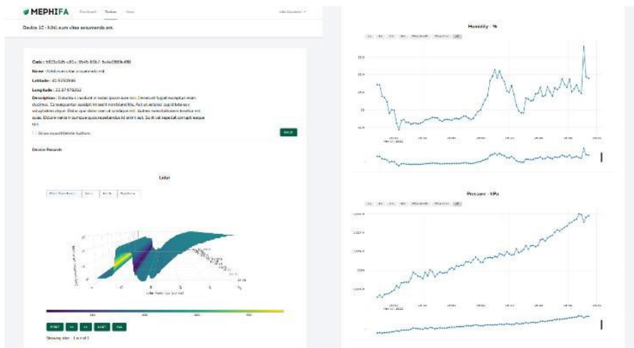

FIGURE 14. Device details page.

completed the registration process and have their email verified are "registered" users and can log in to the platform but they can't see or access any data.

A user with admin privileges can change the status of a user from "registered" to "active" which means that now the user can see the devices, the sensor records or the alerts but it doesn't have access to sensible actions like deleting devices, records, alerts or users. From the Users page, a user can disable or enable email notifications related to sensor alerts.

On the Devices page, we have access to the devices saved in the application and we can edit, delete the records, delete the device or create a new one, Fig. 13. Each device is created with a unique serial number that is used to validate incoming requests to save new sensor readings in the database.

Because there are situations where a device can't send the gathered data to the server (e.g., bad GSM signal or the server is unavailable), the app allows the loading of data for a device from an external file. The file can contain multiple sensor readings encoded as a JSON string and are generated on the device in the case the connection with the server malfunctions. When the connection with the server is re-established, before sending the new readings, the Python script checks also for old records that were not sent and if it finds any, it sends all the available data to the server. The script also saves in a separate file all the data it gathers as a backup, which can be used in the case that something happens to the server database.

The details page for each device displays along with the set properties (code, name, description, latitude and longitude), all the records found in the database, grouped by sensor name, Fig. 14. Each device can have multiple sensors and they are of two major types based on the returned value:

- Single value: humidity, NH3, oxidizers, reducers, pressure, temperature
- Multi-value: LiDAR (the LiDAR probes multiple points and for each point it returns the distance to that point and the power that changes depending on the type of surface it encounters)

For the graphs displayed in the dashboard we searched for a JavaScript library that can easily display graphs of time series and 3D surfaces. The readings of the various sensors will be displayed as a graph where the plotted interval can be easily selected from the whole recorded history. For the LiDAR sensor, each record consists in a series of values and the whole history, or a section of it, can be displayed as a 3D surface. We've taken into consideration the following free libraries: Plotly, MetricsGraphics.js, C3.js, Dygraphs, D3.js, AnyChart.js. From all of them, we've chosen Plotly because its features are covering all the project needs and it's easier to use compared with D3.js

For the single value sensors, multiple readings are displayed as a time-series plot that shows the evolution of that sensor throughout an interval of time, as in Fig. 15.

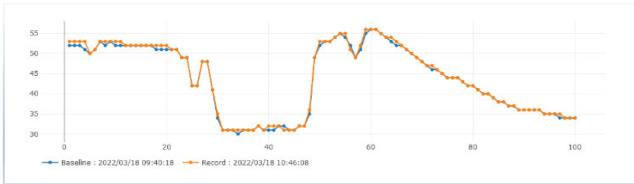

FIGURE 15. Single value plot.

FIGURE 16. LiDAR sensor plot.

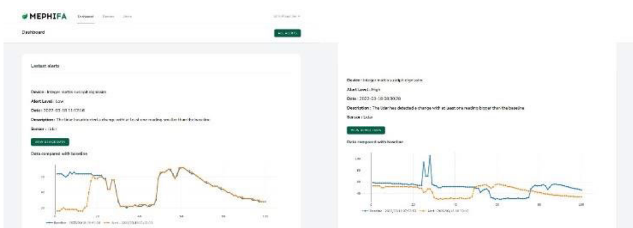

FIGURE 17. Expert system dashboard.

For the multi-value sensor, to show the evolution in time, a 3D shape plot is used, where the height of the points is controlled by the recorded distance and the colour is used to show the power returned by the LiDAR sensor, Fig. 16.

For each new LiDAR record added to the database, the new readings are compared with a set baseline and if the differences found are higher than a set threshold, an alert is generated and a notification email is sent to the users that have opted to receive the notifications. In the initial testing of the LiDAR sensor, we found that a threshold of ± 10 cm was enough to filter out the sensor errors while also having enough fidelity to detect small changes from the set baseline. Because a device can be moved to a new location, the app allows setting multiple baselines so that new records are compared with a proper profile and not an old baseline that would certainly generate an alert.

The Dashboard page displays the latest alerts and we can also access the page where all the alerts are shown in a table. Alerts can be viewed (the record values are plotted along with the baseline values) or deleted, as shown in Fig. 17.

The generated alerts can be of two levels: high or low. The high alert is generated when the new data shows that at least a point has a distance value greater than a threshold compared with the baseline (the distance to a point where it was a tree increased which means that the tree was cut) or when the particulates sensor detects smoke. The device uses the PMS5003 particulates sensor to detect the smoke and the threshold used to generate an alarm is 1000 μg/m3 for the

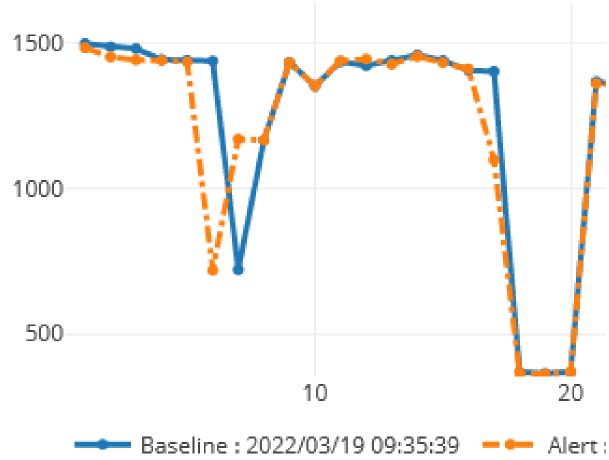

FIGURE 18. Effects of rotation drift of the LiDAR platform.

concentration of PM2.5 particulates (particles with a diameter of 2.5 microns or less).

The low alerts are generated when the recorded distances are lower than the baseline which may happen if an object passes in front of the LiDAR sensor or there are special meteorological conditions such as fog or rain. Shorter distance measurements can appear in normal situations, such as movements due to wildlife or falling branches. To account for these limitations, signals from several sensors need to be corelated before sending out an alert.

IV. EXPERIMENTS

In order to assess the effectiveness of the expert system, a remote sensing hub was installed near the edge of a forest in Romania, near Bucharest. It was closely monitored during the first hours to identify issues and solve them. Some of them are presented in the following paragraphs.

The mechanical connection between the LiDAR and the motor had a slight movement. The connection cable of the LiDAR was thick and the rotation of the motor was quickly adding tension to the cable. Because of this, in some instances, the power of the motor was too weak to counter the tension and it would remain stuck in the same position and the LiDAR was probing the same point multiple times. Another problem with the tension in the cable was that throughout multiple runs the alignment of the starting point would drift to the left or to the right depending on the way the cable was positioned. The cable was replaced with a lighter one, leading to a significant improvement. However, some drifts could still be seen, as in Fig. 18.

The initial connection of the motor controls was overlapping with the particles sensor and we couldn't get any readings from the sensor. Taken separately they would work just fine but connected together the particles sensor was unavailable. The solution was to connect the LiDAR sensor through a serial connection at a USB port of the Raspberry Pi and leave the RT/TX pins to be used by the particle sensor.

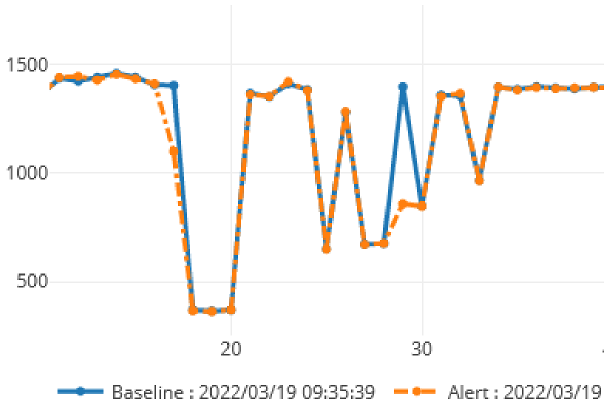

FIGURE 19. Alert generated for foreign objects.

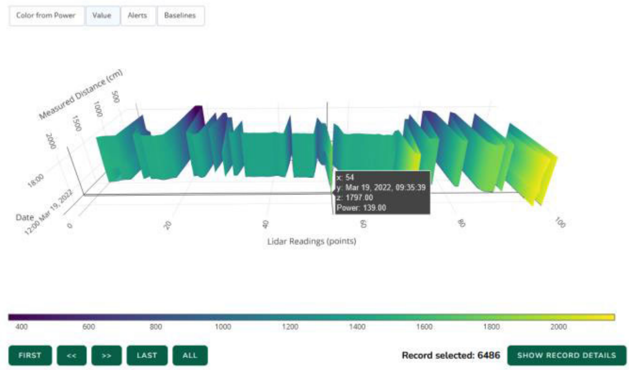

FIGURE 20. LiDAR sensor data visualization in the web interface.

The motor has a step of 0.5 degrees which translates into 120 readings while covering 60 degrees of movement. Because the LiDAR has maximum distance reading of 180 meters, we can calculate the maximum distance between two reading points (when the readings are near the 180 meters limit) to be around 1.57 meters. This means that the resolution of the LiDAR readings near the 180 meters limit is too low to capture a detailed representation of the environment. We have tried to parallelize the LiDAR readings and the movement of the motor such that we would get multiple readings while the motor moves one step. The frequencies of the LiDAR readings and of the motor are both around 100 actions/second and this means that we can't get multiple LiDAR readings while the motor moves one step. Keeping them as sequential operations (movement, reading, movement, reading, etc.) ensured that we get a fixed set of readings, equal to the number of motor steps and that each time we would probe the same point. If we would keep them as parallel operations the number of readings would fluctuate with 1 or 2 readings at each run and that would make the comparison of different runs very difficult. Another benefit of keeping them sequential was the possibility of having multiple readings for each point so that we can counter a slight variation we would get in an initial reading after the LiDAR was moved to a new point.

Our experiments showed that using the LiDAR sensor to read distances through a glass window would return erroneous readings since the laser ray would be reflected or refracted depending on the angle of incidence. For the same reason, having glass surfaces in the environment that was probed, was also a big problem.

By placing other objects between the sensors hub and the trees we confirmed the LiDAR was able to pick up the foreign objects and the central hub of the expert system generated alerts. Fig. 19 shows the baseline measurement in blue and the new measurement in orange. There are foreign objects successfully identified based on the median baseline and the current reading.

The LiDAR sensor probes the environment on a horizontal plane and this means that the position of the device is limited by the height we use to take the measurements (usually 1 m from the ground). This makes the device vulnerable to malicious agents and should be positioned at safe higher position.

On the server side, the initial implementation for the visualization of the LiDAR readings was displaying all the records found in the database and that resulted in slow loading times for the 3D surface plot. The code was updated to allow the user to load and display the records in sections and this made the application perform faster while increasing the quality of the user interaction with the plots as inspecting only a section of the records was easier than managing all the records in one go. The presentation of the LiDAR data acquired from the test site can be seen in the web interface, as presented in Fig. 20.

V. CONCLUSION

Forests play a major role in the global ecosystem, beyond wood production. These areas need to be protected from illegal logging, forest fires, deforestation and other unwanted activities. In order to achieve this goal, modern technology can be employed as a distributed real-time sensing mesh on forest areas.

IoT is a viable solution for implementing real-time monitoring and alerting in forest areas. The expert system developed and presented in this article has proven that LiDAR is a viable data source in forest environments. It can identify foreign objects and can contribute to enforce indicators of smoke and fog in the environment using LiDAR pattern changes based on atmospheric perturbances such as mist, fog or smoke.

Small reaction time in case of critical events is very important. The proposed system has proven to be effective in identifying abnormal situations and sending alerts to the users registered in the web application. Furthermore, real-time knowledge of the conditions in the forest contributes to increasing the safety of the humans that intervene in crisis situations.

While the LiDAR sensor proved to be useful, we also identified limitations of the implementation presented in the article. The rotation of the platform using a stepper motor leads to small drifts that can affect the algorithms checking for abnormal situations. To solve this issue, a self-correction mechanism based on markers at the leftmost and rightmost limits of the scanning range could be devised in future research.

The distance mapping created using the LiDAR sensor can be affected by factors such as atmospheric conditions, wildlife movements or falling branches. To avoid false positive alerts, the system corelates information from all sensors, but a future research direction could use classification algorithms to identify patterns that should not trigger an alert.

Another direction for future work is to explore the use of proposed method in other application domains and to investigate positive synergies with other types of sensors and technologies, such as wireless sensing and cameras with computer vision software.

ACKNOWLEDGMENT

The authors would like to thank their employing companies, Tesagon International SRL and the Politehnica University of Bucharest, for supporting the MEPHIFA research project and the publication of this article.

REFERENCES

[1] J. R. Boyle, J. C. Tappeiner, R. H. Waring, and C. T. Smith, "Sustainable forestry: Ecology and silviculture for resilient forests," in *Reference Module in Earth Systems and Environmental Sciences*. Amsterdam, The Netherlands: Elsevier, 2016, doi: 10.1016/B978-0-12-409548-9.09761-X.

[2] J. Ghazoul, "Deforestation and land clearing," in *Encyclopedia of Biodiversity*. Amsterdam, The Netherlands: Elsevier, 2013, pp. 447–456, doi: 10.1016/B978-0-12-384719-5.00281-1.

[3] J.-L. Dupuy, P. Vachet, J. Maréchal, J. Meléndez, and A. J. de Castro, "Thermal infrared emission-transmission measurements in flames from a cylindrical forest fuel burner," *Int. J. Wildland Fire*, vol. 16, no. 3, p. 324, 2007, doi: 10.1071/WF06043.

[4] P. Kanakaraja, P. S. Sundar, N. Vaishnavi, S. G. K. Reddy, and G. S. Manikanta, "IoT enabled advanced forest fire detecting and monitoring on ubidots platform," *Mater. Today, Proc.*, vol. 46, pp. 3907–3914, Jan. 2021, doi: 10.1016/j.matpr.2021.02.343.

[5] G. Patra and L. Goswami, "WITHDRAWN: Forest protection using wireless sensor network and IoT," *Mater. Today, Proc.*, May 2021, Art. no. S2214785321028820, doi: 10.1016/j.matpr.2021.03.742.

[6] D. Riaño, E. Chuvieco, S. L. Ustin, J. Salas, J. R. Rodríguez-Pérez, L. M. Ribeiro, D. X. Viegas, J. M. Moreno, and H. Fernández, "Estimation of shrub height for fuel-type mapping combining airborne LiDAR and simultaneous color infrared ortho imaging," *Int. J. Wildland Fire*, vol. 16, no. 3, p. 341, 2007, doi: 10.1071/WF06003.

[7] Y. G. Sahin and T. Ince, "Early forest fire detection using radio-acoustic sounding system," *Sensors*, vol. 9, no. 3, pp. 1485–1498, Mar. 2009, doi: 10.3390/s90301485.

[8] J. Papán, M. Jurecka, and J. Púchyová, "WSN for forest monitoring to prevent illegal logging," in *Proc. Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, Sep. 2012, pp. 809–812.

[9] *Rainforest Connection—Stopping Illegal Logging & Protecting Wildlife*. Accessed: Jun. 11, 2023. [Online]. Available: https://rfcx.org/our_work

[10] H. Zhou, A. Taal, S. Koulouzis, J. Wang, Y. Hu, G. Suciu, V. Poenaru, C. de Laat, and Z. Zhao, "Dynamic real-time infrastructure planning and deployment for disaster early warning systems," in *Computational Science—ICCS*, Y. Shi, H. Fu, Y. Tian, V. V. Krzhizhanovskaya, M. H. Lees, J. Dongarra, and P. M. A. Sloot, Eds. Cham, Switzerland: Springer, 2018, pp. 644–654, doi: 10.1007/978-3-319-93701-4_51.

[11] D. T. Bui, Q.-T. Bui, Q.-P. Nguyen, B. Pradhan, H. Nampak, and P. T. Trinh, "A hybrid artificial intelligence approach using GIS-based neural-fuzzy inference system and particle swarm optimization for forest fire susceptibility modeling at a tropical area," *Agricult. Forest Meteorol.*, vol. 233, pp. 32–44, Feb. 2017, doi: 10.1016/j.agrformet.2016.11.002.

[12] N. Sarasvathi and F. K. Chelsea, "Study and implementation of Internet of Things (IoT) based forest fire automation system to detect and prevent wildfire," *INTI J.*, vol. 1, no. 15, pp. 1–5, 2018. [Online]. Available: http://eprints.intimal.edu.my/1149/

[13] V. G. Ambrosia, S. S. Wegener, D. V. Sullivan, S. W. Buechel, S. E. Dunagan, J. A. Brass, J. Stoneburner, and S. M. Schoenung, "Demonstrating UAV-acquired real-time thermal data over fires," *Photogrammetric Eng. Remote Sens.*, vol. 69, no. 4, pp. 391–402, Apr. 2003, doi: 10.14358/PERS.69.4.391.

[14] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *Int. J. Syst. Sci.*, vol. 37, no. 6, pp. 351–360, May 2006, doi: 10.1080/00207720500438480.

[15] E. A. Hinkley and T. Zajkowski, "USDA forest service-NASA: Unmanned aerial systems demonstrations—Pushing the leading edge in fire mapping," *Geocarto Int.*, vol. 26, no. 2, pp. 103–111, Apr. 2011, doi: 10.1080/10106049.2011.555823.

[16] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robotic Syst.*, vol. 65, nos. 1–4, pp. 533–548, Jan. 2012, doi: 10.1007/s10846-011-9560-x.

[17] *Nepal Tests New Unmanned Aerial Vehicle Technology to Stop Wildlife Crime*. Accessed Jun. 11, 2023. [Online]. Available: https://wwf.panda.org/wwf_news/?206154/Nepal-tests-new-unmanned-aerial-vehicle-technology-to-stop-wildlife-crime

[18] *Conservation Drones*. Accessed: Jan. 19, 2022. [Online]. Available: https://conservationdrones.org/

[19] J. Paneque-Gálvez, M. McCall, B. Napoletano, S. Wich, and L. Koh, "Small drones for community-based forest monitoring: An assessment of their feasibility and potential in tropical areas," *Forests*, vol. 5, no. 6, pp. 1481–1507, Jun. 2014, doi: 10.3390/f5061481.

[20] L. P. Koh and S. A. Wich, "Dawn of drone ecology: Low-cost autonomous aerial vehicles for conservation," *Tropical Conservation Sci.*, vol. 5, no. 2, pp. 121–132, Jun. 2012, doi: 10.1177/194008291200500202.

[21] A. Horcher and R. Visser, "Unmanned aerial vehicles: Applications for natural resource management and monitoring," in *Council on Forest Engineering Proceedings 2004: Machines and People*. Hot Springs, AR, USA, 2004, pp. 853–858. [Online]. Available: https://www.researchgate.net/publication/237232718_Unmanned_Aerial_Vehicles_Applications_for_Natural_Resource_Management_and_Monitoring

[22] C. Nast. *Here Come the Drones!* Wired U.K. Accessed: Jun. 11, 2023. [Online]. Available: https://www.wired.co.uk/article/here-come-the-drones

[23] C. D. Oancea, "GSM infrastructure used for data transmission," in *Proc. 7TH Int. Symp. Adv. TOPICS Electr. Eng. (ATEE)*, May 2011, pp. 1–4.

[24] A. Y. L. Chong, N. Darmawan, K. B. Ooi, and B. Lin, "Adoption of 3G services among Malaysian consumers: An empirical analysis," *Int. J. Mobile Commun.*, vol. 8, no. 2, p. 129, 2010, doi: 10.1504/IJMC.2010.031444.

[25] T. Liang, C. Huang, Y. Yeh, and B. Lin, "Adoption of mobile technology in business: A fit-viability model," *Ind. Manag. Data Syst.*, vol. 107, no. 8, pp. 1154–1169, Oct. 2007, doi: 10.1108/02635570710822796.

[26] A.-E. Marcu, G. Suciu, E. Olteanu, D. Miu, A. Drosu, and I. Marcu, "IoT system for forest monitoring," in *Proc. 42nd Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2019, pp. 629–632, doi: 10.1109/TSP.2019.8768835.

[27] C. D. Oancea, "Aspects of renewable energy supply to small consumers," in *Proc. Int. Conf. Expo. Electr. Power Eng.*, Oct. 2012, pp. 964–967, doi: 10.1109/ICEPE.2012.6463809.

[28] R. Singh, A. Gehlot, S. V. Akram, A. K. Thakur, D. Buddhi, and P. K. Das, "Forest 4.0: Digitalization of forest using the Internet of Things (IoT)," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5587–5601, Sep. 2022, doi: 10.1016/j.jksuci.2021.02.009.

[29] J. Zhang, Y. Li, W. Xiao, and Z. Zhang, "Online spatiotemporal modeling for robust and lightweight device-free localization in nonstationary environments," *IEEE Trans. Ind. Informat.*, vol. 19, no. 7, pp. 1–11, Jul. 2022, doi: 10.1109/TII.2022.3218666.

[30] J. Zhang, Y. Li, H. Xiong, D. Dou, C. Miao, and D. Zhang, "HandGest: Hierarchical sensing for robust-in-the-air handwriting recognition with commodity WiFi devices," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19529–19544, Oct. 2022, doi: 10.1109/JIOT.2022.3170157.

[31] K. S. Tan, S. C. Chong, B. Lin, and U. C. Eze, "Internet-based ICT adoption: Evidence from Malaysian SMEs," *Ind. Manag. Data Syst.*, vol. 109, no. 2, pp. 224–244, Mar. 2009, doi: 10.1108/02635570910930118.

[32] *Particulate Matter Sensor Breakout (for PMS5003)*. Accessed: Jun. 11, 2022. [Online]. Available: https://shop.pimoroni.com/products/particulate-matter-sensor-breakout

[33] *Enviro for Raspberry PI—Enviro*. Accessed: Jun. 11, 2022. [Online]. Available: https://shop.pimoroni.com/products/enviro

CIPRIAN M. COMAN received the bachelor's degree in computer science and engineering and the M.S. degree in information technology management from the Politehnica University of Bucharest, in 2011 and 2013, respectively, where he is currently pursuing the Ph.D. degree in IoT technologies for Industry 4.0.

From 2009 to 2014, he was a Teaching Assistant for the applied electronics optional course, algorithms design course, and software project management course with the Politehnica University of Bucharest. He is currently leading an invocative ICT company, Tesagon International SRL, Romania. He has experience as a researcher in several projects, as well as industry experience as a software engineer, a system architect, and the manager. His interests include the IoT, automation, digital industry, clean energy, cloud systems, web technologies, artificial intelligence, and blockchain.

BOGDAN C. TOMA received the degree from the Faculty for Automation Control and Computers, Politehnica University of Bucharest, Bucharest, Romania, in 2005.

From 2005 to 2018, he was a Software Developer and a Designer in the field of animations and graphics. Since January 2019, he has been a Software Developer and a Research Assistant with Tesagon International SRL, Bucharest, where he is a part of the research and development fellowship. He is actively involved in the research projects of the company.

MIHAI A. CONSTANTIN received the B.S. and M.S. degrees in database systems from the Petroleum-Gas University of Ploieşti, Romania, in 2010 and 2012, respectively.

From 2014 to 2016, he was a Purchasing Specialist, gaining hands-on experience in an engineering company. Since 2017, he has been a Software Developer and a Research Assistant with Tesagon International SRL, where he is a part of the research and development fellowship.

Mr. Constantin was awarded a studies grant by the Romanian Education Ministry.

ADRIANA FLORESCU (Senior Member, IEEE) received the degree in engineering and the Ph.D. degree in electronics from the Politehnica University of Bucharest, Romania, in 1988 and 2001, respectively. Since 1993, she has been with the Department of Industrial Electronics and Informatics, Faculty of Electronics, Telecommunications and Information Technology, Politehnica University of Bucharest. She has been a Professor, since 2015, and a Ph.D. Coordinator, since 2016. Her research interests include power electronics, renewable energy conversion, artificial intelligence, electric vehicles, medical electronics, and the IoT. She has published more than 100 publications, one patent, and ten books in these fields. She has been an IEEE member of the Power Electronics Society, since 1995, and the Social Chair of the Executive Committee of the IEEE Romanian Section, since 2013.