

Received May 3, 2019, accepted July 19, 2019, date of publication July 26, 2019, date of current version August 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2931456

Crowd-Sourced Wildfire Spread Prediction With Remote Georeferencing Using Smartphones

NIKOS BOGDOS¹ AND ELIAS S. MANOLAKOS^{1,2}, (Senior Member, IEEE)

¹Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, 15784 Athens, Greece

²Bouvé College of Health Sciences, Northeastern University, Boston, MA 02115, USA

Corresponding author: Elias S. Manolakos (eliasm@di.uoa.gr)

ABSTRACT Wildfires are natural hazards with severe consequences worryingly worsening for many climate-change affected regions of our planet. Unfortunately, technologies that can provide real-time fire-line information, such as satellites, in-field sensors, and social media texts, exhibit low spatial/temporal resolution or cannot be deployed cost-effectively in widespread geographical areas. We present the design, development, and implementation of a novel software service, called *CITISENS*, which by exploiting commodity smartphone sensors allows ordinary citizens to easily georeference a fire-line in real-time and report its coordinates as they are photographing a wildfire. The location/orientation sensors and the camera are used to compute the view-ray of the smartphone, and a digital elevation model is employed to estimate the ray's intersection with the topography. We have tested the georeferencing accuracy obtained and it is to be on par with, or even better, than that of existing satellite wildfire hotspot services. When combined with *FLogA*, a flexible wildfire spread simulator we have also developed, *CITISENS* offers the following unique advantages: real-time prediction of burn probabilities, dynamic assimilation of citizen-reported hotspots into ongoing simulations for improved predictive accuracy, and decision support to issue citizen alarms based on the estimated time-dependent risk at their location due to an approaching wildfire.

INDEX TERMS Citizen science, georeferencing, smartphones, wildfire monitoring, wildfire spread simulation.

I. INTRODUCTION

We have become the passive observers of the dire consequences of climate change on wildfires' frequency and scale. Admittedly, the wildfire periods are becoming longer and more intense, and there is nothing to suggest that this trend will change in the foreseeable future. Never before was it of such paramount importance to develop novel methods and tools empowering our communities in combating more effectively the evermore dangerous wildfires. Primarily when wildfires affect the Wildland Urban Interfaces (WUIs) [1], guarding the community against their catastrophic effects becomes even more challenging, as demonstrated many times before with the most recent cases being the tragic events at Mati a suburb of Athens, Greece [2] and in the state of California, US [3].

Wildfire hotspot monitoring services are currently based on satellite or in-field sensors. Although very useful, both

technologies present serious drawbacks limiting their practicality. Understandably, we cannot deploy cost-effectively in-field wireless sensor networks of sufficient density in all areas we want to protect, and satellite sensors cannot provide adequate temporal accuracy when in polar orbit [4], or spatial accuracy when in geostationary orbit [5].

On the bright side, we are experiencing a fast-growing trend towards manipulating geographic information in social media, as we –the citizens– are becoming geographic data providers during a crisis event. Such collective social behavior gives rise to the so-called Volunteered Geographic Information (VGI) datasets [6]. However, social media text does not provide sufficient information about the geography of a wildfire. The crucial question then becomes: Given the widespread social participation and interaction during a wildfire event [7]–[10] can we create easy to use and effective tools empowering our communities to manage wildfire risk? We are addressing this critical question by developing a pilot Citizen Science program [11] aiming to raise wildfire management awareness and efficiency by decentralizing the

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

efforts, shifting some of the burden of fire line monitoring from the civil protection authorities to the concerned citizens themselves.

The swarm of sensors available in smartphones today (gyroscopes, accelerometers, magnetometers, GPS, cameras) can be exploited by citizens to georeference and report visible wildfire hotspots just as they are photographing a wildfire. Also, georeferencing and camera pose estimation using smartphones, or other camera systems equipped with a Global Navigation Satellite System (GNSS) and an Inertial Measurement Unit (IMU), have been put to the test in recent years. For example, [12] develops a close-range Augmented Reality (AR) system for GIS Registration and [13] demonstrates direct georeferencing on Unmanned Aerial Vehicles (UAV). Moreover, [14], [15] and [16] use a smartphone as a Mobile Mapping System to georeference close range objects employing photogrammetric techniques, to overcome the low accuracy of navigation sensors [17], [18]. Furthermore, [19] uses smartphones effectively to georeference video streams and demonstrates how they can become geospatial video providers.

Along these lines, we have developed *CITISENS* (CITizens as SENSors), a novel software service introducing a whole new concept in “hotspot” data generation and wildfire risk estimation, promoting active citizens participation in creating a wildfire citizens observatory. By exploiting the ubiquitous smartphone sensors, we introduce here new methods allowing the concerned citizens (firefighting volunteers, local residents, etc.) to effortlessly contribute fire-line measurements at a safe distance during a wildfire event by only using their phones. The final outcome is a dynamically formed VGI dataset representing the state of the fire shape in real-time. This can provide valuable input enabling the formation of an effective crowd-sourced Dynamic Data Driven Assimilation System (DDDAS) [20]. For wildfire spread simulation and improving its accuracy in predicting the hazard’s spatio-temporal evolution. Prior DDDAS approaches have demonstrated improvements in simulation accuracy under controlled conditions, especially when in-field sensors offer high-quality measurements of a wildfire’s front-line [21]–[24]. However, the limited efforts to validate wildfire DDDAS approaches using real-time, noisy measurements from real wildfires have not proved as successful [25]. This is why in [26] we have introduced a flexible data assimilation approach, where we first calibrate the mechanism producing the wildfire simulation model’s output before attempting to adjust its input parameters when significant simulation drift is observed.

We demonstrate here that *CITISENS* can fill the gap between high-quality but low-availability measurements from in-field sensors and high-availability but low-quality measurements from satellite sensors. To the best of our knowledge, such an approach where citizens contribute georeferenced remote targets on the fly, without the need for special equipment or excessive training, has not been implemented before in wildfire monitoring, or, as a matter of

fact, in any other field of environmental sciences. With the use of the *CITISENS* service, citizens can empower wildfire monitoring, wildfire course prediction and ultimately make possible the first crowd-sourced wildfires DDDAS approach.

The rest of the paper is organized as follows: Section II presents the workflow of the *CITISENS* service and describes the different user roles. Section III presents the software architecture of the two main applications implementing the service. Section IV first describes the georeferencing testing scheme employed, and then presents and discusses the testing results. Finally, Section V summarizes our findings and points to interesting future research directions.

II. CITISENS SERVICE WORKFLOW

CITISENS has been designed to support two distinct viewpoints: that of the citizen-volunteers who want to contribute wildfire reports, and that of the authorities (decision-makers) who wish to collect, visualize and evaluate these reports, and potentially use them to feed a crowd-sourced DDDAS for improved wildfires’ course prediction. To support all phases of this citizens-authorities win-win collaborative interaction, *CITISENS* integrates two distinct applications: A mobile app for hotspot generation and reporting and a desktop application for hotspot arrivals monitoring and decision support.

Figure 1 presents an activity diagram with the overall flow of information of a typical wildfire hotspot reporting process. Everything starts when a citizen (and service registered user) locates a visible wildfire hotspot and targets it using her smartphone camera. The *CITISENS reporter* smartphone application obtains the location, orientation, and elevation to compute the device’s view ray. Then it uses a Digital Elevation Model (DEM) to calculate locally on the phone the intersection of the calculated view ray with the topography and estimate the target’s geographic parameters (coordinates, elevation, distance from the citizen). All data is then made available to a decision center which can filter the contributed reports and proceed to either initiate a new wildfire simulation or to assimilate the reports into a DDDAS to update its prior wildfire spread predictions and associated confidence levels. Methods that can be used for hotspot reports filtering prior to assimilation and simulation initiation are beyond the scope of this work, but we discuss a course of direction on this topic in Section V. The decision-makers could also set up the service to alert the citizen reporters when the fire burn probability of their location is expected to exceed a certain threshold and possibly make available the wildfire model’s risk prediction maps. In Section II B. we explain how we calculate the fire burn probability maps after running Multiple Simulation Scenarios of the wildfire’s course.

The following sections provide detailed information about the *CITISENS* software applications and the distinct roles they are serving.

A. REPORTING HOTSPOTS

The mobile application, *CITISENS reporter*, was developed to allow citizens to photo-shoot, georeference, and report

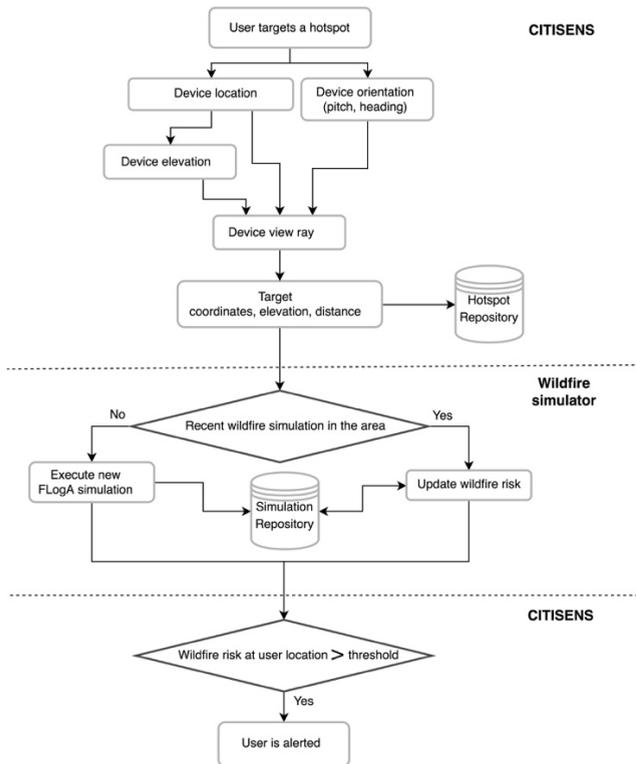


FIGURE 1. Activity diagram showing the flow of information in the *CITISENS* service. A citizen targets a wildfire hotspot using his/her smartphone camera and by using the device's view ray and a Digital Elevation Model (DEM). The *CITISENS* reporter application calculates the geographic information of the target (coordinates, elevation, distance from the citizen). The accumulated hotspot reports are then used to either initiate a new wildfire simulation or to be assimilated into an ongoing simulation and update probabilistic wildfire spread predictions. According to the outcome of the simulation, the authorities may decide to alert users based on the wildfire risk at their location.

a wildfire hotspot effortlessly using their smartphones. The most important requirement for its development was to support an intuitive workflow for end-users, a prerequisite for any Citizen Science-based approach aiming to generate wildfire hotspot reports of high quality and quantity. To that end, users have to center the wildfire hotspot on their screen and photograph it using their smartphone's camera; then the application does all the rest seamlessly locally on the phone to provide within milliseconds the geographic coordinates of their target. Figure 2 shows a screenshot of the main interface of the *CITISENS* reporter. After taking the photo, the geographic coordinates of the captured wildfire hotspot are produced and sent to the *CITISENS* server along with relevant information. Table 1 lists all the types of data collected from citizen reports.

To reduce the inherent measurements noise of smartphone orientation sensors, the *CITISENS* reporter supports a user-friendly calibration procedure, that users need to perform only once to maximize the georeferencing accuracy. The users can compare in real-time the actual view of their camera to the *virtual view*, as computed by the sensors of their smartphone, and manually eliminate any offset between the two. Figure 3 presents an example of the actual camera

TABLE 1. Information stored in the *CITISENS* server for each report.

| Description | Data type |
|-------------------------------------|-----------------|
| <i>ReportID</i> | <i>Int</i> |
| <i>Date & time</i> | <i>Datetime</i> |
| <i>Hotspot location</i> | <i>Double[]</i> |
| <i>Hotspot distance</i> | <i>Double</i> |
| <i>Photograph of the target</i> | <i>Byte[]</i> |
| <i>userID</i> | <i>Int</i> |
| <i>User location</i> | <i>Double[]</i> |
| <i>User location accuracy</i> | <i>Double</i> |
| <i>Device orientation</i> | <i>Double[]</i> |
| <i>Device orientation deviation</i> | <i>Double[]</i> |
| <i>Figure-8 mode detection</i> | <i>Boolean</i> |
| <i>Calibration</i> | <i>Boolean</i> |
| <i>Device orientation offset</i> | <i>Double[]</i> |
| <i>Using camera/virtual view</i> | <i>Boolean</i> |
| <i>Device info</i> | <i>String[]</i> |



FIGURE 2. The user interface of the *CITISENS* reporter mobile app. The geographic coordinates of the spot behind the cross in the middle of the screen are calculated instantly once the user hits the shutter button.

view and the corresponding virtual view of the same targeted location. In [27], we provide a short video demonstrating the *CITISENS* reporter interface and how the device's sensors calibration can be performed. We urge the readers to watch this short video to get a sense of the user experience before proceeding with the rest of the paper.

B. CITISENS OBSERVATORY

On the decision support side, the desktop application called *CITISENS* viewer takes full advantage of the potentially numerous citizen-generated reports while also providing the means to assess their validity. The data gathered is available server-side along with functionalities enabling decision-makers to view the wildfire hotspots from multiple angles with the option to discard unreliable reports, roll back in time and animate the reporting process at any speed.

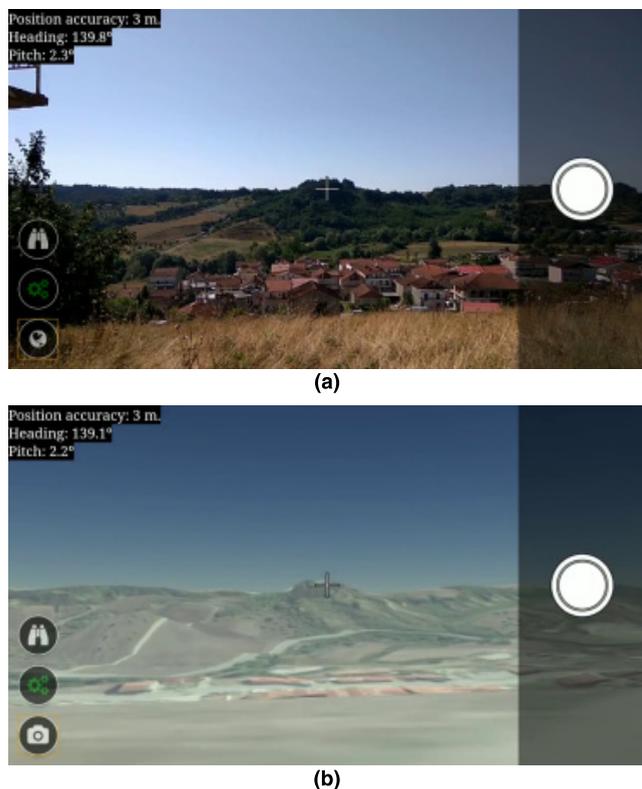


FIGURE 3. Screenshots from the *CITISENS reporter* mobile application depicting (a) the actual camera view and (b) the corresponding virtual view. The user can easily toggle between the two views to calibrate the device's orientation sensors. For details, please see the short video [27].

In addition, the *CITISENS viewer* application may call *FLogA* [28], our *Fire Logic Animation* software tool, which produces probabilistic spatiotemporal wildfire course predictions and provides simulations to interested parties for significant wildfire events in Greece, a sample of which can be viewed in the *FLogA* group YouTube channel [29]. The latest version of the *FLogA 2.0* software has been shown to produce very accurate wildfire course predictions when tested by independent wildfire researchers [30], [31].

FLogA is a unique tool in that it uses only publicly available data sources and can execute realistic wildfire spread simulations for a defined forest area anywhere in Europe by generating on the fly all input layers required for a simulation. The resulting simulation output comes in vector, raster or geoanimation formats, which visualize an overlay of the wildfire's probabilistic propagation map on top of the actual forest topography. Regarding the weather data, *FLogA* samples, around obtained reference values, the humidity, wind speed, and wind direction parameters, to generate and execute in parallel *Multiple Simulation Scenarios* (MSS). This is a unique feature of *FLogA* that has been designed considering that the wind is the most dynamic factor affecting a wildfire's behavior [32] and that its uncertainty should be reflected in the simulation results, thus enabling decision-makers to draw probabilistically quantifiable *what-if* conclusions. The different scenarios included in an MSS are defined by applying either parameters scanning (i.e., deterministic sampling)

or random sampling of the humidity, wind direction, and wind speed in a predefined range for each parameter. This range can be set by the user of *CITISENS viewer* or default to a user-configurable window around the reference values obtained from the METAR station closest to the forest area. The user of the *CITISENS viewer* also determines the number of simulation scenarios and the sampling method for each weather parameter.

The scenarios of an MSS are simulated using a full combinatorial experiment returning results for all possible parameter value combinations, regardless of the chosen sampling (deterministic or stochastic). For example, let's assume that the closest METAR station to a wildfire reports 10 m/s wind speed and 270° wind direction, and that the user wants to consider 3 wind speed scenarios and 3 wind direction scenarios with deterministic equidistant scanning in the ranges ± 2 m/s and $\pm 20^\circ$ around the METAR reference values. Then *FLogA* will form an MSS considering 3 wind speeds [8], [10], [12] m/s and 3 wind directions [250, 270, 290]°, which when combined will give rise to a $3 \times 3 = 9$ -scenarios MSS simulation.

The nominal (default) case is that each scenario of an MSS affects the burn probability of a cell of a defined forest equally. In this case, if *FLogA* executes an N -scenarios MSS, the results are combined to produce a wildfire risk probability heatmap where each scenario reaching a cell increases its burn probability by $1/N$, i.e., a cell that is reached say by 5 (out of the 9) scenarios in our previous example will have a predicted burn probability of $5/9$ ($N=9$). In the next section, we present a framework allowing us to update the scenario weights dynamically based on their predictive accuracy judged periodically against citizens-contributed hotspot reports used as ground truth.

Figure 4a shows the *CITISENS viewer* along with an example of the functionality enabled by using the reported wildfire hotspots as input to the *FLogA* wildfire simulator. A geo-animation of the wildfire's predicted spread pattern is produced, and color indicates the probability for an area to be affected at a specific future time instant.

C. A CROWD-SOURCED DDDAS FOR WILDFIRES

A major drawback in using static simulation reference parameters, e.g., for the wind speed, which are assumed to remain constant throughout the simulation, is their susceptibility to errors in conjunction with the inherent inaccuracies of the implemented fire behavior models. These imperfections may induce significant errors in simulation-based predictions as time progresses (a phenomenon known as simulation drift). This is the main reason why the concept of Dynamic Data Driven Assimilation Systems (DDDAS) was introduced [20]. According to the DDDAS paradigm, the predictive modeling process should exploit any available in-field measurements to benefit from updated sensor data. In this manner, the robustness of predictions of a DDDAS-enabled wildfire simulator can be improved, as the simulation becomes less vulnerable to severe drifting from the ground truth. This is especially

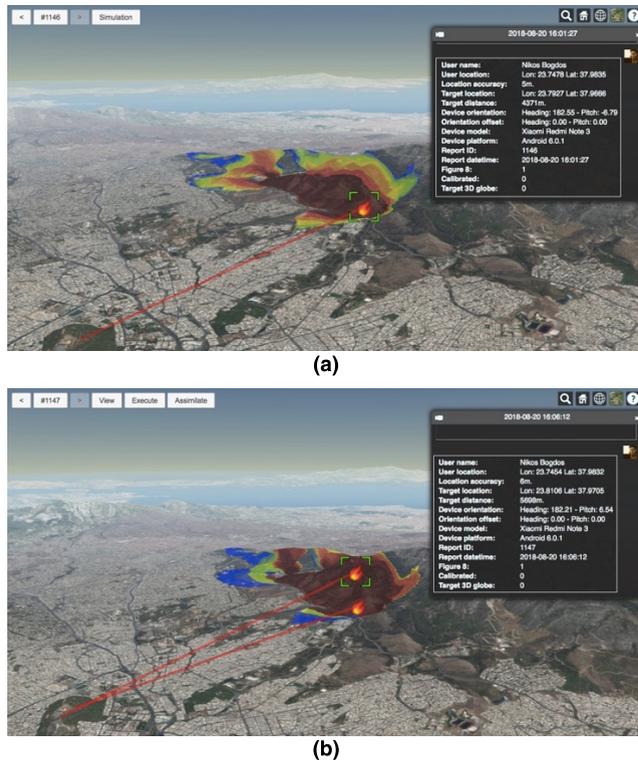


FIGURE 4. The user interface of the *CITISENS viewer* application. (a) A wildfire hotspot report arrives and *CITISENS* utilizes *FLogA* to produce a simulation of the wildfire risk heatmap patterns for the next 3 hours. Areas with colors closer to dark red have a higher probability of being affected. (b) A subsequent hotspot report arrives 5 minutes after the 1st report, and *CITISENS* assimilates it using the simulator's DDDAS. The wildfire risk probability map is updated based on the latest hotspot data, as explained in Section II-C.

important for large-scale wildfire simulations, which run the risk to produce imprecise predictions if not periodically calibrated.

CITISENS enables the unique capability to assimilate citizens-contributed hotspot reports into an ongoing wildfire simulation to improve its predictive accuracy and combat simulation drift. A visualization of the concept of dynamic assimilation of hotspot reports is provided in Figure 4, which shows the state of the predicted wildfire burn probabilities heatmap before (Figure 4a) and after (Figure 4b) the assimilation of a validated hotspot report.

The *CITISENS* wildfire DDDAS algorithm, as currently implemented, takes as input the defined MSS (utilizing initially uniform scenarios weighting) and reconfigures the weights dynamically according to the prediction performance of the individual scenarios measured against reported hotspot data contributed by citizens. The reconfiguration of scenarios weights works as follows: The user of the *CITISENS viewer* application (presumably a trusted public entity) observes the hotspot reports as they arrive and has the option to assimilate them at any time to update the wildfire simulation. Let's assume that at time t (time of an assimilation iteration) there exist R new trusted hotspot reports ready to be assimilated. Each hotspot report r was generated at some earlier time

$t_r \leq t$ that is after the previous assimilation iteration time. Given an MSS with S simulation scenarios, we want to create a dynamic data assimilation scheme that identifies the best performing scenarios and improves the prediction accuracy of the MSS based on this knowledge. Thus, we define for every scenario s at time t a performance metric $p_s(t)$ taking into account the distance, in space and time, from the wildfire's front for scenario s of all the R reports, as shown in Equation (1) below:

$$p_s(t) = 1 - \frac{1}{R} \sum_{r=1}^R (d'_s(r) \times (1 - \delta'(r))) \quad (1)$$

$$d'_s(r) = \frac{d_s(r) - \min(d)}{\max(d) - \min(d)} \quad (2)$$

$$\delta'(r) = \frac{\delta(r) - \min(\delta)}{\max(\delta) - \min(\delta)} \quad (3)$$

where $d'_s(r)$ is the normalized distance of report r from the front-line predicted for scenario s . Similarly, $\delta'(r)$ is the normalized delay of report r from the assimilation time over all the R reports. As suggested by Equations (2) and (3) linear scaling was used for the aforementioned normalization of the penalty terms.

As defined, $p_s(t)$ will return a value in the interval $[0, 1]$ scoring the performance of simulation scenario s at time t . This performance depends on the weighted sum of R factors which incorporate two types of penalties for each report; a space and a time penalty. For any given report r , the space penalty term increases with the distance between the hotspot report's location and the scenario's wildfire front prediction. Similarly, the time penalty considers the recency of every hotspot report in the current assimilation iteration. In this way, the sum in equation (1) accumulates the contribution of all the spatiotemporal report penalty products, which is then normalized (divided by R) to provide unit interval values, in the range $[0, 1]$, where zero (0) corresponds to no penalty; i.e. reports without any spatial offset from the wildfire's front line, and one (1) corresponds to maximal penalty of this assimilation iteration; i.e. hotspot reports at maximum distance from the wildfire's predicted front.

Although out of scope for this work, we remark that Equation (1) can be easily modified to also include a relative *trust* component so that we can assign to each hotspot report a relative confidence value. In Section V, we discuss how this generalization can be exploited.

The weight $w_s(t)$ of scenario s at time t assesses its performance relative to the other scenarios in the MSS. Specifically, it is computed using Equation (4) below

$$w_s(t) = \frac{p_s(t)}{\sum_{n=1}^N p_n(t)} \quad (4)$$

where $p_s(t)$ is calculated as in Equation (1), and N is the total number of scenarios in the MSS. Equation (4) ensures that each scenario weight is proportional to its performance and that $\sum_s w_s(t) = 1$. In this way, outlying simulation scenarios, with large penalties and thus weak performance,

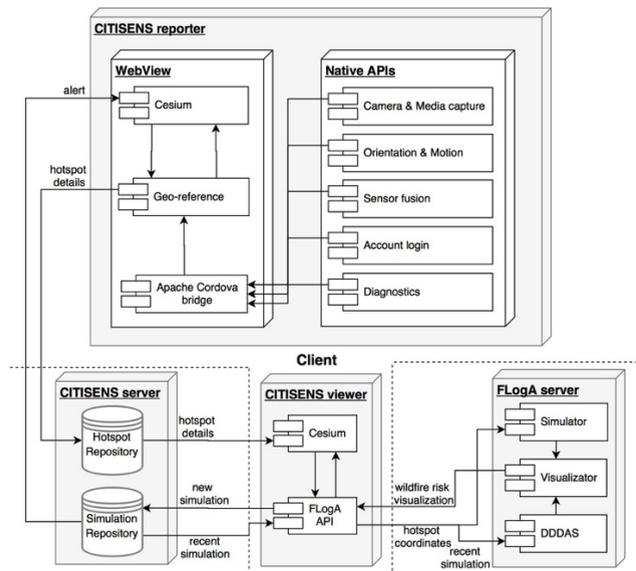


FIGURE 5. The software architecture of the *CITISENS* service. The *CITISENS reporter* client mobile application generates hotspots and stores them in the *CITISENS server* repository. The hotspots can be visualized using the *CITISENS viewer* web application which can call the *FLogA server* application to generate wildfire predictions or to assimilate hotspot reports into an ongoing simulation.

will have their weights mostly zeroed out, whereas in cases where most scenarios have similar performance their weights will be approximately equal.

Overall, the proposed DDDAS framework reconfigures the simulation output by evaluating the N scenarios and assigning weights to them according to their relative prediction performance based on reports received during the most recent assimilation iteration period. As a consequence, of updating the scenario weights a location's burn probability estimate may change, because each scenario s that burns a cell now increases the cell's burn probability, not by a constant but by a time-varying $w_s(t)$ quantity. We should remark that the proposed approach does not require a large amount of hotspot reports to work, as even a small number of reports per assimilation iteration can potentially affect the wildfire's risk probability map significantly. Also, note that the user of the *CITISENS viewer* can decide the weather parameter values that the wildfire simulation uses as reference inputs to generate the MSS around them. If the default reference values provided by the closest METAR station get overridden with values that deviate significantly from reality, the simulation predictions will notably start drifting in comparison to the arriving hotspot reports.

III. CITISENS SERVICE SOFTWARE ARCHITECTURE

In this section, we present the inner workings of the *CITISENS reporter* and *CITISENS viewer* applications. Figure 5 provides the complete view of the service's software architecture, and we use it below to discuss the functionality offered by each software component.

A. CITISENS REPORTER MOBILE APPLICATION

The *CITISENS reporter* application runs in the citizens' smartphones. Regarding mobile applications development, the available options are either to implement a web-based design, where the application would come in the form of a website running in the users' browser, or a native standalone application that is installed in the mobile devices. The former approach enjoys faster development cycles, especially when targeting multiple mobile platforms, while the latter can offer greater user experience and higher performance. A third approach that is gaining in popularity is to create a web-based design which is wrapped with a JavaScript bridge framework offering bindings to different native Application Programming Interfaces (API). In this way, it becomes possible to build mobile applications with native performance and user experience levels without suffering from the multi-platform development overhead. For the *CITISENS reporter*, we first implemented a proof of concept web application, and since our final objective was to build an Android application package (APK) to be installed on Android smartphones, we wrapped it using the Apache Cordova framework [33], which resulted in a web application running inside Android's WebView. In this way, we gained easy access to native APIs which could be exploited for better performance in situations where the web application suffered (e.g., for camera capture, orientation sensor readings, etc.). All the above are illustrated in the software components diagram of Figure 5.

In Figure 5, which presents the inner workings of the *CITISENS reporter*, we can see that the main software components running inside WebView are the Cesium and Georeference components. Cesium [34] provides a WebGL based 3D mapping engine, which also offers convenient access to high-quality elevation data [35] and works in conjunction with the Georeference JavaScript component, which continuously computes the device's view ray based on the Rotation Vector [36] provided by the Apache Cordova bridge. The Rotation Vector is a Kalman filter implementation fusing gyroscope data with measurements from the compass and the accelerometer [37] and is provided in the form of a virtual orientation sensor offering an absolute orientation relative to the magnetic north and gravity. To account for the magnetic declination from the true north, we apply a correction to the orientation vector by utilizing the latest iteration of the World Magnetic Model [38] through a magnetic declination estimation API [39].

By combining the pitch and heading information of the device's view ray with the terrain data obtained from Cesium, the Georeference component calculates the location where the view ray and the topography intersect. This calculation is performed by an iterative algorithm which, starting from the user's location, advances step-by-step the view ray until the ray's altitude becomes less or equal than the ground elevation at the same point. The iteration step was fixed to 30m, as this is the typical resolution of the STK World Terrain data.

The *CITISENS reporter* is currently available as an application for Android O/S devices. The O/S version should be 5.0 (Lollipop) or higher because the Android WebView was updated to support WebGL [40] at this O/S version, which is needed for the Digital Elevation Model (DEM) loading. In terms of hardware, the *CITISENS reporter* imposes minimal requirements; it requires a device with a camera, a gyroscope, an accelerometer, a compass, a GPS receiver, and an active internet connection. Moreover, a modern chipset with 2GB of RAM or higher is sufficient for smooth performance. Motivated by the recent trend of wearable devices, many OEMs have started to use motion and orientation sensors of greater accuracy, so in general, a modern chipset should provide not just smoother performance but improved georeferencing accuracy as well.

B. CITISENS VIEWER WEB APPLICATION

The *CITISENS viewer* web application plays a central role in delivering the functionality of the *CITISENS* service. It can run on any modern WebGL capable web-browser (mobile browsers included) and contains the Cesium and FLogA API JavaScript software components.

As for the *CITISENS reporter* mobile application, the Cesium component provides the 3D mapping engine through which decision-makers can visualize the reported wildfire hotspots by utilizing the hotspot repository located in the *CITISENS* server. Furthermore, by using Cesium's World Terrain Data the *CITISENS viewer* generates the elevation data layer which is passed to the FLogA API that interfaces with the FLogA server to offer wildfire simulation features. Finally, wildfire simulation results are stored and can be retrieved via the *CITISENS* server simulation repository.

As presented in Figure 5, the FLogA server comprises of the Simulator, Visualizator, and DDDAS software components. The Simulator contains a PHP script that uses the reference weather data to generate Multiple Simulation Scenarios (MSS). Each scenario is then executed by the simulation core, a C software component utilizing the fire behavior functions library fireLib [41]. Ultimately, the results of the MSS simulation are combined by the Visualizator component that generates wildfire visualizations in the form of KML layers [42] and passes them back to the *CITISENS viewer*. In the case of subsequent hotspot reports for the same area, the user of the *CITISENS viewer* has the option to use an existing wildfire simulation and assimilate the new reports (to update the burn probabilities) using the PHP-written DDDAS software component. The functionality of the DDDAS software component is discussed in Section II-C.

As a web application, the *CITISENS viewer* requires just a modern WebGL capable browser to run. Any hardware that can run such a browser will handle comfortably its relatively light workload, as all the georeferencing computations are performed by the *CITISENS reporter* mobile application while the wildfire scenario simulations are executed on the



FIGURE 6. The *CITISENS viewer* application displays a *reference report* which targets an easily identifiable location in the photorealistic virtual world. At the top right corner, we see the corresponding view from the camera.

FLogA server. As a result, the *CITISENS viewer* can run on any mobile platform as well.

IV. GEOREFERENCING ACCURACY

We describe next the testing method used to provide an estimate of the expected georeferencing accuracy delivered by the *CITISENS reporter* mobile application.

A. TESTING SCHEME

The two locations used for the georeferencing tests were: 1) in Attica, Greece, and specifically at the Hymettus [43] mountain which has experienced frequent wildfire incidents in recent years and is in close proximity to a large number of neighborhoods of the city of Athens, and 2) in Northern Greece, district of Ioannina, in the Pindus National Park (also known as Valia-Kalda [44]) that belongs to the Natura 2000 ecological network of protected areas.

To measure the geo-referencing accuracy, we built a dataset consisting of *reference* and *testing* reports pairs. As a *reference* we considered reports generated using the *CITISENS reporter* and captured using the virtual view; i.e., the 3D globe. We did not use the camera view for them, so the georeferencing was not affected by the orientation sensors' accuracy. We gathered *reference* and *testing* reports by spotting places in the physical world that could be easily identified using the virtual view and by using the 3D globe directly we obtained their true geographic coordinates and used them as a reference (ground truth) for the corresponding *testing* reports. Figure 6 presents an example of a *reference* report, as seen by the *CITISENS viewer* application, for an easily identifiable hill peak. The reference geographic coordinates of this peak were compared against those computed by the *testing* reports when trying to target it, but this time using the camera view (Figure 3).

Two users created the reference/testing reports dataset via the *CITISENS reporter* mobile application. The *CITISENS reporter* application was running on three Android smartphones with different hardware and software characteristics (see Table 2). As the georeferencing computations are based on the position and the orientation of the mobile device, the camera or screen pixel density did not affect the georeferencing accuracy in any way.



FIGURE 7. A user/tester targets a virtual hotspot (fire icon). The lines drawn represent a reference report (red) compared with a No calibration test report (blue), a Light calibration test report (green) and a Full calibration test report (yellow).

Our objective was to assess both the absolute georeferencing accuracy and how it fluctuates with the user-to-hotspot distance after applying different orientation sensor calibration options. This is why each reference report was compared to three different testing reports for the same target generated using the following calibration modes:

- *No calibration*: This is the cold-shot, instant georeferencing mode; the user notices a hotspot, opens the application and targets the hotspot using her camera as soon as possible.
- *Light calibration*: Same as above but now we assume that the user has already performed a “figure-8” calibration motion before making her first report. For more details please see the short video in [27].
- *Full calibration*: The user has completed the full calibration procedure to bring the virtual and the camera view close to parity before generating the first report. For more details, please see the video in [27].

Figure 7 presents an example of a comparison of a reference report with a testing reports triplet generated using the aforementioned three modes of hotspot reporting. The entire dataset generated consists of 101 reference reports and their corresponding 101 testing report triplets.

Finally, for each testing report, we measured the offset of the pitch and heading of the mobile devices (in degrees) and the offset of the georeferenced coordinates (in meters) against the corresponding reference report. In this way, we can compare the pitch, heading, and georeferencing accuracy of the three testing report modes.

B. RESULTS AND DISCUSSION

Based on the scheme described above, we generated reports for 101 hotspots to obtain a reference/testing reports dataset. The distance between the users and the targets ranged between ~150m. to ~6km in a straight line.

Figures 8 presents boxplots of the distribution of observed offsets in the magnetometer’s heading and the gyroscope’s pitch readings (in degrees) when using the three different

TABLE 2. The smartphone devices used for the georeferencing tests.

| | LG Nexus 5 | Xiaomi Redmi Note 2 | Xiaomi Redmi Note 3 Pro |
|---------------------|-------------------------|---------------------|-----------------------------|
| <i>Android OS</i> | 6.0.1 | 5.0.2 | 6.0.1 |
| <i>Chipset</i> | Qualcomm Snapdragon 800 | Mediatek MT6795 | Qualcomm Snapdragon 650 |
| <i>CPU</i> | Quad-core 4x2.3GHz | Octa-core 8x2.0GHz | Hexa-core 4x1.5GHz 2x1.6GHz |
| <i>GPU</i> | Andreno 330 | PowerVR G6200 | Andreno 510 |
| <i>RAM</i> | 2GB | 2GB | 3GB |
| <i>Gyroscope</i> | InvenSense MPU6515 | MTK | Bosch BMI160 |
| <i>Magnetometer</i> | AKM AK8963 | MTK | Yamaha YAS537 |

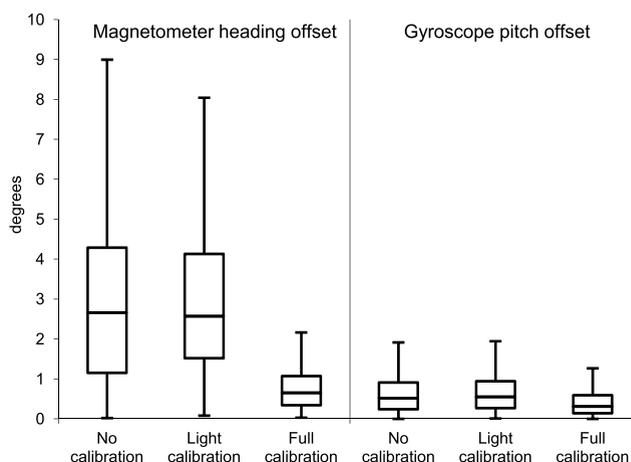


FIGURE 8. Boxplots (N=101) depicting the distribution of the magnetometer heading offset and the gyroscope pitch offset of the shots dataset (measured in degrees) under the three different phone calibration modes.

shot modes. The medians of the magnetometer’s heading offset were 2.66, 2.57 and 0.65 degrees for the *No calibration*, *Light calibration* and *Full calibration* shots respectively, while the corresponding medians of the gyroscope’s pitch offset were 0.52, 0.55 and 0.31 degrees respectively.

In general, the magnetometer’s heading is seen as having more significant offsets than the gyroscope’s pitch. The *Light calibration* shot mode did not decrease the median offsets over the *No calibration* shots, but in the case of the magnetometer’s heading, it reduces the offset’s interquartile range. As expected, the *Full calibration* mode exhibits the smallest heading and pitch median offsets and interquartile ranges.

Figure 9 provides the distribution of the distance offsets, i.e., the distance between the calculated and the true target coordinates (measured in meters). Again, *Light calibration* decreases the offset variance but does not improve the median

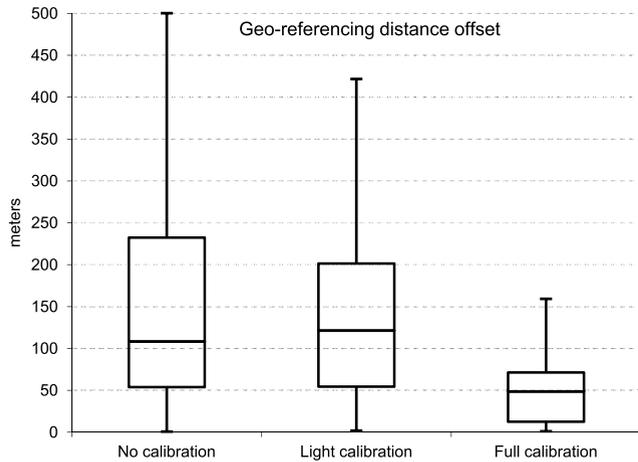


FIGURE 9. Boxplots (N=101) depicting the georeferencing distance offset distribution (in meters) for the shots dataset under three different calibration modes.

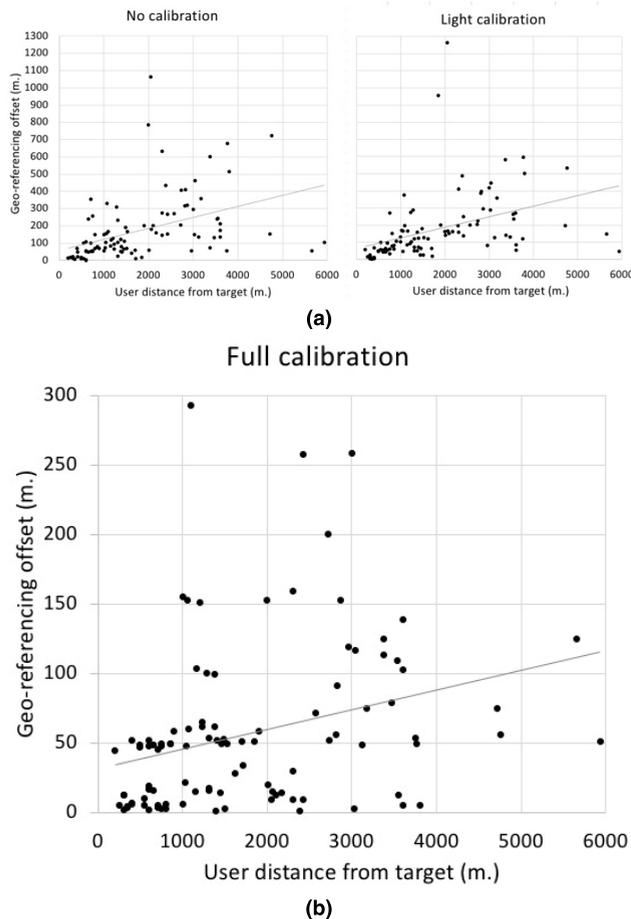


FIGURE 10. The georeferencing distance offset (in meters) in relation to the distance of the user from the reported hotspot. The number of points/shots is $N = 101$. (a) The linear regression model $y=ax+b$ coefficients are $a=0.06$ and $b=59.85$ for the *No calibration* mode and $a=0.06$ and $b=64.45$ for the *Light calibration* mode. (b) The linear regression coefficients are $a=0.01$ and $b=31.48$ for the *Full calibration* mode.

over the *No calibration* mode. However, the *Full calibration* mode provides a significant improvement again both in terms of variance and median offset (48.66 meters).

We consider the georeferencing accuracy measured, to be high for a Citizen Science based hotspot reporting scheme, especially when compared to available satellite hotspot detection services. It should be noted though, that the georeferencing accuracy depends on the mobile device used and on the distance between the user and the target. Figure 10 shows how the distance of the user from the target affects the georeferencing accuracy based on our reference/testing shots dataset. It is noteworthy that even though the *Full calibration* mode exhibits the best geo-referencing performance, even the *No calibration* mode is good enough most of the times, as it enables very fast, real-time remote hotspot georeferencing with potentially higher accuracy than satellite hotspot data products which currently offer up to 375m. spatial resolution with infrequent overpasses as a result of their polar orbits [45]. Moreover, the *No calibration* mode requires virtually no skills as the user only needs to target the wildfire hotspot with her camera keep it stable and take a regular picture shot.

The *No calibration* and *Light calibration* reporting modes exhibit a noticeable accuracy drift as the distance of the user and the targeted hotspot increases, which means that the validity of the resulting hotspot coordinates should be assessed carefully above a certain user-to-hotspot distance threshold. On the other hand, the *Full calibration* mode is not seen to be affected as much, which is also shown by the significantly decreased slope of the fitted linear regression models presented in Figure 10.

V. CONCLUSIONS

We have presented novel methods enabling the generation of quality Volunteered Geographic Information (VGI) from citizens during wildfire events and a VGI framework that can support a crowd-sourced Dynamic Data Driven Assimilation System (DDDAS) for effective wildfires course prediction. The main goal of this research is to develop methods empowering citizens to actively participate in combating a wildfire by allowing them to easily contribute accurate georeferenced wildfire hotspots remotely, at a safe distance, using their smartphone camera. The developed *CITISENS* service consists of a set of collaborating applications providing to decision-makers take advantage of streams of georeferenced hotspots that can be used to improve the quality of wildfire propagation predictions. Our tests demonstrate that the georeferencing accuracy of reports generated using the *CITISENS reporter* mobile application is on average much better than what satellite hotspot services offer today in their best case (polar orbit).

Currently, all contributed hotspot reports are considered of equal importance for data assimilation purposes, but in the future, we plan to introduce a *trust* component to relatively weight reports. Trust to a report can be assigned by different means: a) By training and then using a Convolutional Neural Network (CNN) to recognize the existence of a wildfire in the user’s photograph accompanying the report, b) by assessing the quality of the device’s sensory data based on statistical

properties of their measurements; e.g., the average deviation of the magnetometer readings, c) by promoting reports of particular users, e.g., registered volunteers in an area with a history of contributing quality reports. These developments would allow us to develop a fully automated, reliable DDDAS for wildfires without a human in the loop. Finally, the assignment of trust could also be crowd-sourced to remote registered users who are not in the region at the time of the wildfire but are willing to rate the reports using a stripped-down version of the *CITISENS viewer* application. These users could review and compare the discrepancies between the reports' virtual and camera views and contribute scores that would rate both the reports and their reporters. Multiple individuals could evaluate the same reports similar to the reCAPTCHA model [46]. Given sufficient historical data, reports could also be rated automatically based on the reporter user's ranking, thus promoting good practices.

Although discussing the operationalization of the service is beyond the scope of the paper, we mention here the relevant challenges we are currently addressing. The hardware and software requirements of the system need to be revisited to support multiple software platforms to widen the user base as much as possible; e.g., an iOS version of the *CITISENS reporter* application will be developed. Future versions of the mobile app will also support push notifications and alerts based on the predicted burn probability at the user's location. Moreover, we plan to engage with all interested bodies and the firefighting volunteer community in deploying the service in areas of Southern Europe, the US and beyond, and also in helping to train ordinary citizens on wildfire behavior basics to eliminate any participants' risk.

We strongly believe that the *CITISENS* service can improve the quantity and quality of the available wildfire hotspot data by enabling ordinary citizens to safely participate in a socially beneficial Citizen Science activity with minimal training. To the best of our knowledge, this is the first effort to enable civic involvement in a critical Citizen Science endeavor, which aims to allow concerned citizens to provide active support towards guarding their communities during catastrophic events without taking any risk. We believe that this line of research is very timely and contributes significantly to the mitigation of devastating consequences as those experienced by the victims and survivors of recent tragic wildfire incidents [2], [3].

REFERENCES

- [1] S. Stewart, V. Radeloff, R. B. Hammer, and T. Hawbaker, "Defining the wildland-urban interface," *J. Forestry*, vol. 105, no. 4, pp. 201–207, 2007.
- [2] F. Cage, G. Swann, and J. Holder. (2018). Greece wildfires: Satellite imagery shows devastation in Mati—Visual guide. The Guardian. [Online]. Available: <https://www.theguardian.com/world/ng-interactive/2018/jul/25/how-the-fires-in-mati-greece-spread-a-visual-guide>
- [3] Wikipedia Contributors. 2018 California Wildfires. Wikipedia, *The Free Encyclopedia*, Accessed: Oct. 7, 2018. [Online]. Available: https://en.wikipedia.org/w/index.php?title=2018_California_wildfires&oldid=862967357
- [4] C. O. Justice, L. Giglio, S. Korontzi, J. Owens, J. T. Morisette, D. Roy, J. Desclotres, S. Alleaume, F. Petitcolin, and Y. Kaufman, "The MODIS fire products," *Remote Sens. Environ.*, vol. 83, nos. 1–2, pp. 244–262, 2002.
- [5] N. I. Sifakis, C. Iossifidis, C. Kontoes, and I. Keramitsoglou, "Wildfire detection and tracking over Greece using MSG-SEVIRI satellite data," *Remote Sens.*, vol. 3, no. 3, pp. 524–538, 2011.
- [6] M. Zook, M. Graham, T. Shelton, and S. Gorman, "Volunteered geographic information and crowdsourcing disaster relief: A case study of the haitian earthquake," *World Med. Health Policy*, vol. 2, no. 2, pp. 7–33, 2010.
- [7] J. N. Sutton, L. Palen, and I. Shklovski, "Backchannels on the front lines: Emergency uses of social media in the 2007 Southern California Wildfires," in *Proc. ISCRAM Conf.*, Washington, DC, USA, 2008, pp. 1178–1204.
- [8] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: What Twitter may contribute to situational awareness," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2010, pp. 1079–1088.
- [9] M. Imran, S. Elbassuoni, C. Castillo, F. Diaz, and P. Meier, "Extracting information nuggets from disaster-related messages in social media," in *Proc. ISCRAM Conf.*, Baden-Baden, Germany, 2013, pp. 1–10.
- [10] V. Slavkovikj, S. Verstockt, S. Van Hoecke, and R. Van de Walle, "Review of wildfire detection using social media," *Fire Safety J.*, vol. 68, pp. 109–118, Aug. 2014.
- [11] J. L. Dickinson, B. Zuckerberg, and D. N. Bonter, "Citizen science as an ecological research tool: Challenges and benefits," *Annu. Rev. Ecol. Evol. Systematics*, vol. 41, pp. 149–172, Dec. 2010.
- [12] M. Sun, M. Li, F. Zhang, Z. Wang, and D. Wu, "Hybrid tracking for augmented reality GIS registration," in *Proc. Japan-China Joint Workshop Frontier Comput. Sci. Technol. (FCST)*, Nov. 2007, pp. 139–145.
- [13] N. Pfeifer, P. Glira, and C. Briese, "Direct georeferencing with on board navigation components of light weight UAV platforms," *Int. Arch. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 39, pp. 487–492, Aug. 2012.
- [14] A. Al-Hamad, A. Moussa, and N. El-Sheimy, "Video-based mobile mapping system using smartphones," *Int. Arch. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 40, no. 1, p. 13, Nov. 2014.
- [15] N. Alsubaie and N. El-Sheimy, "The feasibility of 3D point cloud generation from smartphones," *Int. Arch. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 41, pp. 621–626, Jul. 2016.
- [16] N. M. Alsubaie, A. A. Youssef, and N. El-Sheimy, "Improving the accuracy of direct geo-referencing of smartphone-based mobile mapping systems using relative orientation and scene geometric constraints," *Sensors*, vol. 17, no. 10, p. 2237, 2017.
- [17] Y. Li, "Integration of MEMS sensors, WiFi, and magnetic features for indoor pedestrian navigation with consumer portable devices," Ph.D. dissertation, Dept. Geomatics Eng., Univ. Calgary, Calgary, AB, Canada, 2016.
- [18] X. Niu, Y. Li, H. Zhang, Q. Wang, and Y. Ban, "Fast thermal calibration of low-grade inertial sensors and inertial measurement units," *Sensors*, vol. 13, no. 9, pp. 12192–12217, 2013.
- [19] A. Milosavljević, B. Predić, and D. Rancić, "Transforming smartphone into geospatial video provider," *Facta Univ., Ser., Autom. Control Robot.*, vol. 16, no. 1, pp. 1–13, 2017.
- [20] F. Darema, "Dynamic data driven applications systems: A new paradigm for application simulations and measurements," in *Proc. Int. Conf. Comput. Sci.* Berlin, Germany: Springer, 2004, pp. 662–669.
- [21] M. Denham, K. Wendt, G. Bianchini, A. Cortés, and T. Margalef, "Dynamic Data-Driven Genetic Algorithm for forest fire spread prediction," *J. Comput. Sci.*, vol. 3, no. 5, pp. 398–404, 2012.
- [22] K. Wendt, A. Cortés, and T. Margalef, "Parameter calibration framework for environmental emergency models," *Simul. Model. Pract. Theory*, vol. 31, pp. 10–21, Feb. 2013.
- [23] C. Brun, T. Margalef, and A. Cortés, "Coupling diagnostic and prognostic models to a dynamic data driven forest fire spread prediction system," in *Proc. ICCS*, 2013, pp. 1851–1860.
- [24] T. Artés, A. Cencerrado, A. Cortés, and T. Margalef, "Real-time genetic spatial optimization to improve forest fire spread forecasting in high-performance computing environments," *Int. J. Geograph. Inf. Sci.*, vol. 30, no. 3, pp. 594–611, 2016.
- [25] T. Artés, A. Cencerrado, A. Cortés, T. Margalef, D. Rodríguez-Aseretto, T. Petroliaqkis, and J. San-Miguel-Ayanz, "Towards a dynamic data driven wildfire behavior prediction system at European level," *Procedia Comput. Sci.*, vol. 29, pp. 1216–1226, Jan. 2014.

- [26] G. Bismpikis, V. Papataxiarhis, N. Bogdos, E. S. Manolakos, and S. Hadjiefthymiades. (2014). *SWeFS: Sensor Web Fire Shield for Forest Fire Detection and Monitoring*. [Online]. Available: <http://hdl.handle.net/10316.2/34013>
- [27] FLogA Group. (2018). *CITISENS Smartphone Geo-Referencing*. Accessed: Aug. 29, 2018. [Online]. Available: <https://www.youtube.com/watch?v=7uFWJTqHFac>
- [28] N. Bogdos and E. S. Manolakos. "A tool for simulation and geo-animation of wildfires with fuel editing and hotspot monitoring capabilities," *Environ. Model. Softw.*, vol. 46, pp. 182–195, Aug. 2013.
- [29] FLogA Group. (2018). *FLogA Group Channel*. Accessed: Sep. 28, 2018. [Online]. Available: <https://www.youtube.com/channel/UCMANFVtEK00bX3VtqlEJXyw>
- [30] M. Athanasiou, "GREECE: To measure and document wildfire behavior. . . and share the research findings with those who can apply the lessons," *Wildfire*, vol. 27, no. 2, pp. 32–37, Apr. 2018.
- [31] A. Miltiades, "Development of an optimal methodology for forecasting forest fire behaviour in Greece," Ph.D. dissertation, Dept. Geol. Geoenvironment, Nat. Kapodistrian Univ. Athens, Athens, Greece, Jan. 2016. [Online]. Available: <https://pergamos.lib.uoa.gr/uoa/dl/frontend/en/browse/1309360>
- [32] J. H. Scott, "Introduction to wildfire behavior modeling, national inter-agency fuels, fire, & vegetation technology transfer," Nat. Interagency Fuels, Fire, Vegetation Technol. Transf., Rocky Mountain Res. Station, US Forest Service, Fort Collins, CO, USA, Tech. Rep., 2012.
- [33] S. Bosnic, I. Papp, and S. Novak, "The development of hybrid mobile applications with Apache Cordova," in *Proc. 24th Telecommun. Forum (TELFOR)*, Nov. 2016, pp. 1–4.
- [34] CesiumJS. (2018). *Geospatial 3D Mapping and Virtual Globe Platform*. Accessed: Aug. 7, 2018. [Online]. Available: <https://cesiumjs.org/>
- [35] Cesium. (2018). *Introducing Cesium World Terrain*. Accessed: Aug. 7, 2018. [Online]. Available: <https://cesium.com/blog/2018/03/01/introducing-cesium-world-terrain/>
- [36] Android Developers. (2018). *Motion Sensors*. Accessed: Aug. 7, 2018. [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_motion
- [37] Android Open Source Project. (2018). *Sensor Stack*. Accessed: Aug. 7, 2018. [Online]. Available: https://source.android.com/devices/sensors/sensor-stack#sensor_fusion
- [38] A. Chulliat, S. Macmillan, P. Alken, C. Beggan, M. Nair, B. Hamilton, A. Woods, V. Ridley, S. Maus, and A. Thomson, "The US/UK world magnetic model for 2015–2020," *Nat. Geophys. Data Center*, Boulder, CO, USA, Tech. Rep., 2015. doi: [10.7289/V5TB14V7](https://doi.org/10.7289/V5TB14V7).
- [39] NOAA. (2018). *NCEI Geomagnetic Calculators*. Accessed: Aug. 7, 2018. [Online]. Available: <https://www.ngdc.noaa.gov/geomag-web/>
- [40] C. Marrin, "WEBGL specification," Khronos WebGL Working Group., Beaverton, OR, USA, Tech. Rep., 2011.
- [41] C. D. Bevins, "Firelib user manual and technical reference," Intermountain Res. Station, Forest Service, U.S. Dept. Agric., Fort Collins, CO, USA, Tech. Rep., 1996.
- [42] Google Developers. (2018). *Keyhole Markup Language*. Accessed: Aug. 7, 2018. [Online]. Available: <https://developers.google.com/kml/documentation/>
- [43] Wikipedia Contributors. (2017). *Hymettus in Wikipedia, The Free Encyclopedia*. Accessed: Aug. 2, 2018. [Online]. Available: <https://en.wikipedia.org/w/index.php?title=Hymettus&oldid=764496328>
- [44] Wikipedia Contributors. (2018). *Pindus National Park Wikipedia, The Free Encyclopedia*. Accessed: Aug. 2, 2018. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Pindus_National_Park&oldid=844042124
- [45] W. Schroeder, P. Oliva, L. Giglio, and I. A. Csizsar, "The New VIIRS 375 m active fire detection data product: Algorithm description and initial assessment," *Remote Sens. Environ.*, vol. 143, pp. 85–96, Mar. 2014.
- [46] E. Schenk and C. Guittard, "Crowdsourcing: What can be outsourced to the crowd, and why?" in *Proc. Workshop Open Source Innov.*, Strasbourg, France, vol. 72, 2009, p. 3.



NIKOS BOGDOS holds a M.Sc. degree in advanced information systems from the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, Greece, where he is currently pursuing the Ph.D. degree. He has participated as a Researcher to EU funded research projects related to wildfires modeling and simulation. His research interests include simulation, visualization, machine learning, mobile computing, citizen science, and their applications to environmental risk management.



ELIAS S. MANOLAKOS holds a Diploma degree in EE from the National Technical University of Athens, Greece, the M.Sc. degree from the University of Michigan, Ann Arbor, and the Ph.D. degree from the University of Southern California. He has served as a Visiting Scholar with the Harvard's Wyss Institute of Biologically Inspired Engineering, and as a tenured Faculty with the ECE Department of Northeastern University, Boston. He is currently a Visiting Professor with Northeastern University, Boston, and a Professor with the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens. He has authored or co-authored with his students, more than 130 publications in refereed journals and conference Proceedings. He enjoys multidisciplinary research and has played a leadership role in more than 20 funded research projects in the EU and the U.S. His research interests include predictive modeling using machine learning and advanced signal processing, high performance and embedded computing, and their applications in life sciences and environmental sciences. Prof. Manolakos has been elected and served for two terms for the IEEE SPS Technical Committees on Machine Learning for Signal Processing and on the Design and Implementation of Signal Processing Systems. He has also served in the Editorial Board of several journals, such as the IEEE TRANSACTION ON SIGNAL PROCESSING, the IEEE TRANSACTION ON NEURAL NETWORKS, and the IEEE SIGNAL PROCESSING LETTERS.

• • •