Border surveillance monitoring using Quadcopter UAV-Aided Wireless Sensor Networks

Sarra Berrahal, Jong-Hoon Kim, Slim Rekhis, Noureddine Boudriga, Deon Wilkins, and Jaime Acevedo

Original scientific paper

***Abstract*—In this paper we propose a novel cooperative border surveillance solution, composed of a Wireless Sensor Network (WSN) deployed terrestrially to detect and track trespassers, and a set of lightweight unmanned aircraft vehicles (UAVs) in the form of quadcopters that interact with the deployed WSN to improve the border surveillance, the detection and investigation of network failures, the maintenance of the sensor network, the tracking of trespasser, the capture and transmission of real-time video of the intrusion scene, and the response to hostage situations. A heuristic-based scheduling algorithm is described to optimize the tracking mission by increasing the rate of detected trespassers spotted by the quadcopters. Together with the design of the electrical, mechanical and software architecture of the proposed VTail quadcopter, we develop in this paper powerless techniques to accurately localize terrestrial sensors using RFID technology, compute the optimal positions of the new sensors to drop, relay data between isolated islands of nodes, and wake up sensors to track intruders. The developed VTail prototype is tested to provide valid and accurate parameters' values to the simulation. The latter is conducted to evaluate the performance of the proposed WSN-based surveillance solution.**

***Index Terms*—Border surveillance, WSN, Quadcopter, Network maintenance, Tracking, Heuristics**

I. Introduction

Monitoring national borders is remarkably one of the major concerns of any country wishing to protect and control its own infrastructure and reinforce public safety and economic well-being. In this context, specialized government agencies are created to promote security measures in order to control and monitor their country's borders. A variety of solutions have been used for detecting, tracking and recognizing illegal activities, unwanted infiltrations, and unauthorized trespassers (e.g., smugglers, terrorists, illegal immigrants, or hostile forces) and preventing from unlawful cross-borders activities. A commonly used practice is to physically build a wall or fence between two separate nations. The United States Customs and Border Protection (CBP) implemented an Integrated Surveillance Intelligence System to remotely monitor illegal crossings of the border with night-day cameras [1]. The Department of Homeland Security (DHS) has also made use of their resources to secure the northern and southwest USA borders by increasing the number of patrol agents, aerial coverage, and restricting points of entry. All of these approaches are generally requesting intensive human involvement which is tedious, error-prone, costly, and time-consuming.

Faced to the aforementioned limitations, Wireless Sensor Networks (WSNs) have emerged as a promising tool that assists authorities in monitoring the security of critical areas and properties, such as borderlines. These networks are provided as a collection of autonomous sensor devices that are able to create a multi-hop radio network, maintain a decentralized connectivity, and perform a pervasive detection and monitoring of physical and environmental conditions (e.g., temperature, motion, pressure) around them through cooperation and self-organization.

Among the main properties that should be satisfied by a good WSN-based application of border surveillance, one can cite: a) the efficient deployment of sensor devices to maximize the network coverage; b) the optimal use of computational, communication, and storage resources by sensors to maximize the network lifetime; c) the ability to rapidly detect and investigate nodes failure and network partitioning to avoid coverage holes; and d) the capability to minimize the cost associated to the repair of failures, the redeployment of damaged nodes, and the reconfiguration of the network.

To guarantee a continuous and pervasive monitoring of the borderline and allow the authorities to timely respond to intrusions, several problems associated to the deployment and use of WSN-based border surveillance systems need to be addressed. These problems range from the destruction of sensor nodes during their landing, to the appearance of communication and sensing holes (e.g., due to sensors' energy depletion, or the occurrence of transient transmission impairments). An economical and rapid intervention of the network administrator should be possible to detect and investigate failures in the network, and timely repair them (e.g., by dropping new sensors at precise locations).

Several border surveillance applications were proposed in the literature [2], [3], [4], [5], [6], [7], [8], showing either the use of Unmanned Aircraft Vehicles [9], or terrestrial Wireless Sensor Networks, but not a deep cooperation of both of them. Consequently, these existing solutions remain unable to guarantee neither a continuous and improved monitoring, nor a timely investigation and reaction to network failures.

In this paper we design a cooperative border surveillance application that integrates a set of lightweight Unmanned Aircraft Vehicles (UAVs), in the form of quadcopters, and a terrestrially deployed WSN. UAVs interact with the WSNs to improve the border surveillance, the detection and investiga-

Manuscript received 8 November, 2015; revised March 11, 2016.

S. Berrahal, S. Rekhis and N. Boudriga are with the Communication Networks and Security Research Lab., University of Carthage, Tunisia (E-mails: berrahal.sarra@gmail.com, slim.rekhis@gmail.com, noure.boudriga2@gmail.com).

J.-H. Kim, D. Wilkins and J. Acevedo are with the Discovery lab, Florida International University, USA (E-mails: kimj@cis.fiu.edu, dwilk014@fiu.edu, jacev030@fiu.edu).

tion of network failures, the maintenance of the network, the tracking of trespassers, and the response to hostage situations. To optimize the network maintenance as well as the tracking mission, we consider that the supervised border is subdivided into adjacent intervention areas, each one is monitored by a single quadcopter. In addition, a heuristic-based scheduling algorithm is proposed to prioritize the actions that should be taken by the quadcopter to increase the rate of successfully detected and captured trespassers. In this paper, the electrical, the mechanical, and the software architecture of the proposed VTail quadcopters are designed, and a prototype is developed and tested. In addition, green techniques are proposed to allow the quadcopters to accurately localize sensors, detect coverage holes, identify and investigate sensors' failures, fix coverage holes by dropping sensors after computing their suitable positions, relay urgent data between isolated island of sensors, wake up unreachable sensors to track intruders, and transmit real image capture of the trespassers being detected by terrestrial sensor nodes.

The contributions of this paper are four-fold: (1) Green techniques for the accurate localization of sensor nodes and the investigation of coverage problems by quadcopters are developed; (2) Through the integration of WISP (a battery-free and wirelessly powered platform for sensing and computation) to the wireless sensor network nodes, and thanks to the use of Dual-port nonvolatile memory (an EEPROM with RFID and Serial Interfaces), the configuration state of sensor devices can be powerlessly read or updated by the quadcopters, allowing to investigate several types of failures; (3) The developed quadcopters behave as enhanced mobile sensors, which cooperate with the terrestrially deployed sensors to enhance the accuracy of the trespassers detection and to optimize the network maintenance. They provides an economical and efficient response tool that allows to quickly respond to various types of incidents (e.g., sensor's coverage problems, sensors failures, trespassers detection) by intervening in the field to drop additional sensors at precise positions (since they are able to fly at very low altitude and speed, the coordinates of sensors' landing point can be determined with high accuracy), or to capture and transmit aerial photo and video footages of the zone being crossed by trespassers; and (4) The use of a scheduling algorithm together with a set of heuristics to select the best action that should be taken by the quadcopter to maximize the successful rate of trespassers' tracking.

The remainder of the paper is as follows. Section II gives an overview of several border surveillance approaches that were proposed in the literature. In section III, a thick strip border surveillance system based on a WSN is presented, showing the description of the network architecture, the nodes deployment strategy, and the target detection and tracking approach. Section IV provides a set of advanced functions provided by the quadcopters to improve the quality and accuracy of a WSN-based border surveillance system. Section V describes the design of the quad-copter device and its electrical, mechanical and software architecture. In Section VI, we discuss the system features validation through some conducted simulations. The last section concludes the paper.

II. State of the Art

Several recent proposals, based on the use of WSNs or unmanned aerial vehicles, have been proposed for the purpose of border surveillance. Nonetheless, the available works do not address in an efficient manner the requirements of such a critical mission as it will be shown in the following.

A. WSN-based border surveillance solutions

In [4] and [5] military surveillance and reconnaissance applications were proposed by deploying a flat and a homogeneous WSN is deployed along the monitored borderline. All nodes in the WSN have the same physical capabilities and are in charge of sensing the surrounding environment to detect vibration/seismic activity or magnetic anomaly, which indicates that intruders are crossing the border. In [10] a Self-healing Autonomous Sensor Network (SASNet) is proposed. The latter is a tiered WSN-based architecture for military surveillance applications that is built using a set of (short range) sensor nodes and wireless gateways. The gateway actsas an intermediary between the sensor nodes and the remote command station, and uses Beyond Line-Of-Sight (BLOS) communication in order to bridge the sensed data and alarms to a remote user. However, due to the limited and inaccurate information collected by sensors, false alerts are likely to be generated, due to animal crossing or environmental impact. The use of these WSNs does not allow a fast investigation and intervention in the case of warnings.

In BorderSense [3], a three-layer hybrid network architecture for border surveillance was proposed. The latter uses wireless multimedia sensor nodes attached to surveillance towers, mobile sensor nodes that roam throughout the monitored border, and scalar sensors (e.g., vibration sensors) deployed in underground or on the ground. These sensor nodes are randomly deployed along the border. A process could be used with a predefined spatial density. However, although [3] provides several advantages in terms of minimization of the human involvements and improvement of the detection accuracy of the border surveillance systems. It does not provide mechanisms to either rapidly detect and investigate nodes failures, or to minimize the cost associated to the repair of failures, the redeployment of damaged nodes, and the reconfiguration of the network.

In [11] a border surveillance approach using a stationary WSN as a Sensor Fence, was proposed. These sensors are in charge of detecting and tracking multiple targets crossing the border based on a fusion-driven decentralized sensor scheduling scheme. The latter aims to provide an energy-efficient track estimation by enabling dynamic space-time clustering of powerful sensing nodes around the estimated positions of several moving targets. A Probabilistic Finite State Automata (PFSA) approach is run on each sensor node to control the communication and sensing devices in an energy-efficient manner.

B. Toward the needs of quadcopters for enhancing border surveillance

Typically, in a WSN-based border surveillance system, sensors are linearly arranged due to the linear nature of the borderline, creating a specific class of these networks. To provide an economical large-scale monitoring of the borderline, even in critical environments whereimpractical for humans to be present, an aerial vehicle is generally used to drop sensors. However, several issues are facing the use of these networks. First, the physical location of sensors' landing points cannot be determined with a high accuracy, even if advanced models for the controllable and random deployment of nodes thrown from the air, are used [12], [3]. Therefore, multiple coverage holes may appear. In addition, since sensors are thrown from aircraft vehicles, some of them could be damaged during landing. Second, after an operational period of time, some sensor nodes may go out of energy, and their sensing range may be affected by the variation of the vegetation surrounding them. Third, over time, threats affecting the monitored zone could vary, making the density of the nodes within the vulnerable area insufficient to guarantee an accurate good detection and tracking. Fourth, some transient troubles could occur (e.g., rainy weather), creating coverage and communication problems (e.g., isolated islands of sensor nodes). Fifth, false alarms triggered by these sensors require unnecessary human intervention which is in turn expensive and even dangerous [13]. However, due to their sensitivity, alerts generated by sensor nodes should be timely exchanged and forwarded through the network to the control center, otherwise trespassers could cross the border undetected.

Quadcopters offer unique capabilities and are very flexible devices in terms of the advantageous tasks they can perform including hovering (at lower altitudes) above a point of interest in the monitored area (especially in narrow and unreachable areas). Therefore, a better approach to enhance border surveillance missions would consist in using terrestrially deployed WSNs together with Unmanned Aerial Vehicles (UAVs) in order to enhance the quality of detection, guarantee a continuous and pervasive monitoring of the borderline, and provide an economical and rapid detection and network failures, investigate the failure problems, and intervene in time to repair them.

C. Use of Unmanned Aerial Vehicles (UAVs) for border surveillance

Several other works in the literature focused on the design of lightweight quadcopters, among which we cite the most important.

Authors in [14] designed an UAV to enter an enclosed area of unknown dimensions to find and obtain an object of known properties while evading security detection. The UAV or Quadcopter in this case uses a method of randomly searching an area for the target while mapping its environment and making a return route. An attachment and delivery system is made with a box attached to the bottom of the quadcopter. The box opens in the front via a pulley mechanism to drop an object on the ground, and has also a sticky tape attached to the bottom of the box to retrieve objects. The UAV will know its environment and will not require mapping to return home. The UAVs method of finding its target is based on a random search, and the proposed pickup/delivery system is not suitable. In fact, the use of a sticky tape is unreliable for give for solid attachment, as dropping a sensor out the front end of a box could, not only damage the sensor, but also place it in a wrong orientation.

In [15] a helicopter capable of flying autonomously using a vision-based algorithm to pickup and drop-off a designated payload, is designed. The helicopter travels to a GPS way point with an accuracy of one meter horizontal radius and half a meter vertical radius. The helicopter is controlled over a Wi-Fi network via a ground computer. It has an approximate payload of 19lbs, and it can pick up a target with a hook. However, the hook for the payload carrying is not efficient, as the servo system needs to be of a high torque value to hold 19lbs, and requires an important source of energy. The hook system is not suitable for targets that have a solid connection to the surface underneath it. The pan tilt system also has a slower reaction time to pick up object while flying towards it.

Authors in [16] took interest to mechanical system modeling, design, estimation and control for aerial grasping. The design of the grabber implements a quadrotor with a spring-loaded mechanism with hooks to penetrate and grab the surface of a porous object in nature. Rigid body dynamic models are used to verify grasp and control the overall movement of the quadcopter with or without a payload. However, the quadrotors were designed to be controlled by a computer and not on-board. The gripper can successfully hold objects such as wood, foam, and fabric.

In [17] the design of a swarm of quadcopters used to build structures with magnetic segments, is proposed. Each quadcopter is fitted with a grabbing mechanism and are all controlled with a motion tracking system. While the quadrotors can perform the tasks without the motion tracking system, the claw grabbing mechanism works only for a specific subset of objects with an increased need for accuracy.

The authors of [18] described the design of a quadcopter which is controlled by a motion capture system and uses three arms attached to its base to grab objects on the ground. Each arm has two degrees of freedom and extends towards the ground. However the quadcopter requires manual liftoff and can continue to fly using the autopilot. The autopilot causes to move the quadrotor in an oscillatory fashion. The arms are not ideal for carrying objects since the farther they extend, the weaker they become. The ends of the arms are also in the form of hooks that are not equipped with a real grabbing mechanism.

The work in [19] showed the ability of one to three helicopters to carry a payload with a cord attached between the helicopter(s) and the object. Control algorithms were designed so that if three helicopters are used to carry an object, they would evenly distribute the weight. However, the method of carrying an object requires an initial setup by an outside system (or human) to tie the rope to the object. This method is not optimal for an autonomous system of helicopters.

D. Cooperative WSNs and UAVs for Border Surveillance

In [8] a quadcopter UAV is designed to monitor the border area and locate and track intruders (using GPS), to guide military troops, and to shoot videos of the occurred events from a long distance. The collected data will be received by the processor and transmitted to the controller via zigbee. The controller monitors the quadcopter device via remote IR and controls its flying. The multimedia data is transmitted via a Wireless camera to be analyzed and recorded. The designed quad-copter would facilitate the intervention in unreachable areas to minimize the risk of losing human lives. However, the only use of quad-copters to survey very long borders makes the solution unscalable and unable to provide a continuous surveillance, unless a high number of long distance quad-copters are used simultaneously all the time. This would make the surveillance unpractical and highly expensive.

In [20] a path planning problem for a team of unmanned aerial vehicles patrolling a network of roads and pursuing intruders using Under Groud Sensors is proposed. Since this problem is shown to be intractable and NP_hard, a heuristic algorithm that aims to coordinate the UAVs during surveillance and pursuit is provided. In this algorithm the revisit deadlines are used in order to schedule the vehicles' paths nominally. The algorithm uses detections from the sensors to predict possible intruders' locations and plans the paths for the UAVs by minimizing a linear combination of missed deadlines and the probability of not intercepting intruders. Finally, the heuristic algorithm interacts with the sensor nodes to trigger the capture of an image of the intruder by a loitering UAV.

In the solution presented in [21], we had described a border surveillance solution built using a set of lightweight Unmanned Aircraft Vehicles (UAVs) that interacts with a terrestrially deployed WSN in order to improve the border surveillance through a set of advanced functions including the detection and investigation of network failures, the maintenance of the deployed sensor network, the tracking of trespassers illegally crossing the monitored area, and the response to hostage situations. Compared to [21], the current work considers the following aspects: (i) the subdivision of the monitored area into multi-intervention areas to facilitate and improve the tracking and the maintenance tasks of the quadcopter while reducing the energy consumption;(ii) the development of a scheduling algorithm together with a set of heuristics to select the best action that should be taken by the quadcopters, and therefore in order to increase the rate of successfully tracked and spotted trespassers; and (iii) the description of the electrical and mechanical architecture of the developed quadcopter.

III. A THICK BORDER STRIP SURVEILLANCE WSN

In this section, we describe the architecture of the terrestrial border surveillance wireless sensor network, the features provided by each type of node, and the nodes deployment scheme.

A. Network architecture

We consider a thick linear and hierarchical wireless sensor network, as described in Figure 1, which integrates three types

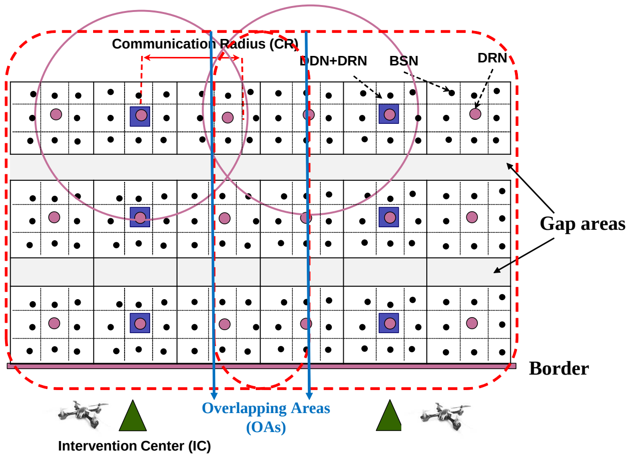


Figure 1: The WSN based subsystem architecture

of sensors: Basic Sensing Nodes (BSNs), Data Relay Nodes (DRNs), and Data Dissemination Nodes (DDNs). The BSNs are elementary sensor devices forming the first layer of the architecture. They are low powered and resource impoverished nodes used for the detection of moving objects, the alerting, and the cooperative relaying of messages to/from the second layer (i.e., the DRN nodes). The DRNs are resource rich nodes equipped with powerful energy and communication resources. They form the second layer of the network, and are responsible of collecting alerts from the different BSN nodes in their vicinity, and cooperating with neighbor DRNs to forward these alerts to the third layer of the network (i.e., the DDNs). The DRNs are also in charge of discovering the different deployed BSNs, and scheduling their activity and managing routes towards them. The DDN form the third network layer. They represent a set of sink nodes in charge of collecting data from their neighbor DRNs, pre-processing and aggregating them, and forwarding them to the Network Control Center (NCC).

The wireless sensor network we are designing follows a thick linear topology. In this context, the DRNs and DDNs are deployed linearly through the border, while the BSN are distributed around the DRNs and delimited by two lines. Gap areas are introduced between two strips to enlarge the width of the monitored area.

B. Nodes deployment scheme

To provide an accurate detection and tracking of trespassers by the border surveillance wireless sensor network, the methods and techniques used for the deployment of nodes should achieve a maximum coverage of the entire supervised area, and should allow sensors to form a connected communication network.

Let R_c^d and R_c^b denote the communication range of a DRN and a BSN, respectively. We denote by R_s^b the sensing range of a BSN. We assume that the thick line border is a rectangle of length L and width W, that we partition into equal squares. Each one of these squares of width W is partitioned into a set of equal subsquares. We place a DRN in the central subsquare of each square, and a BSN in all the remaining subsquares.

Since DRNs are deployed in a linear manner, they must be mutually connected to guarantee radio connectivity. In this context, every DRN node should have at least two neighbor DRNs in its communication range (R_c^d), and the length of every square should be lower or equal to half the DRN's communication range ($W \leq \frac{1}{2} \times R_c^d$). The minimum number of DRNs to be deployed would be equal to $2 \times L/R_c^d$.
The dimension of a subsquare is chosen with respect to the sensing coverage of a sensor node, so that every point in a subsquare will be in a sensing disk of a sensor. Typically, the deployment pattern to use should guarantee that in every subsquare a BSN is placed. To guarantee that successive BSN sensors are able to communicate with each other, while providing full monitoring, we assume that $R_c^b \geq 2 \times R_s^b$. For each square the DRN is placed at the central subsquare, while at least a BSN is placed at each one of the remaining subsquares. DDNs, representing gateways to the NCC are placed at a regular interval after a predefined set of DRNs.

In practice, BSNs and DRNs are randomly deployed from the sky using aerial vehicles. We propose to use the scheme proposed in [12] for the aerial deployment of a 3-layer hierarchical WSN capable of monitoring a 2D area. The position of the sensors' landing point is determined with respect to the wind speed vector of the aircraft vehicle (transporting the sensors), the wind forces experienced by the sensors thrown from the air, and the interval separating two successive droppings times.

As depicted in Figure 1, the optimal position of every BSN, DRN, or DDN is at the central of a subsquare. Even if the technique proposed in [12] allows to control the error related to the variation of the landing patterns, some sensing coverage holes may occur due to the sudden variation of the wind velocity and the geographical features of the landing area. These holes should be detected and eliminated after deployment, otherwise they would allow invasive intruders to cross the thick line evade detection.

C. Target tracking

The target tracking is a collaborative task involving all nodes in the network. When a trespasser crosses the borderline and enters in the sensing range of a BSN. The latter detects it, generates an alert, and forwards it to the DRN of the same square. As long as the trespassers continue to move and cross areas covered by sensors, neighbor BSNs, which are located in the same or adjacent square, generate and forward their alerts to the DRNs. These alerts, which are subsequently forwarded by DRNs to the NCC, provide measurements of the successive locations of the intruder. Based on the received measurements, the NCC traces back the trajectory of the intruder, and determines its velocity and direction. Based on that trajectory, it predicts the next zone to be crossed within a next predefined period of time. The NCC informs the DRNs located in the predicted trajectory, and instructs them to wake up BSNs available on that trajectory. As long as new alerts are received by the NCC, the difference between the predicted and the observed trajectory is computed and used to enhance the predicted trajectory in the next time period.

D. Need for proactive and reactive techniques for an accurate detection and tracking

An efficient detection and tracking of intruders by sensor nodes requires that the monitored area is totally covered, and the BSNs are able to communicate with their neighbors and generate routes toward DRNs. The existence of sensing or communication coverage holes in the network, would prevent either the detection of trespassers and the generation of alerts, or the routing of received alerts toward the DRN. Holes in the network appear due to the following reasons:

- As sensors are thrown from aircraft during deployment, some of them could be damaged.
- Sensors are prone to faults and malfunctioning. Calibration drifts, for example, which increase throughout the sensor lifetime, could decrease the detection accuracy.
- A sensor could run out of energy depending on the quantity of detected events and generated and forwarded alerts.
- Due to modifications on the environment, under which a sensor is deployed (e.g., vegetation, temperature, noise), or the occurrence of transient troubles (e.g., rainy weather), several irregularities could arise on the sensing and transmission range, contributing to the creation of coverage holes.

In addition to holes, threats affecting the monitored zone could vary over time, requiring sometimes to increase the density of nodes within the vulnerable area to guarantee a good detection and tracking.

To guarantee a good quality of detection and tracking, a WSN-based border surveillance applications should allow the prediction, detection and identification of a wide set of sensor faults, the tolerance of the monitoring system to these faults, and the ability to recover from them. Energy consumption of a sensor node, for example, should be monitored, and the instant of failure should be predicted based on the history of resources consumption. A replacement procedure should be developed so that the NCC can proactively respond to failure by replacing the sensor before it becomes faulty. The efficiency of the replacement procedure depends on the size of the border area, the mean time separating two successive faults, and the frequency of events generation.

IV. Border surveillance quality improvement: use of quad-copters

In this section we introduce the use of unmanned aircraft vehicle platforms of quadcopters, to interact with a terrestrially deployed wireless sensor network and be used as a tool for the proactive response and investigation of faults occurring on the deployed sensors.

A. Quadcopter objectives

The main objectives of the quadcopter are: a) Localization of terrestrial sensors and detection of sensing and transmission coverage holes; b) Detection of several types of nodes failures, such as battery depletion, and routing problems; c) Transporting and dropping of lightweight sensor nodes; d) Correction of coverage holes by dropping sensors at suitable positions; e) Relaying of data between isolated island of BSN nodes, and between isolated DRNs and the NCC; f) Tracking of objects crossing the border by capturing and transmitting real-time video of the intrusion area; g) Waking up of isolated sensors to track mobile trespassers and trace their trajectory.

The use of quadcopters for enhancing the quality of border surveillance offers several advantages. First, it is able to fly over hazardous and risky areas, allowing to prevent the loss of human life. Second, it is an inexpensive platform that can be built from scratch using components available in the market, and easily assembled due to its non-complex mechanical architecture. Third, it does not rise safety and legislative issues thanks to its small dimension and ability to fly at very low altitude. However, we should mention that such advantage is granted to quadcopter unless some conditions are satisfied including: (i) The quadcopter should not fly over or within 150 meters of a congested area or an organised open-air assembly of more than 1,000 persons; (ii) The quadcopter should not fly within 50 meters of any vessel, vehicle or structure which is not under the control of the person in charge of the aircraft; and (iii) The quadcopter should not fly within 50 meters of any person[1].

B. Quad-copter design requirements

To achieve the aforementioned objectives we design a quad-copter that has the following characteristics. First, it represents a mobile sensor that is able to communicate with the WSN deployed on the ground. Second, it is able to perform a long distance communication with the NCC using a packet oriented service connection (such as 3G or rural mobile network) to receive navigation data, transmit the locally collected data, and relay data between isolated nodes. Third, it can be remotely piloted and controlled over thousands of meters, and is able to fly at a tunable altitude (up to several tens of meters). Fourth, it is equipped with a set of on-board sensors for safe flying (e.g., Attitude and Heading Reference System (AHRS), GPS receiver, 2D LIDAR obstacle detection, compass, and accelerometer). Fifth, it has an attached camera to capture high-resolution images and real-time videos of the intrusion scene. The images and video will be processed by a computer vision algorithm to minimize the rate of false alerts. Sixth, it can transport and drop tiny sensor nodes.

In order to reduce to the maximum possible the overhead of the energy required by terrestrial sensors to respond to the requests generated by the quadcopter, we introduce the use of Radio Frequency energy harvesting techniques to powerlessly localize sensors and collect and modify configuration data. We integrate to every sensor a Wireless Identification and Sensing platform (WISP) which is a programmable battery-free sensing and computational platform [22], [23] that can be powered and read by a standards compliant Ultra-High Frequency (UHF) Radio Frequency Identification (RFID) reader. A WISP uses an ultra-low-power programmable micro-controller powered by RF energy to encode its unique ID and additional data in order to perform sensing and computation tasks [24]. We equip the quad-copters with long-range RFID readers to read the WISP tags deployed on the WSN nodes located on the ground.

[1]http://blog.oscarliang.net/laws-quadcopter-drones-uk/

C. Coverage holes detection and maintenance

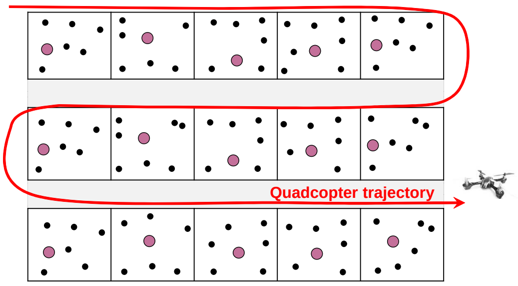
To detect sensing and transmission coverage holes, the quadcopter needs to compute the current positions of neighbor BSNs and DRNs within a predefined geographic area. To compute the coordinates of a sensor node, say s, the quad-copter proceeds as described in Figure 2. First, it computes its coordinates (x_p, y_p) at position p thanks to the use of an embedded GPS receiver, or using triangulation with the 3G network access points. Second, while flying at a constant speed (from position p to another position q) and in parallel to the upper boundary of the strip representing the thick border, the quadcopter performs two successive measurements of the distances d_p and d_q (between itself and the sensor node) at two positions p and q, respectively. Knowing its speed and the time difference between the two instants of measurements, the quadcopter computes the distance separating the two positions p and q. The coordinates (x_s, y_s) of the sensor s are computed by solving the two equations $x_s^2 + y_s^2 = d_p^2$ and $(d_{pq} - x_s)^2 + y_s^2 = d_q^2$. We obtain x_s and y_s as follows. Since the quadcopter is always flying at the upper boundary (see Figure 2), the y_s' value cannot be negative.

$$\begin{cases} x_s = (d_p^2 - d_q^2 + d_{pq}^2)/(2d_{pq}) \\ y_s = \sqrt{(d_p^2/2d_{pq})} * (d_p^2 - d_q^2 + d_{pq}^2) \end{cases} \quad (1)$$

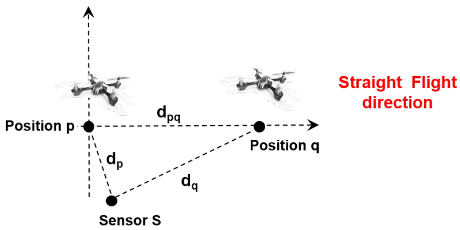
To compute the distances d_p and d_q the quadcopter performs an RFID based localization by estimating the physical distance separating it to the passive WISP tag embedded in the sensor. Several techniques can be used for the distance estimation such as Radio Signal Strength or the time difference of arrival. In [25] a technique that combines the advantages of acoustic location (high degree of precision and simplicity) and the use of RFID technology (powerless computation and unlimited lifetime) is proposed to provide a high accuracy in comparison with the existing techniques. Using it, the WISP tag embedded to the sensor will be equipped with an acoustic tone detector. Once interrogated, the WISP powerlessly generates an ultrasound signal after the reception of an acoustic beacon, measures the acoustic Time of Flight, and stores the latter in the tag to be read by the RFID reader integrated in the quad-copter.

After computing the ground sensors' positions (both BSNs and DRNs), the quadcopter checks if: a) the distance separating two neighbor DRNs does not exceed R_c^d; b) each BSN has the required number of neighbor BSNs; and c) the distance between two neighbor BSNs or between a BSN and a DRN does not exceed R_s^b. If one of these conditions is not satisfied, the quad-copter drops additional nodes in the adequate zone to overcome coverage and connectivity problems.

Having computed the positions of the two horizontal DRNs say D_i and D_j, which are unable to communicate together, the quadcopter computes the position of the new DRN to be dropped, so that it will be at the intersection of the



(a) Quadcopter's Followed Trajectory



(b) Quadcopter's Trajectory

Figure 2: Quad-copter-based localization correction

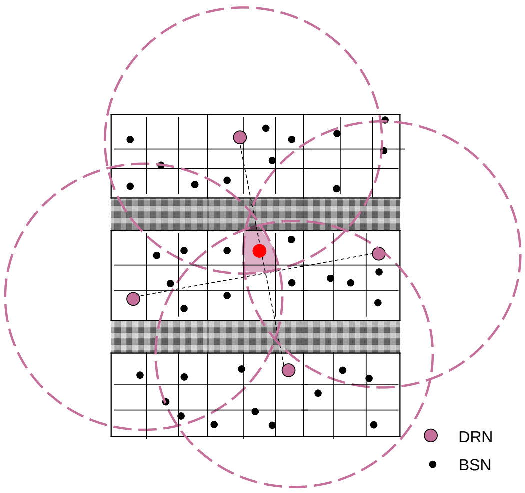


Figure 3: Deployment of DRNs

communication coverage areas of D_i and D_j and inside the squared zone as depicted in Figure 3. If possible, the position of the new DRN will be also in the communication coverage of one or two vertical neighbors, so that DRNs located at different strips could communicate together to relay alerts from a border strip to another (this feature is needed during tracking, so that a DRN could ask its vertical neighbors to wake up sensors located in the trajectory of the intruder).

To avoid the appearance of isolated islands of nodes showing the existence of sensing and transmission coverage holes between candidate group of nodes, the deployment of additional BSNs should guarantee that each BSN has at least one BSN in its sensing range R_s. To respond to coverage holes, the quad-copter drops new BSNs as follows. First, it determines the nearest BSNs of both isolated islands and selects one of them. Second, it drops the new BSN as far away as possible from the actual point, provided that the

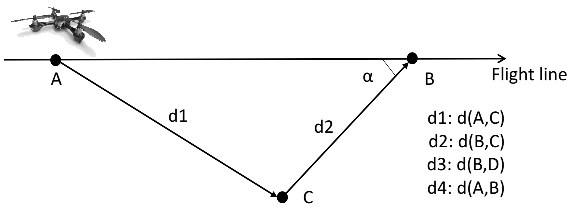


Figure 4: Localization error estimation

distance separating them will be equal to $R_s - \epsilon$ where ϵ is a small value representing the estimated error in computing the distance between nodes. The quadcopter repeats the same operation until no coverage hole exists.

D. Localization error estimation

Since the determination of the distance separating the quadcopter to the tags requires that the WISP computes the acoustic Time of Flight, through two successive interrogations of the RFID tag, a non negligible period of time is required to determine that distance. During the interrogation of the WISP, the quadcopter moves using a static speed. Therefore, during the two successive (interactions with the WISP, the geographical coordinates of the quadcopter may vary slightly, which may lead to an inaccurate estimation of the sensor's location. We estimate in this section the localization error as follows:

Let t be the time elapsed between the first and the last wireless communication with the WISP, s be the propagation speed of the signal, and v be the speed of the quadcopter. As the quadcopter speed is constant, The distance d_4 can be computed as $d_4 = t \times v$.

We denote by $d_3 = (d_1 + d_2)/2$ the effective measured distance (between the quadcopter and the sensor) due to the movement of the quadcopter between the two interrogations of the WISP, and let d_2 be the distance supposed to be measured by the quadcopter. Using the Pythagorean theorem, we obtain the following equalities, assuming that the quadcopter is enable to stimate the Angle of Arrival α of the signal received from the WISP.

$$d_1 = \sqrt{(d_4 - d_2\cos(\alpha))^2 + (d_2\sin(\alpha))^2} \tag{2}$$

$$d_3 = (\sqrt{(d_4 - d_2\cos(\alpha))^2 + (d_2\sin(\alpha))^2} + d_2)/2 \tag{3}$$

Since $d_3 = t \times s$, we can obtain the value of d_2 in function of t, s, v and α, and estimate the error e related to the computation of the distance between the quadcopter and the sensor as $e = d_3 - d_2$

E. Maintenance of failed nodes

Sensor failures could occur due to different reasons such as battery depletion, or hardware/software manufacturing, or calibration drift. We extend the sensor node architecture by integrating to the WISP a dual Access EEPROM, which can be accessed through a wired serial port from the embedded micro-controller, or through a wireless RFID reader. The use of the RFID interface will allow the memory of the sensor to be read and updated remotely and powerlessly. Therefore, the failures can be investigated even if the sensor is unable to respond to the quadcopter's requests. The used dual-port access memory stores the configuration state of the deployed sensors, including: a) A routing table showing at least a route to the nearest DRN. A route is a four-uplet in the form of $\langle DRN\, id,\ next\, hop,\ distance,\ timestamp\rangle$, describing the id of the DRN, the identity of the next hop, the number of hops to reach the DRN, and the time of the last update (performed periodically or upon a failure detection); b) A timestamped value of the remaining battery energy computed and updated by the sensor each period of time T; c) The value of energy average consumption computed over a predefined number of hours; d) The content of alerts that remained in the sending buffer for a period of time exceeding a threshold Th. Failures in sending the buffered data could occur due to the unavailability of neighbor sensors in the route towards the BSN; and e) A status flag describing whether the sensor is in active or sleeping state. To detect and investigate failures, the quadcopter uses its RFID reader to read the content of the dual access memory. Failures can be detected as follows:

a) Detection of unreachable nodes: Some nodes could be unreachable even if they are in the transmission coverage of their neighbors, especially due to transmission impairments. This failure can be detected by noticing the availability of alerts that remained in the sensor node's sending buffer for a period of time exceeding a threshold T_{max}. Basically, the quad-copter traverses a linear path on the monitored area and checks the content of the dual access memory of the sensors therein deployed.

We consider a set $A = \{a_i | i = 1, ..., n\}$ of alerts available in a sensor buffer at a given instant T_{read}, which denotes the time when the quad-copter starts reading the content stored in the sending buffer. The waiting time of a given alert is denoted by $\Theta(a) = |T(a) - T_{read}|$, where $T(a)$ denotes the generation time of alert a. If there exist an alert $a \in A$ such that $\Theta(a)$ is greater than a predifined threshold T_{max}, $(\Theta(a) \geq T_{max})$, a transmission failure is detected. The quadcopter proceeds by copying the content of these buffered alerts and immediately forwarding them to the NCC. After being acknowledged, it deletes these alerts from the dual access memory of the sensor node.

b) Detection of out-of-coverage BSN nodes by the quad-copter: The use of the quad-copter aims to increase the coverage of the entire border surveillance system by investigating and repairing sensor nodes connectivity problems such as the existence of out-of-coverage nodes. We assume that every alert reception should, normally, be acknowledged by the NCC by τ seconds; otherwise the unacknowledged alert must be retransmitted in a later time until it is finally received. The number of retransmissions is limited to a threshold value Ret_{max}. If such a threshold is reached, the buffered alert should be tagged in order to be read and relayed using the quad-copter (i.e., via the WISP).

More formally, at a given instant t, the quadcopter checks if the routing table is empty or contains an outdated route to the DRN. It checks, within the buffered alerts on every node, if the current time value t_{read} is higher than the sum of the route timestamp and the period of update, t_{up}, $(t_{read} > timestamp(j) + t_{up})$ and whether the maximum number of retransmissions is exceeded.

If it is the case, the NCC instructs the quadcopter to correct coverage holes by relaying these alerts and deleting them from the buffer after being acknowledged. In addition, the quadcopter is in charge of dropping additional sensors along the uncovered zone as discussed in the previous section.

c) Powerless detection/identification of critical battery level and out-of-energy nodes : To determine whether the sensor is still active, the quadcopter checks whether the timestamp of the last update is recent, and whether the expected remaining energy has reached the zero value considering the average consumption of energy. If sensor's battery is depleted, the NCC instructs the quadcopter to drop a new sensor node in that location.

We consider that the quad-copter is also able to detect whether the sensor's battery status is critical before completely depleting its energy. The quad-copter reads the last update of the remaining battery energy level E_{level} and checks if it is lower than a threshold E_{th} based on the following inequality:

$$E_{level} - (E_{mean} * T_Q) < E_{th} \tag{4}$$

where E_{mean} is the mean consumption energy level of a given sensor per a unit of time and E_{th} is a threshold value that defines the minimum level of energy required to keep a sensor node active for the period T_Q that denotes the quad-copter mean intervention time.

Consequently, if the residual energy of a deployed sensor goes below this threshold, an alert is automatically forwarded to the NCC, which instructs the quad-copter to intervene in that location by replacing the sensor node before its battery becomes out of energy .Tracking assistance

This subsection focuses on the description of an UAV-assisted tracking scheme that exploits the cooperation between the UAVs and the deployed WSN in order to provide an accurate target state estimation and an efficient tracking of trespassers.

F. Intruders Exit Point Estimation

After detecting a trespasser, alerts generated by BSNs are forwarded to the NCC through the hierarchical DRNs and DDNs. The NCC predicts the trajectory of the trespassers, instructs the DRNs on that trajectory to wake up sensors in their vicinity, and sends the quadcopter to the trespassers' exit point to remotely capture real-time video of the intrusion scene. To predict the intruder's exit point, we consider the following hypotheses:

- The intruders seek to take the shortest path, when crossing the border area, and try to not be exposed for a long time to the sensor nodes deployed along the border, so as to minimize the likelihood of being detected .
- An intruder, as illustrated in Figure 5, is moving along a trajectory whose direction is typically perpendicular to the borderline, but due to the geographical characteristics

of the border areas, it can be inclined vertically with a tilt angle α.

- The quadcopter is always flying at the upper boundary. Therefore, to maximize the number of detected trespassers while minimizing the energy consumption we consider that the capture of trespassers is performed at the exit point of border strip.

The estimation of the intruder's exit point is based on the estimation of the next sensor by which it will pass and the projection of the position of the last sensor that has detected the intrusion on the x-axis.

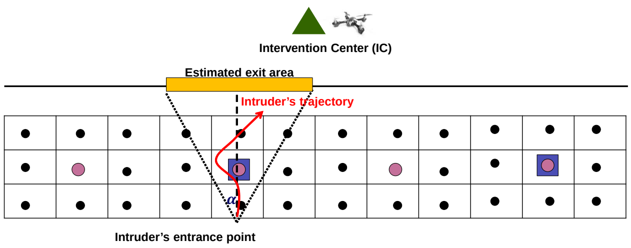


Figure 5: Typical Example of Intruder's Trajectory

Let us consider a set $N_{int} = 1, ..., n$ of intruders which are crossing the border line at an instant t, and let (x_i, y_i) be the position of the intruder i. This position is varying over time. We consider that from the different positions computed over time, we can estimate the X's coordinate, say $\hat{x}_i$, of the trespasser at the upper boundary axis, at the moment when it will exit the borderline. Such an estimation can be done using prediction algorithms [26], [27], and based on the trespasser's velocity and direction as determined by the deployed sensors. The control center keeps track the estimated exit point of each trespasser and updates it as long as new sensors detect its displacement. These estimated exist points are shared with the different intervention centers which will send their quadcopters to spot trespassers when they will cross the upper boundary axis, allowing to reduce the rate of false positives by determining the type of moving object.

G. Quadcopter-based tracking

The quadcopter is initially located at the intervention center (IC). To track intruders, it reads the list of estimated exist points of trespassers and goes back and forth along the borderline to spot tress passers. However, the quadcopter is constrained to return to the IC for the purpose of maintenance (replacement of the depleted battery by a fully charged one). Therefore, efficiently scheduling the list of tasks that should be performed by the quadcopter is necessary to minimize energy consumption, reduce unnecessary displacement, and avoid battery depletion before reaching the intervention center. While the global direction of the trespassers' trajectory is almost perpendicular to the borderline, their instantaneous velocity and direction is random, as illustrated in Figure 5). Therefore, the estimation of their exit instant and x's coordinate is uncertain. Consequently, the scheduling of quadcopter's' actions should be based on heuristics that use the information acquired from sensor nodes and available in NCC in order to further the selection, with respect to a predefined conditions. In this sub-section, we focus on describing a priority-aware heuristic-based tracking algorithm for effectively tracking, using quadcopters, multiple trespassers in the monitored area. Such an algorithm meets the following properties:

- The NCC is in charge of scheduling the tracking tasks based to their times of exit, and sending them to the quadcopter;
- The execution of a tracking task cannot be interrupted once the quadcopter starts performing it;
- Once the quadcopter finishes the execution of a tracking task, it receives another instruction or is ordered to return to the intervention center for battery replacement.

We consider a set of tracking instructions $Q = \{q_i | i = 1, ..., n\}$, where a tracking instruction, $q_i = (\hat{x}_i, \Theta(\hat{x}_i))$, is a tuple describing the estimated X's coordinate of the trespasser's exit point at the borderline $(\hat{x}_i)$, and the time instant when it will be there $\Theta(\hat{x}_i)$. To serve the set of instructions in Q, two main heuristics are used, namely the Earliest Exit Time First (EETF) and the Nearest Exit Time First (NETF).

- ***The Earliest Exit Time First (EETF)***: this heuristic encourages the selection of the task to be served according to the time of exit value. In other words, quadcopter servers the task related to the trespasser that will be the first to cross the borderline.
- ***The Nearest Exit Time First (NETF)***: this heuristic suggests selecting the task that minimizes the energy consumed to move the quadcopter. In this context, the quadcopters serves the nearest task to its current position.

In the sequel, we denote by $Select\,(Q, E_Q, E_{Qth}, H)$ the function that will select the task $q \in Q$ to be served based on the heuristic H, considering a remaining battery energy equal to E_Q. Before serving a task and moving from a location to another, the quadcopter should guarantee that the estimated energy at the arrival is higher or equal to a predefined threshold E_{Qth}.

Algorithm 1 describes how tasks are served by the quadcopter, considering an available set of tasks Q, a battery energy level equal to E_Q, and an energy threshold E_{Qth} (i.e., the minimal acceptable energy level of the quadcopter energy to guarantee its availability). Whenever there are new tasks in the set Q received from the network control center and not already served by the quadcopter, the quadcopter is ordered by the intervention center to serve the task selected with respect to the heuristic H. If the energy remaining in the quadcopter battery is sufficient to move it to forth toward the location of the task and back toward the intervention center, the task is served. Otherwise, the quadcopter is moved directly to the intervention center for maintenance purpose. After the accomplishment of every task, the energy remaining in the quadcopter battery is inspected, its location is updated, and the task is set to served. Since the quad-copter has the capability to communicate with the deployed sensors within a communication radius ρ, the NCC can also instruct the quadcopter to check if the sensors located on the predicted trajectory and within its communication radius are all woke up by remotely reading

Algorithm 1 Priority-aware heuristic-based tracking algorithm

Track (Q, E_Q, E_{Qth}, H)
Begin
$\forall q \in Q$: $Served(q) \leftarrow false$
// All received tasks are assumed to be not already served
$x \leftarrow GetQuadLoc()$
// Set x to the quadcopter location on the borderline
While$(Q \neq null$ && $\exists q \in Q$ $such\, that\, Tracked(q){=}false)$ $q \leftarrow Select\,(Q, E_Q, x, H)$
// Select the next node to serve (the task $q \in Q$) considering the heuristic H, the remaining energy E_Q, and the quadcopter location x.
$E_r = ReqEgy(q, x, Loc(q))$
// determine the required energy to serve the task by moving the quadcopter from the current location x to the task's location $Loc(q)$
$E_c = ReqCEgy(Loc(q))$
//determine the required energy to reach the control center once the task q is served at location $Loc(q)$
If $E_Q - E_r - E_c \leq E_{Qth}$
//the quadcopter has enough energy to serve the task and be able to reach the control center once the task is served
Then $Serve(q)$; $E_Q \leftarrow E_Q - E_r$; $Served(q) \leftarrow True$;
//Move the quadcopter to serve the task, update the remaining energy, and set the task as solved
Else $GoToConCenter(\,)$; $E_Q \leftarrow E_Q - E_c$;
//Move the quadcopter to the control center to change its battery, and update the remaining battery energy after such a maintenance.
End If
$x \leftarrow GetQuadLoc()$
// Update the the new position of the quadcopter on the borderline
END While
End

the dual access memory and checking the value of the status flag. If some sensors are found to be still in sleeping state, the quadcopter informs the NCC (this failure can be investigated later), and sends a pulse to the sensor (by writing directly to the dual access memory) to change its configuration state.

H. Coping with the execution of complex tasks on WISPs

Radio Frequency energy harvesting techniques are used to powerlessly locate sensor nodes located in the ground and collect and modify configuration data, or even weak up sleeping sensors. Using long-range RFID readers, quadcopters can interact with the WSN by reading the WISP tags deployed on each sensor node. Therefore, the quadcopter device should be in direct visibility with the target sensor node to efficiently perform it tasks, namely trespassers' tracking and network maintenance. However, since the energy required by these tasks may overtake the quantity of energy harvested by the WISP, sudden discontinuation and interruption of the executed operation may happen. To cope with such an issue, we propose to use one of the following two mechanisms, which were proposed in the literature to allow the execution of complex algorithms on a WISP, while introducing different steps of energy harvesting during that execution.

Several mechanisms were proposed in the literature to handle such a challenging issue. In [28], a mechanism enabling the execution of complex computational algorithms was proposed. The algorithm is split into a virtual set of instructions (the energy required by each instruction is estimated) that will be executed sequentially and will be divided into blocks. The division of the virtual set of instructions into blocks should take into consideration that the energy consumption of every block does not exceed a predefined (estimated) threshold. The WISP proceeds by the execution of the first block of instructions upon receiving from the RFID reader the required quantity of energy. Upon the block execution is terminated, the program computational state is saved to a flash memory, and a new energy harvesting request for the execution of the next bloc is sent to the programmer. The WISP waits for harvesting the required energy from the RFID reader and then resumes the program's execution by performing the next block of instructions. The execution of such a process will be repeated until the execution of the last bloc of instructions of the whole complex task. One drawback of such an approach consists of the delay overhead caused the wasted waiting time after the execution of every bloc of instructions. During that time the WISP should wait for the response of the RFID reader transported by the quadcopter, in order to harvest energy.

Mementos [29] is another proposed approach aiming that deals with the energy harvesting problem. The solution consists of a software system which is implemented under an enhanced WISP, and which integrates an energy-aware state checkpointing system that allows spliting the complex tasks into several life cycles. Trigger points are used at: (i) the compilation time to call energy estimation functions to estimate the available energy level; and (ii) run time to predict power losses. The program computational state is stored in a non-volatile memory when an energy interruption is predicted. Once enough energy is received, the previous program execution is not lost but the last saved state will define the new start execution point. Compared to the previous solution, the interruption of tasks' execution in this approach is only performed if a potential power loss is predicted by the energy estimation function, which could reduce the unnecessary overhead delays. However, energy consumption problems and additional delays may be experienced due to the repetitive and useless execution of energy estimation functions.

V. Prototyping the quad-copter

This section focuses on describing the developed prototype of the quad-copter (i.e., the quadcopter framework, and the electrical, mechanical, and software architecture), and on how this prototype has been tested to detect network status and intruders.

A. Quadcopter framework

We chose to work with a quadcopter, which has a "Y" shaped VTail design, over the conventional "X" orientation. The VTail design, shown in Figure 6 is modeled after the shape of the letter "Y" with a tail in the shape of the letter "V". The base setup of the VTail quadcopter contains 1240kV motors, 30Amp electric speed controllers (ESCs), two 8045 and two 9047 propellers, and a 2.4GHz 8 Channel radio receiver. The major differences between the VTail and conventional quadcopters are the weight, motors, and battery. The VTail's credentials allow it to carry a heavier payload, have longer flight times, and achieve more agile flight maneuvers. This unique construction promotes a more stabilized flight, combining the natural agility of a tricopter (a multirotor with three motors and three propellers) setup, the stability of the "X" style quadcopter (a multirotor with four motors and four propellers), and removes the disadvantage pending on servo control to turn in place.

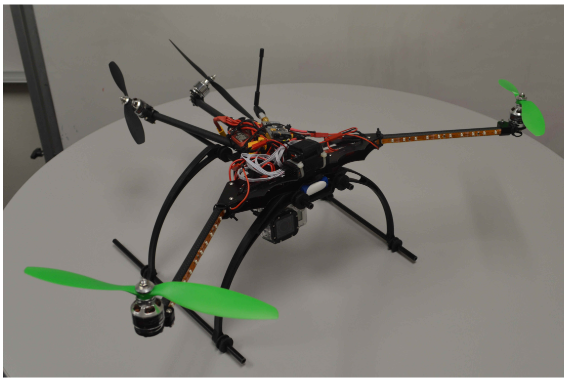


Figure 6: VTail quadcopter: Bird view

B. Quadcopter's electrical architecture

The electrical architecture shown in Figure 7, is centred around the use of a well-known flight controller called the KK2 Board made by RC store HobbyKing. This board uses the Atmega324 PA, an 8-bit microcontroller operating at 20MHz with 32 general purpose input/output pins, I2C communication protocol, Universal Asynchronous Receiver/Transmitter (UART) serial communication line, and analog to digital conversion (ADC) channels. The KK2 Board has a library of pre-installed software to compute and set the different orientations of quadcopters, which is especially useful since the VTail form is rarely supported. This board is responsible for sending pulse width modulated (PWM) signals ranging from 1.5ms to 2.0ms every 20ms to four electric speed controllers (ESCs),

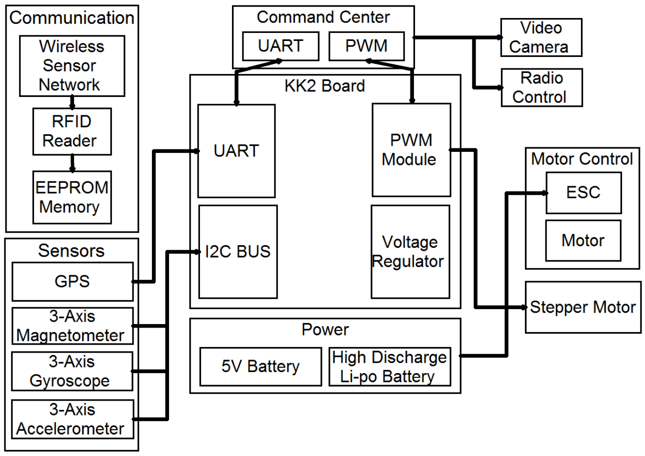


Figure 7: Electrical architecture

which control the speed and therefore, thrust of each individual motor on the quadcopter. The sensors on the board include a sensitive gyroscope and accelerometer system to keep up with the VTail's unique agility and auto-levels the quadcopter in the air at a high refresh rate. A separate GPS system is used to keep track of the UAV in relation to the Earth at all times. The sensors on the ground have radio frequency identification (RFID) tags that relay valuable information that needs to be retrieved by the UAV. To accomplish this, the ID-12LA RFID reader is placed on the quadcopter to store the information on the tag from the wireless sensor network. The quadcopter operates through a radio control frequency of 2.4GHz with the aid of a live video stream captured by a GoPro video camera and transmitted by an 800mW 1.3GHz transmitter.

C. UAV Mechanical Architecture

In the case where a sensor is damaged or is malfunctioning in the field, the UAV will need to be able to retrieve the data from the broken sensor and replace it with a working one. A workable solution is to attach a magnetic locking mechanism to the base of the quadcopter. In Figure 8 there is a permanent magnetic ring on the bottom of the structure. That ring will hold the top of the sensor to be deployed on the ground. The way that the UAV drops the sensor is by a stepper motor-driven threaded turning rod that passes through a hole in the base of the structure, making the platform able to move upwards. The rod will physically push the top of the sensor down to create enough to separate the sensor from the magnetic ring, dropping the sensor on the ground in its designated location. The stepper motor is controlled by a channel on the KK2 board as seen in Figure 7.

D. Quadcopter software architecture

The software designed for the quadcopter is an infinite loop. The process starts with the control input from either the user or a set of instructions saved in memory. The next step is to decide the best route on how to accomplish the given

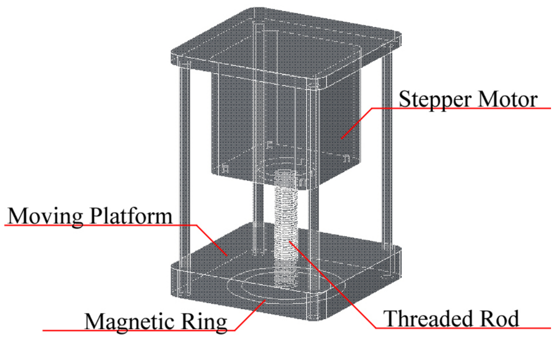


Figure 8: Magnetic Sensor Attachment

input by capturing the current sensor data provided by the accelerometer, gyroscope, magnetometer, and GPS. Then, the software that is run on the KK2 Board uses a PID controller to handle the predicative calculations, the motor speed output, and the error percentage based on the difference between the expected and actual attitude of the quadcopter. The ESCs each have their own processor that operates at a relatively high clock speed in order to react in real-time to the KK2 Board's rapid string of commands. That command is used as an objective to reach and in turn, is translated as a task that is attempted until it is completed. Then, the process starts over with a new input resulting into a different task. Possible tasks include, but are not limited to, travelling to a set of GPS coordinates, deploying/retrieving a sensor, and surveillance and data collection using near field communication.

E. Testing the quadcopter's Prototype

We tested our prototype on a test scenario with two control method setups, namely manual control and GPS based auto-control, for providing valid parameters to simulations. In the test scenario, three triangular positions (Intruder position A, and Intruder position B, and intervention center C) were arranged with GPS coordinates. Our quad-copter flights over each intruder A and B, and then flights back to C. We equipped the designed quadcopter with a 3000mAH battery. In comparison with a normal quadcopter, which has a 3300mAH battery, our quadcopter can reach a maximum flight distance of 3390 m before it has to return to its take-off area, while the Crossfire's maximum distance is 1872 m. This difference is mostly due to the fact that the VTail is lighter, more agile, and faster than the Crossfire, overcoming the battery disadvantage. The prototype made 10 flights over intruder positions, and we measured the battery consumption, and also the flight speed, distance, and mission completion times based on GPS positions. Our quad-copter showed a top speed of 11.5m/s.

To spot a trespasser, the range of the VTail quadcopter can be extended by considering the range of the camera used to take a picture. The camera used for the simulation was a GoPro Hero 3 Black Edition, having a resolution of 12 Mega Pixels. The range of this camera considering the need to recognize a face, is given by :

$$\begin{cases} H = r_w \times (w_m/w_p) \\ D = f \times (H/h) \end{cases} \tag{5}$$

where H is the width of the scene (m), r_w is the width of the resolution of the scene (px), w_m is the width of the face to detect in meters (m), w_p is the width of the face to detect in pixels (px), D is the distance to the scene (m), f is the maximum focal length of the camera (mm), and h is the width of the CMOS (mm). Obviously, experimentally finding the Optimal flight speed of the quadcopter can increase flight efficiency so that battery lifetime and flight distance can also be extended due to the phenomenon called "helicopter transnational lift". If the altitude of the UAV were to be 10 meters, the distance that can be added to the radius of the VTail is 99.08 meters measured on the ground and derived from Pythagorean theorem $sqrt((99.586m)^2 - (10m)^2))$.

In this experimentation, we were unable to find the optimal speed of our quad copter yet because of the limited coverage of the flight speed control system currently used. Such a feature will be be developed in a future work.

VI. Simulation

Each BSN or DRN is able to estimate the remaining lifetime by calculating the average energy consumption in J/S over a history period. Before its lifetime reaches a threshold value Th, a node forwards a notification to the NCC which intervenes by sending a quadcopter to replace that node and consequently extend the network lifetime. We suppose that the time of intervention of the quadcopter is constant and defined as the total time required to: a) fly to the suitable zone; b) compute the position of the new BSN to drop; c) drop the new BSN; d) wait for the new BSN to attach itself to the network; and e) read the BSN's WISP to check whether the routing table of the BSN has a new route to the DRN.

A. Simulation Model

We consider a thick line WSN deployed along a rectangular area of 6000 meters length and 150 meters width. A DRN is placed each 150 meters, and each DRN is encircled by 8 BSNs. The distance between two BSNs is set to 30 m. The IC is located in the middle of the intervention zone and on border line. We assume that: (a) during the operation period of the network, a sensor can be replaced several times thanks to the use of the quadcopter, which is able to repair one or several failures simultaneously; (b) the energy consumption of a sensor depends on the number of alerts generated and forwarded and on the duration of sensing period; (c) the monitored area can be divided into adjacent and non-overlapping intervention areas (each area is under the control of one quadcopter); and (d) a set of trespassers are crossing the border line starting from a point of entrance p, by following a linear trajectory and using a constant velocity $v = 1m/sec$. The line connecting the entrance point to the exit point of the intruder is assumed to be perpendicular to the borderline. Due to the geographical characteristics of the border areas, the intruder's trajectory is typically vertical to the borderline (i.e., the intruder wants to quickly cross the border and remain the minimum time exposed to the WSN nodes), but it can be inclined vertically at an angle α. For each intruder, the point of entrance p and the vertical inclination α are chosen randomly (following an uniform distribution), where $p \in [O, 6000]$ and $\alpha \in [-\frac{\Pi}{6}, +\frac{\Pi}{6}]$. As long as a moving trespasser is in the sensing range of a BSN, it generates an alert with a constant rate of $1\, packet/sec$. Such an activity will lead to an energy consumption considering the following parameters: Transmission $(59.2\mu J/Byte)$, reception $(28.6\mu J/Byte)$, sensing $(6 \times 10^{-3}\, \mu J/msec)$. The used battery has an initial power equal to $8640\, J$, and the alert datagram size is equal to 36 bytes. The time of intervention of the quadcopter is the total time required to: a) fly to the suitable zone; b) compute the position of the new BSN to drop; c) drop the new BSN and wait for its attachment to the network; and e) read the WISP of the BSN to check whether it has a new route to the DRN. We developed our own simulator using Matlab tool.

B. *Estimation of the BSNs' lifetime span*

The first simulation we conducted aims to evaluate the average rate of BSNs' lifetime span with respect of the threshold Th (The lifetime notification threshold). Figure 9 illustrates the variation of the average rate of BSNs' lifetime span in terms of the time of intervention of the quad-copter, considering a variable number of quad-copters. Let T_i be a period of operational time between the $(i-1)^{th}$ and i^{th} failure (In particular T_0 denotes the operation time before the first failure). Then, the average rate of BSNs' lifetime span, denoted by L, is given by:

$$L = (\sum\nolimits_{i=1...n} T_i)/(Simulation\, time - T_0) \tag{6}$$

Based on the obtained results, we notice that the average rate of BSNs' lifetime span rises with the increase of the number of quad-copters. Subsequently, the more the number of quad-copters we have, the better the network lifetime gain we obtain. In addition, we notice that for a given number of quad-copters the curves decrease with the growth of the intervention time. In fact, when the quad-copter is able to reach the failed nodes rapidly (i.e. intervention time < 15 minutes) the gain is considerably important regardless of the number of quad-copters. As long as the intervention time is getting higher than 15 minutes, the gap between the average rates increases with the increase of the number of quad-copters.

Figure 10 shows the evolution of the average rate of BSNs' lifetime span with respect to the number of intruders, considering different values of the quad-copter intervention time. The rate of gained network lifetime decreases with the increase of the number of intruders crossing the monitored borderline. The higher is the frequency of intrusions, the more the quad-copter fails to reach the BSN nodes before becoming out of energy. We also notice that the negative impact of the number of intrusions/hour on the rate of gained lifetime, becomes more and more important with the increase of the intervention time. In particular, when the quad-copter is able to reach the failed nodes rapidly (i.e.,intervention time $\leq$ 10 minutes), the gain remains considerably important regardless of the number of intruders/hour. When the quad-copter takes more than 10 min to reach the BSN nodes, the gap between the obtained rates for the same number of intruders is important. This gap increases significantly with the increase of the number of intruders.

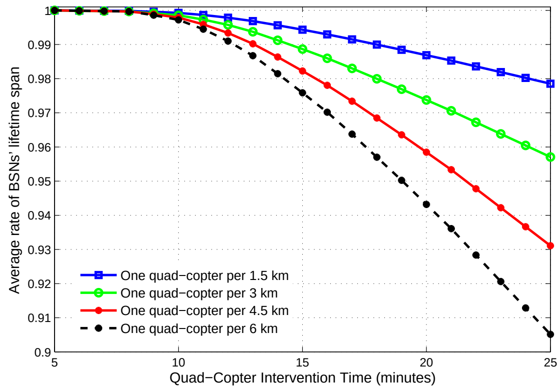


Figure 9: Average rate of BSNs lifetime span vs. intervention time

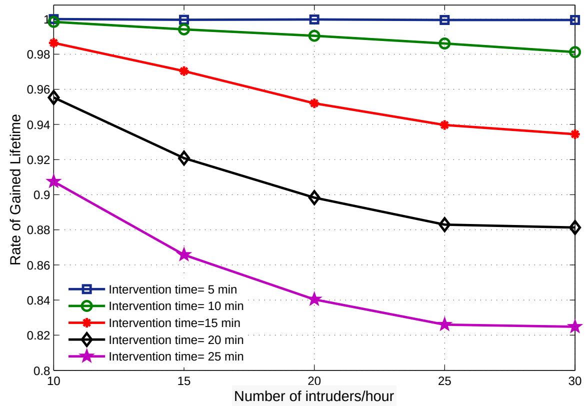


Figure 10: Impact of the number of intruders on the average rate of BSNs lifetime span

Figure 11 shows the evolution of the average rate of BSNs' lifetime span with respect to the mean intervention time of the quad-copter, considering different values of the mean residential time (TR) of intruders under the coverage of one BSN. Based on the obtained results, we notice that the rate of gained lifetime is negatively affected by the increase of the mean intervention time of the quad-copter as well as the increase of the intruders' mean residential time. The lower is the velocity of intrusions, the more the BSNs generate alerts and the more the quadcopter fails to reach the BSN nodes before becoming out of energy. This impact becomes more important when the mean residential time per sub-square is equal to 60 seconds. Indeed, we notice that the curves for TR=60 seconds decrease dramatically when the mean intervention time of the quadcopter is greater than 5 minutes. When the value of TR is in the range [30 50] and the quad-copter is able to reach the failed nodes rapidly (i.e., intervention time <10 minutes), the gain remains considerably important. However, when the quad-copter takes more than 10 min to reach the BSN nodes, the curves are considerably decreasing and the gap between the obtained rates is significantly increasing. The more the mean intervention time is increased the more the average rate of

Figure 11: Impact of the intruders' velocity and Quadcopter's intervention time on the average rate of BSNs lifetime span

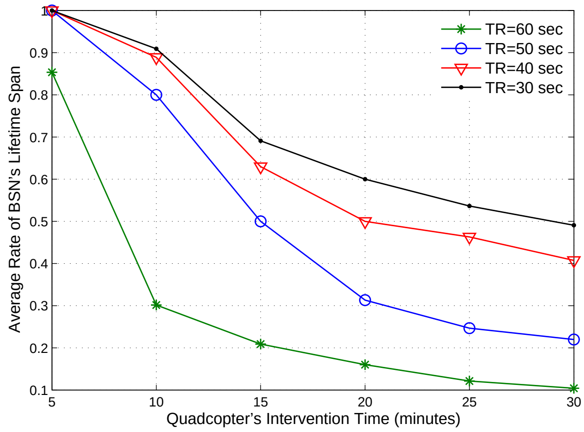
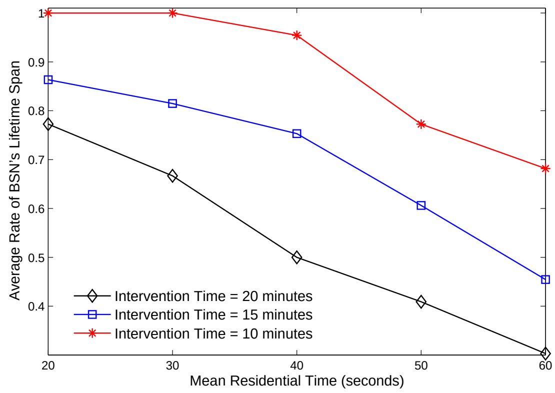


Figure 12: Impact of the Residential Time of Intruders on The average rate of BSN's Lifetime Span



gained lifetime is decreased. For a value of TR equal to 60 sec, the quad-copter may fail to reach the BSNs before completely depleting their energy if it takes more than 5 minutes to repair them. To optimize the BSNs' lifetime, the designer should be able to either reduce the mean intervention time of the quad-copter or to multiply the quadcopter's intervention centers.

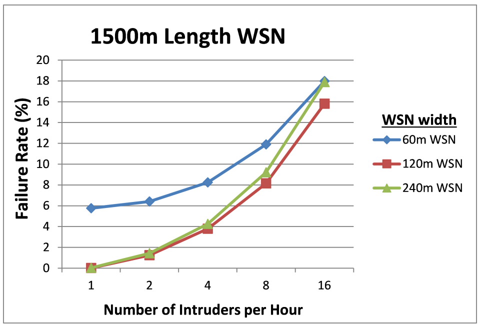
In this simulation, the number of deployed BSNs is increased, and the the border strip is enlarged, so that a BSN can be connected to a DRN in a two-hop path The BSN node keeps the generation of alerts as long as the intruder lasts in its correspondant sub-square. The concept of multi-hop communication has been introduced during simulation to allow BSN nodes that detect trespassers to transmit their alerts through the shortest path to the DRN node. Figure 12 shows the impact of the average residential time of trespassers per sub-square on the evolution of the average rate of BSNs' lifetime span. The illustrated curves are obtained for different values of the quad-copter intervention time (10, 20, and 30 min). We notice that the rate of gained network lifetime decreases with the increase of the average residential time and with the decrease of the quadcopter intervention time. Compared to the previous simulation, whichconsiders that BSNs transmit their generated alerts directly to their DRN neighbor, in this simulation we consider that alerts are forwarded through a one-hop or two-hop route to their destination (i.e., a DRN node in the same sub-square). The gained lifetime is optimized since unnecessary transmissions are avoided. By considering that alerts are transmitted to their destinations (i.e., the DRN node) through the shortest path, only the BSNs on that route will participate in the transmission process. The impact becomes more important when the mean residential time per sub-square is greater than 30 seconds. When the quad-copter takes 10 minutes to reach the BSN nodes, the average rate of BSN's lifetime span starts to considerably decrease once the mean residential time becomes higher than 30. By minimizing the quadcopter's response time and the trespasser's residence time the gained BSN's life time will be considerably improved. The configuration of the quad-copter's mean intervention time should be adjusted by taking into consideration a set of parameters reported by the terrestrially deployed WSNs including the number of intruders, the mean residentail time, and the rate of alerts generation. Therefore, by increasing the quad-copter's intervention time we decrease the probability to successfully replace (i.e., react just in time) the depleted BSNs.

C. *Estimation of the rate of non spotted trespassers*

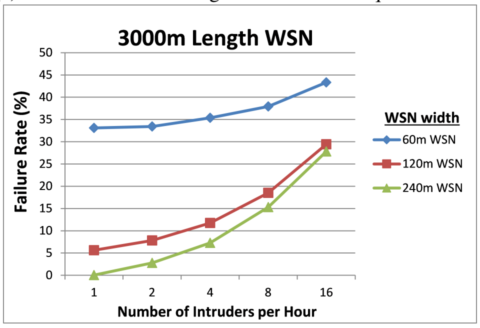
We simulated the percentage of failures in spotting intruders with respect the number of trespassers per hour. The simulation time spans 100 hours. The results are described in Figure 13 considering a simulation area at 1500 meters, 3000 meters, and 6000 meters. For each simulation, we varied the thickness of the WSN considering a border width of 120 meters, 240 meters, and 480 meters. We conducted the simulation with one UAV having a top speed of 11.5m/s. The result of all three plots show that the failure rate of UAV detection increases with the increase of intruders per hour. In fact one UAV will not be able to keep up with the demand of detecting intruders across the border if the intrusions occur frequently, even with the battery being replaced when the UAV returns to its home base. The three plots also show that the rate of failure decreases faster with the increase of border thickness, which increases the total time that the intruder takes to cross the border under the WSN coverage, and reduces the time constraints for the UAV to move and spot the intruder successfully. The longer is the distance to the intruder, the longer the UAV is busy in detecting the intruder, increasing the number of intruders that cross the border undetected. The best results overall show that a WSN thickness of 240 m yields the lowest UAV detection failure rate for a WSN length of 1500 meters.

VII. Conclusion

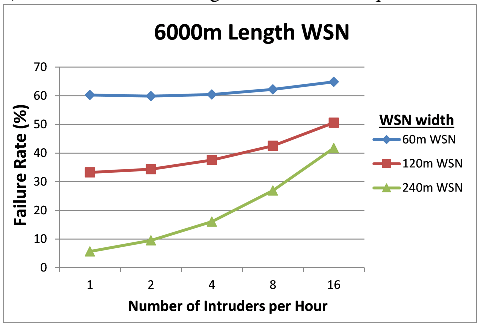
We developed in this work a border surveillance application using quadcopters as a tool for the proactive and reactive response to failures and intrusions, to improve the quality of detection and tracking of trespassers crossing a border supervised by a wireless sensor network. A VTail quadcopter is designed, and a prototype is developed and tested. The



(a) Simulation with a length of borderline equal to 1500m



(b) Simulation with a length of borderline equal to 3000m



(c) Simulation with a length of borderline equal to 6000m

Figure 13: Failure rate w.r.t. number of intruders per hour

designed quadcopter detects coverage holes, identifies and investigates failures, drops new sensors at the suitable positions, relays urgent data, captures real-time video of the scene, and wakes up sensors located on the trajectory of the intruder. A priority-aware heuristic-based tracking algorithm is described to allow the quadcopters in a given intervention area to effectively track and spot intruders. Intrusions detected in the same intervention area are managed by the same quadcopter. However, due to the limited energy of the quadcopter and the detection of tracking tasks with close exit times, some intruders may succeed to cross all the borderline without being intercepted by the quadcopter. To alleviate such a problem we aim to enhance the proposed algorithm by considering the use of overlapping intervention areas, where a trespasser can be tracked at least by one quadcopter.

REFERENCES

[1] R. L. Skinner, "A review of remote surveillance technology along u.s. land borders," Office of Inspections and Special Reviews, Department of Homeland Security Office of Inspector General, Tech. Rep. OIG - 06 -15, December 2005.

[2] R. Bellazreg, N. Boudriga, and S. An, "Border surveillance using sensor based thick-lines," in *Proceedings of International Conference on Information Networking (ICOIN)*, Bangkock, Thailand, January 2013.

[3] Z. Suna, P. Wang, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, "Bordersense: Border patrol through advanced wireless sensor networks," *Ad Hoc Networks*, vol. 9, no. 3, pp. 468 – 477, May 2011.

[4] E. Onur, C. Ersoy, H. Deliç, and L. Akarun, "Surveillance wireless sensor networks: Deployment quality analysis," *IEEE Network*, vol. 21, no. 6, pp. 48 – 53, 2007.

[5] D. Dudek, C. Haas, A. Kuntz, M. Zitterbart, D. Kruger, P. Rothenpieler, D. Pfisterer, and S. Fischer, "A wireless sensor network for border surveillance," in *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Berkeley, CA, USA, November 2009.

[6] A. R. Girard, A. Howell, and J. Hedrick, "Border patrol and surveillance missions using multiple unmanned air vehicles," in *Proceedings of the 43rd IEEE Conference on Decision and Control (CDC)*, Paradise Island, Bahamas, December 2004.

[7] A. Matlock, R. Holsapple, C. Schumacher, J. Hansen, and A. Girard, "Cooperative defensive surveillance using unmanned aerial vehicles," in *Proceedings of American Control Conference*, St. Louis, MO, USA, June 2009.

[8] J. S. Gadda and R. D. Patil, "Quadcopter (uavs) for border security with gui system," *International Journal of Research in Engineering and Technology (IJRET)*, vol. 2, no. 12, pp. 620 – 624, 2013.

[9] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 92, pp. 79–97, June 2014.

[10] L. Lamont, M. Toulgoat, M. D. eziel, and G. Patterson, "Tiered wireless sensor network architecture for military surveillance applications," in *Proceedings of the 5th International Conference on Sensor Technologies and Applications (SENSORCOMM)*, Côte d'Azur, France, August 2011.

[11] J. Hare, S. Gupta, and J. Wilson, "Decentralized smart sensor scheduling for multiple target tracking for border surveillance," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, Washington, May 2015.

[12] R. Bellazreg, N. Boudriga, K. Trimeche, and S. An, "Border surveillance: A dynamic deployment scheme for wsn-based solutions," in *Proceedings of Wireless and Mobile Networking Conference (WMNC)*, Dubai, United Arab Emirates, April 2013, pp. 1 – 8.

[13] D. Bein, W. Bein, A. Karki, and B. B. Madan, "Optimizing border patrol operations using unmanned aerial vehicles," in *proceedings of the 12th International Conference on Information Technology - New Generations*, Las Vegas, NV, USA, April 2015, pp. 479 – 484.

[14] J. LeVous, J. Hoven, V. Habgood, A. Alotaibi, B. Ordway, G. Mohammed-Jones, and H. Guo, "Object retrieval from secure unknown interior spaces using autonomous unmanned aerial vehicles," in *Proceedings of Fourth Symposium on Indoor Flight Issues, International Aerial Robotics Competition*, 2012.

[15] N. R. Kuntz and P. Y. Oh, "Towards autonomous cargo deployment and retrieval by a unmanned helicopter using visual tracking," in *Proceedings of the ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Brooklyn, New York, USA, Augus 2008.

[16] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and control for aerial grasping and manipulation," in *Proceedings of International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, August 2011.

[17] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams," in *Proceedings of Robotics: Science and Systems*, Los Angeles, California, USA, June 2011.

[18] M. Orsag, C. Korpela, and P. Oh, "Modeling and control of mm-uav: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent and Robotic Systems*, vol. 69, no. 1-4, pp. 227–240, 2013.

[19] M. Bernard, K. Kondak, I. Maza, and A. Ollero, "Autonomous transportation and deployment with aerial robots for search and rescue missions," *Journal of Field Robotics*, vol. 28, no. 6, pp. 914 – 931, June 2011.

[20] J. L. Fargeas, P. Kabamba, and A. Girard, "Cooperative surveillance and pursuit using unmanned aerial vehicles and unattended ground sensors," *Sensors*, vol. 15, pp. 1365–1388, 2015.

[21] S. Berrahal, J.-H. Kim, S. Rekhis, N. Boudriga, D. Wilkins, and J. Acevedo, "Unmanned aircraft vehicle assisted wsn-based border surveillance," in *proceedings of the 23rd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, Bol, September 2015.

[22] A. P. Sample, D. J. Yeager, P. S. Powledge, and J. R. Smith, "Design of a passively-powered, programmable sensing platform for uhf rfid systems," in *Proceedings of IEEE International Conference on RFID*, Grapevine, TX, USA, March 2007.

[23] M. Philipose, J. Smith, B. Jiang, A. Mamishev, S. Roy, and K. Sundara-Rajan, "Battery-free wireless identification and sensing," *Pervasive Computer*, vol. 4, no. 1, pp. 37 – 45, 2005.

[24] D. Yeager, P. Powledge, R. Prasad, D. Wetherall, and J. Smith, "Wirelessly-charged uhf tags for sensor data collection," in *Proceedings of IEEE International Conference on RFID*, Las Vegas, NV, USA, April 2008.

[25] Z. Yi and J. R. Smith, "A battery-free rfid-based indoor acoustic localization platform," in *Proceedings of IEEE International Conference on RFID*, Orlando, FL, USA, April - May 2013.

[26] J. Lin, W. Xiao, F. L. Lewis, and L. Xie, "Energy-efficient distributed adaptive multisensor scheduling for target tracking in wireless sensor networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 6, pp. 1886 – 1896, June 2009.

[27] T. He, P. Vicaire, T. Yan, L. Luo, L. Gu, G. Zhou, R. Stoleru, Q. Cao, J. A. Stankovic, and T. Abdelzaher, "Achieving real-time target tracking usingwireless sensor networks," in *Proceedings of the 12th IEEE Real-Time and Embedded Technology and Applications Symposium*, Washington, DC, USA, April 2006, pp. 37 – 48.

[28] N. Ellouze, M. Allouche, H. B. Ahmed, S. Rekhis, and N. Boudriga, "Securing implantable cardiac medical devices: use of radio frequency energy harvesting," in *Proceedings of the 3rd international workshop on Trustworthy embedded devices*, New York, USA, 2013, pp. 35 – 42.

[29] B. Ransford, J. Sorber, and K. Fu, "Mementos: system support for long-running computation on rfidscale devices," *ACM SIGPLAN Notices*, vol. 64, no. 3, pp. 159 – 170, 2011.

Sarra Berrahal is a PhD Student at the Engineering School of Communications (SUP'Com, Tunisia), University of Carthage, and member of the Communication Networks and Security (CNAS) research Laboratory at the same University. She is conducting his research activities in the area of wireless body area network, networking, and quality of service.

Jong-Hoon Kim is currently a chief technology advisor at ArtXpresso L.L.C. and a honorary member of the FIU Discovery Lab. He was a director of FIU Discovery Lab and a visiting assistant professor in school of computing and information science, Florida International University from Jan 2012 to Aug 2014. He received a M.S. and Ph.D. in department of computer science, Louisiana State University at Baton Rouge in 2008 and in 2011 respectively. He received a B.S. from Seoul National University of Science and Technology, Seoul, South Korea in 2005. His research interests are in Tele-robotics, Mobile Robot, HRI, Embedded System, HCI, Sensor Network, and Intelligent System. Currently, his research is focusing on Tele-robotic project, Tele-presence humanoid robot for law enforcement, and Smart UAV project, intelligent UAV for border surveillance.

Dr. Slim Rekhis received the University Habilitation and the Ph.D. degrees in Information and Communication Technologies from the Engineering School of Communications (SUP'Com, Tunisia) in 2013, and 2007, respectively. He is currently an Associate Professor in the School of Communication engineering (SUP'Com, University of Carthage, Tunisia) and member of the Communication Networks and Security (CNAS) research Laboratory at the same University. Dr. Rekhis is conducting his research activities in the area of digital investigation of cyber security incidents, formal modeling, intrusion prevention and detection, wireless networks, networking and quality of service, and wireless security.

Pr. Noureddine Boudriga received his Ph.D. in Algebraic topology from University Paris XI (France) and his Ph.D. in Computer science from University of Tunis (Tunisia). He is currently a full Professor of Telecommunications at the University of Carthage, Tunisia and the Director of the Communication Networks and Security Research Laboratory (CNAS, University of Carthage). He is the recipient of the Tunisian Presidential award in Science and Research (2004). He has served as the General Director and founder of the Tunisian National Digital Certification Agency (2000-2004). He was involved in very active research in communication networks and system security. He authored and co-authored many chapters and books on information security, security of mobiles networks, and communication networks. He published over 300 refereed journal and conference papers. Pr. Boudriga research interests include networking and internetworking security, security management of electronic services, and security-related theories and formal methods.

Deon Wilkins graduated from Bethune-Cookman University in Daytona Beach, Florida with a Bachelor of Science Degree in Computer Engineering and minors in Computer Science and Mathematics. While working towards a Master's Degree in Computer Engineering, he conducts research in the Discovery Lab with Unmanned Aerial Vehicles and with the mobile mechanism team of Telebot.

Jaime Acevedo is an undergraduate student in School of Computing and Information Science at Florida International University, Miami, Florida. He is conducting research on design of Unmanned Aerial Vehicles and its advanced control system at the FIU Discovery Lab.