INT J COMPUT COMMUN, ISSN 1841-9836
Vol.7 (2012), No. 2 (June), pp. 312-324

An Intelligent and Pervasive Surveillance System for Home Security

A. Longheu , V. Carchiolo, M. Malgeri, G. Mangioni

Alessandro Longheu, Vincenza Carchiolo,
Michele Malgeri, Giuseppe Mangioni
Dipartimento di Ingegneria Elettrica, Elettronica ed Informatica
Facoltà di Ingegneria - Università degli Studi di Catania
Viale A. Doria 6, I 95125 - Catania - Italy
{alessandro.longheu, vincenza.carchiolo, michele.malgeri,
giuseppe.mangioni}@dieei.unict.it

Abstract: Domotics is a promising area for intelligent and pervasive applications that aims at achieving a better quality of life. Harnessing modern technologies is valuable in many contexts, in particular in home surveillance scenario, where people safety or security might be threatened.
Modern home security systems endorse monitoring as well as control functions in a remote fashion, e.g. via devices as a laptops, PDAs, or cell phones, thus implementing the pervasive computing paradigma; moreover, the intelligence is now often embedded into modern applications, e.g. surveillance systems could adapt to the environment through a self-learning algorithm.
This work presents an intelligent and pervasive surveillance system for home and corporate security based on the ZigBee protocol which detects and classifies intrusions discarding false positives, also providing remote control and cameras live streaming. Results of tests in different environments show the effectiveness of the proposed system.
Keywords: Home surveillance systems; pervasive systems; ZigBee protocol.

1 Introduction

Home automation [1], [2] exploits the latest technologies to provide an intelligent control of lighting, air conditioning, plumbing systems, home appliances and security systems, achieving comfort, safety, efficiency, costs/energy savings and, in summary, a better quality of life [3], [4].

One of the first scenarios where the penetration of domotics occurred is home surveillance, where sensors, actuators, alarms, controllers or even robots [5] increase safety through continuously environment scanning, sending alarms and recording incidents upon detecting any abnormal event, based on the consolidated principles of corporate security [6].

The amazing part of modern home security systems is that monitoring and control can be performed remotely via devices as a laptops, PDAs, or cell phones. According to this criterion, home security systems can be classified into four categories [7]:

- hardware-based, the simplest systems where both monitoring and control are implemented in hardware,
- passive systems, where only the monitoring is remote (the control is manual),
- phone based systems, with monitoring and control performed through the phone (wired and/or cellular) network
- web-based systems, identical to phone-based but using the Internet as the communication infrastructure.

Hardware-based systems in general offer high performances but with higher costs than other solutions, and they might be subjected to proprietary solutions thus with potentially drawbacks as less interoperability and less flexibility; conversely, lower performances but greatest flexibility at limited costs is achieved with phone- and web-based systems, which are now more and more adopted for home surveillance.

Moreover, the use of these systems together with the cutting-edge hardware and software technologies of mobile devices allows a more effective implementation of the intelligent [8] and pervasive (or ubiquitous) computing paradigma [9], [10].

The intelligence is now often embedded into modern applications, for instance the surveillance system could adapt to the environment through self-learning [11] or it can automatically start some countermeasure as some critical event occurs [12].

The concepts of ubiquitous and pervasive computing, sometimes considered as similar [13] or even overlapped with mobile or embedded computing [14], are implemented on one hand thanks to the advanced pc-based hardware (sensors, actuators etc.) available for home security, in accordance with the ubiquitous computing paradigm where computers "vanish into the background" [16], [15], on the other hand the use of smatphones or PDAs as monitor/controller devices gives users the possibility of accessing information and services anytime from anywhere, as promoted by the pervasive computing approach [17].

Another relevant issue concerning home security systems is the underlying transmission network, indeed a consolidated standardization is still missing [18], whilst the need for a quick deployed and cost-effective wireless solution to support easy remote control and affordable data transmission is emerging rapidly [19].

Currently, several home wireless networks are available as infrared technology (IrDA), Bluetooth and ZigBee protocol. IrDA operates over short distances and is subjected to high error rate, whereas the Bluetooth technology is limited by network capacity and performances.

The most promising standard for wireless home and personal area networks is then represented by the ZigBee technology [20], [21], which comes with features as low complexity, low data error rate, low power and low-cost [22], [23].

The work presented in this paper falls into the scenario outlined above, in particular we propose an intelligent surveillance system for home security that receives intrusion attempts detected by sensors and cameras connected through a ZigBee-based network and classifies such intrusions with a customizable algorithm in order to exclude false positive cases (e.g. leaves moved by the wind); this paper is an extended version of [24].

Potential alarms can be managed via an iPhone® application that receives alarms, and allows users to use the iPhone as a system remote controller and as a monitoring console to view real-time camera images.

The paper is organized as follows: section 2 introduces the architecture of the system, while in section 3 we describe in detail the application that manages devices, processes alarms detection and implements the push notification service. Section 4 shows the calibration that helps false positive intrusion detection as well as the system at work; finally, section 5 presents our conclusions and further works.

2 The Physical Layer

The home surveillance system we propose is represented in Figure 1; its components come from several goals to be addressed:

- First, we want to detect of unauthorized accesses to the perimeter, i.e. whenever a potential intrusion occurs it should be detected using infrared radars and ip cameras)

- the system must be able to classify the events detected, distinguishing between real intrusions and false positives e.g. due to leaves moved by the wind or to animals; this goal is achieved through a customizable algorithm [25] that can be tailored to both indoor and outdoor environments
- Finally, once real intrusions are recognized, the system provides a notification via an iPhone® application so the user is immediately warned; he should also be able to view real-time images from cameras, so real-time countermeasures as siren activation or police action request can be taken.

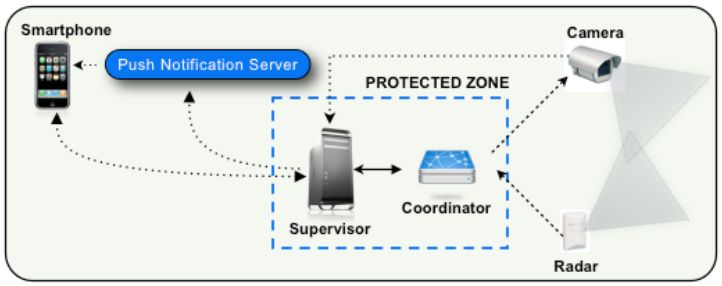

Figure 1: Logical schema of the proposed suirveillance system

Referring to Figure 1, the system key component is the supervisor which manages the network, the data coming from devices (cameras and radars) as well as remote iPhone connections and finally processes video images for intrusions detection.

The supervisor used in our experiments was a medium-performance notebook, including a Pentium® dual-core CPU @ 2.30GHz and 2GB RAM, however the system does not require significant CPU/bandwidth resources, so it can be easily implemented on a low-cost 32-bit microcontroller provided with a serial interface (to allow communications with the coordinator module).

The coordinator acts as the interface between the supervisor and the ZigBee network; it is implemented using a custom ZigBee module, it manages the low-level network and can be configured and managed via standard serial interface.

Moreover, we used only devices that provide standard TTL output, in order to easily achieve interoperability with any programmable digital circuit. To manage the communication between the devices and the ZigBee module we used a microcontroller that listens to events from the device and controls the ZigBee module.

For instance, the PIR (passive infrared) based component, i.e. a typical residential PIR motion detector with multi-Fresnel lens cover, needs to be interfaced with the ZigBee network; to this purpose, we used the PIC16F628A microcontroller (see Fig.2), identical to the well known PIC16F84 except for its additional USART module, which allows a high level management of serial communication. In addition to standard pins connection, as clock and reset, also the RB0/INT (pin 6) and RB2/TX/CK (pin 8) were connected, in particular the former was used to receive asynchronous interrupt signals coming from the PIR radar, whilst the latter was set up as an output for the PIC16F628A and used to connect to the ZigBee network.

The microcontroller run at a 16MHz frequency clock in order to ensure the compatibility with the ZigBee module The microcontroller waits for signals coming from sensors TTL output; as soon as it receives a signal - i.e. a potential alarm - it alerts the coordinator (via the ZigBee module) that controls the camera using the RS232 protocol.

Note that each module acts as a ZigBee End Device, so they spend most of the time in a sleep state, thus saving energy. This however also requires to ”wake up” the device in order to

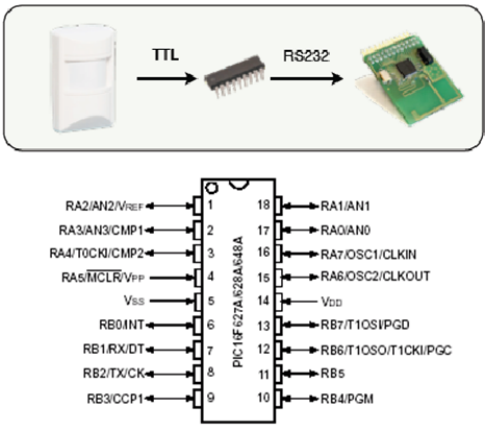

Figure 2: Interfacing a PIR motion detector with the ZigBee network using the PIC16F628A

communicate with it, so the PIC16F628A was programmed so that the device changes from the sleep to the ready state before sending a command, and the same controller allows the device going into the sleep state again after the command has been executed. Finally, note that for the PIR motion detector we also detect potential tampering attempts by connecting the device TAMP pin to the PIC16F628A so that an alarm is sent even in this case.

3 The Application Layer

The architecture described in the previous section operates thanks to the supervisor application, a software that manages devices and evaluates alarms, helping to prune false positive ones. Such application is arranged into the following modules:

- Control module: it controls the physical connection with devices and is based on the QextSerialPort project [27]; this module automatically detects new devices (for instance, a new radar) and allows the supervisor user to add it to the ZigBee network
- ZigBee management module: it is an higher layer manager of the ZigBee devices as cameras and radars;
- Alarm management module: it processes potential alarms to remove false positives and manages the remote alarm notifications as well as control.

All these modules are implemented in C++ within the Qt Framework [26], whereas the OpenCV libraries [28] were used to process images coming from IP camera.

In particular, the available libraries to communicate with the coordinator were developed as an OCX control, so just Windows®-based systems can interface with the system. To overcome this limitation, we develop a C++ version of such libraries; this not only provides more compatibility, but also can be easily interfaced with computer vision (CV) softwares, which are generally written in C language.

The reason for choosing the Qt Framework is that it ensures platform portability and operating system independence; its only drawback is the Phonon framework used by Qt to manage the multimedia layer. Phonon is an abstract framework whose implementation depends on the operating systems, in particular it leverages the Quicktime® libraries for Mac OS-X, DirectShow® libraries within the Windows® platform and GStreamer for Linux OS. Phonon does not allow

an easy single frame management of a video stream, so we exploited the QTKit and QuickTime frameworks to write a class that provided these functionalities.

A complete UML diagram of the supervisor application is depicted in Fig.3, in particular:

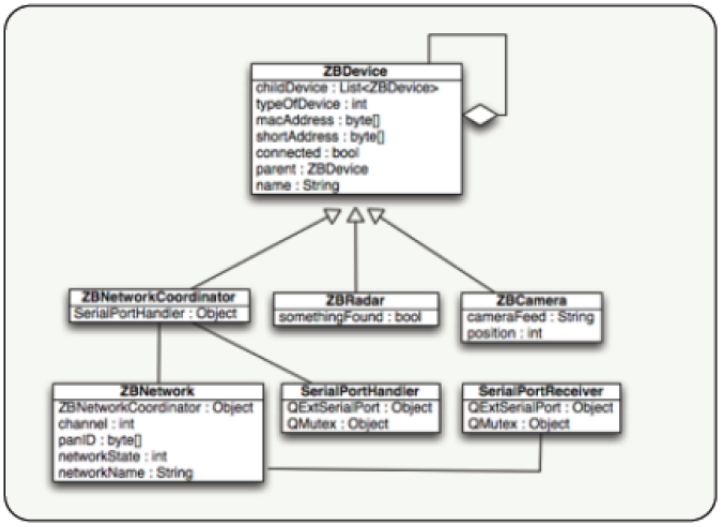

Figure 3: UML diagram of the proposed system

- the ZBDevice class represents a generic ZigBee device with its standards parameters, as the MAC address, the type of device and so on
- three classes are derived from the ZBDevice, i.e. the ZBNetworkCoordinator, the ZBRadar and the ZBCamera, whose roles are clearly understandable; in particular, the coordinator is the only device that can access the physical network via the SerialPortHandler class
- the ZBNetwork class describes the underlying ZigBee network, whereas asynchronous signals in the network are managed by the SerialPortReceiver, that forwards such signals to the right device, for instance as soon as the coordinator is turned on, it sends a message to its supervisor with its mac address in order to be registered following a process mediated by the SerialPortReceiver which also exploits the SerialPortHandler to provide message acknowledgements

A snapshot of the supervisor application is presented in fig.4.

In the following we discuss how the alarms are detected, whereas the subsection 3.2 shows how the remote notification occurs whenever a relevant alarm is detected.

3.1 Alarms detection

The MP4 video stream generated by the IP camera is provided to the supervisor's alarm processing module through a Real Time Streaming Protocol (RTSP) server connected via the wireless connection; this allows the evaluation of whether an intrusion occurred and an alarm should be generated or not.

The received frames are evaluated through the algorithm (details can be found in [25]) in order to assign a precision (i.e. a numeric value) thus establishing whether a false positive has been detected.

To do this, the first (trivial) solution we adopted was to evaluate the difference between the current frame and the previous one, applying a threshold to detect a binary pattern; this worked

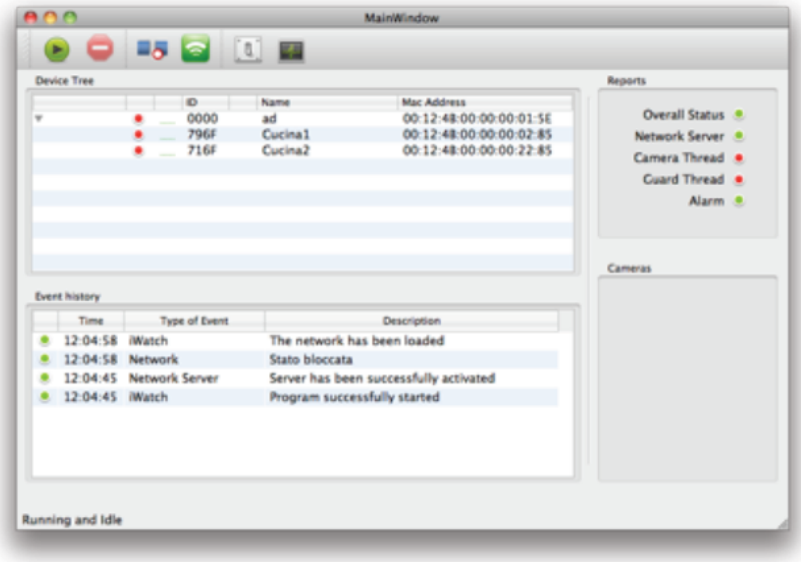

Figure 4: Snapshot of the supervisor application

in almost static background scenarios, as for indoors environments (e.g. a room) with constant lighting, but it is not suitable when frames background is not static, for instance when tree leaves are moved by the wind.

The next step was to apply the Gaussian Background Model [29], citeLee2:2005, in order to effectively remove the (even dynamic) background; the algorithm implemented in the supervisor application works following the steps illustrated in fig. 5: it first converts colour images into a black and white format for binary processing (1), then it detects the foreground (2) and extracts the corresponding bounding rectangle (3), i.e. the area where the gaussian model detected relevant information.

Figure 5: Alarms sources extraction process of the supervisor algorithm

Bounding rectangles across subsequent frames are compared for a given time interval and if something is detected with a specified precision, an alarm is generated; the system stores in XML format any relevant event, should it determine an alarm or not, in accordance with a set of severity levels.

Note that alarm detection is completely customizable by setting algorithm parameters, so a proper configuration allows to correctly distinguish between false positives/negatives and real alarms; for instance, the threshold can be reduced for indoor environments (where the background is almost static) so even a little movement will be detected; similarly, the minimum bounding rectangle can be increased, in order to discard cats/dogs movement detection in outdoor environments and so on.

Parameters should be tailored to the actual scenario the system will be installed into (section 4 shows a typical calibration session); in fig. 6 the C struct implementing such parameters is shown, together with the screenshot of the supervisor application used to set up their values. Details about the meaning of parameters and how they affect the CV recognition process can be found in the OpenCV documentation [28].

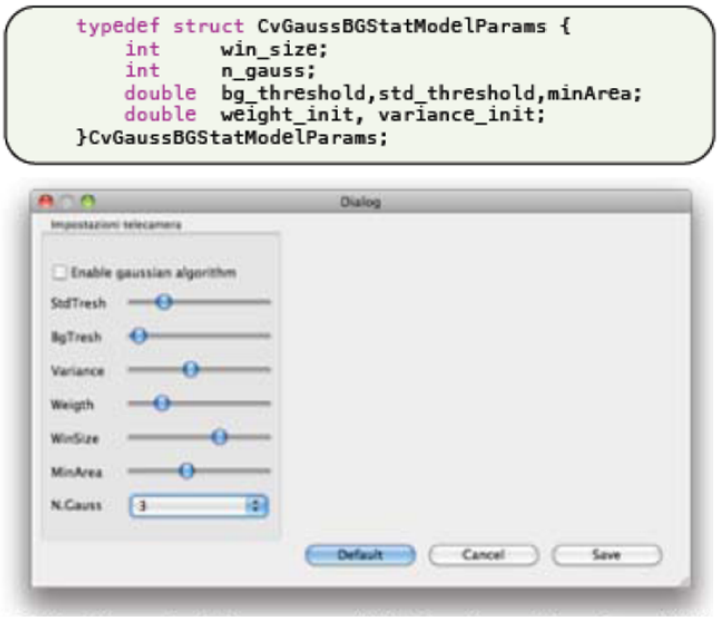

Figure 6: CV Parameters set up

3.2 Alarm notifications and remote control

As soon as a (possibly real) alarm is detected, the system sends notification to the user, also providing a remote control of overall systems functionalities. These are implemented making use of the well-known push technology [31], citePohja:2009, in particular we used the Apple Push Notification Service (APNS in the following) [33].

The supervisor application is first registered at the server providing APNS; then, whenever the supervisor needs to send a notification, it establishes a secure connection to the APNS using the certificate obtained by the APNS during the registration, and finally data are transmitted. The format of the packet being sent is illustrated in figure7, where the deviceToken is a 32 byte device identifier and the Payload represents the notification, created according to the JSON format, a lightweight data-interchange format based on a subset of the JavaScript language [34].

During our tests notifications were received within about 4 seconds since the request, that can be considered a good response time for this kind of application.

To manage notifications and provide remote control functionalities, a mobile application operates in conjunction with the supervisor's software counterpart.

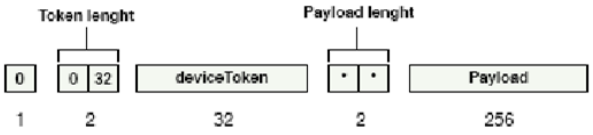

Figure 7: The format of notifications sent by the supervisior

This application was developed in Objective-C within the Cocoa Touch framework [35], and it displays notifications received from the push server, also allowing the management of the ZigBee network (e.g. enabling or disabling devices) and the possibility of remotely examining current camera video stream from the mobile phone, checking whether it deals with a real intrusion on not and applying proper countermeasures, as an alarm activation or a police call for on-site actions.

Finally, note that the system also performs a complete log of all events, each stored with a severity level, a timestamp and a description, so that it will be possible to analyze who, why and when a given alarm notification has been triggered.

4 Experiments and Results

In order to evaluate the performance of the intrusion detection system presented in previous sections, we performed two types of test, the former to assess how the calibration affects alarms classification and the latter is about the behaviour of the ZigBee network, in particular for what concerns the largest area that can be effectively kept under surveillance.

To assess false positive alarms pruning, we used a Panasonic^{TM} standard VGA resolution camera in two different scenarios, an indoor environment (a room) with scarce illumination and the an outdoor environment (a garden in front of a house) with strong lighting.

For each scenario we performed 50 test sessions, 25 with an intruder walking in front of the camera and the remaining 25 without intrusion. Finally, results from each scenario have been evaluated first without any configuration for the detection algorithm parameters (i.e. with default values) and then providing calibration for such values (details are here omitted [25]) In the following we present the test results using diagrams that show the number of tests according to the percentage of accuracy reached (ranging from 0 to 100%).

The measures for the indoor scenario with no calibration (see figure 8 a) show unsatisfactory results. In the set of 25 tests with an intruder, 20% of them shown negative results, i.e. the system did not detect the intruder. The same scenario with no intruder shows incorrect results for 44% of tests (a non-existent intruder is detected); this was due to missing calibration and also by changes in the light coming from the windows (which lead to false positive detection).

After the calibration, results significantly improved as figure 8 b) shows. The algorithm indeed detected the intruder with high accuracy (no false negative occurred), and similarly in the case of absence of the intruder (no false positive have been detected).

The second scenario concerns an outdoor environment with strong lighting (a garden in front of a house in the morning); results are displayed in figure 9. The significantly increasing in illumination allow better results even with no calibration (see fig. 9 a); the algorithm indeed shown high precision in detecting an intruder (no false negatives), whilst we have just 16% of false positives in the case of no intrusion. After the calibration results were further improved (false positives were completely prevented).

The second set of tests was about the ZigBee network, in particular we want to assess the largest area that can be effectively kept under survelliance. To quantify this effectiveness, we

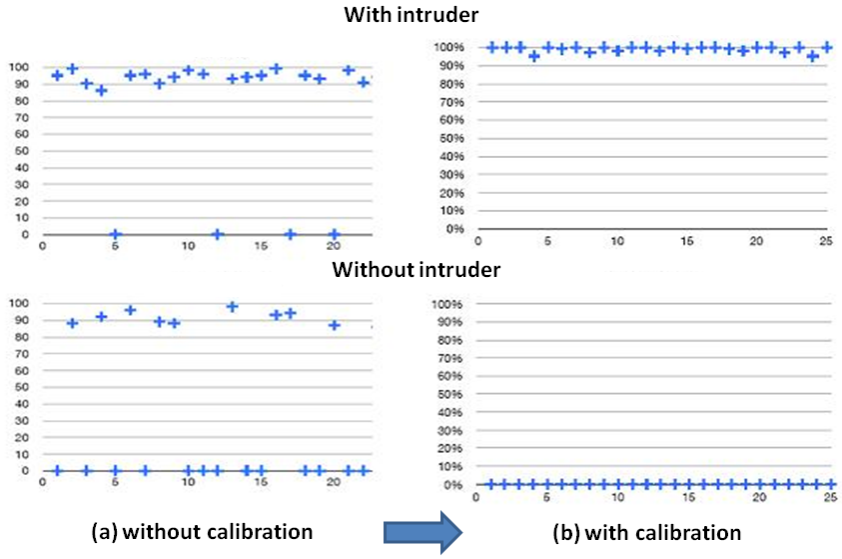

Figure 8: Indoor scenario

used the Link Quality Indicator (LQI), an 8 bit sequence representing a figure of the quality of the link between two ZigBee nodes [36].

To perfom such tests we used a system made up of a coordinator module and a router module which was positioned at different locations (increasing the distance from the coordination) along a given perimeter. At each position we performed 25 measures; results are shown in fig. 10, where the value of LQI for distance from 0 to 12m is plotted, in particular the solid line represents the theoretical LQI value whereas the other line represents measured LQI values. Each point is the average of 25 measures; in the same figure we report some LQI measurements together with corresponding variance values; note that variance remains limited until a distance of 4 meters, whereas it increases significantly over this distance, this is due to the presence of a wall at 5 mt, which hinders the communication between ZigBee modules.

Tests revealed worse performances than those declared on the component datasheets, indeed the system worked well up to a distance of 20 meters even if the LQI decreased significantly since a distance of 5 meters. However, we noted that even with a low LQI the communications between ZigBee modules was acceptable, in particular some connection losses were detected but the system was able to restore the connection in few seconds.

5 Conclusions and Future Works

Several intelligent and pervasive applications are being developed within the domotics context. in particular, an intelligent surveillance system for home security was presented in this paper. It exploits the ZigBee protocol and it can detect and classify intrusions to discard false positive and negative alarms, also providing remote control functions and cameras live streaming to allow users to analyze who and why is triggering alarms. We also presented results about both the detection algorithm effectiveness and the ZigBee network performances.

Some future works include the following issues:

- the application currently was tested on Windows or MacOS X operating systems; to increase portability, we are planning to test it in Linux-based platforms;

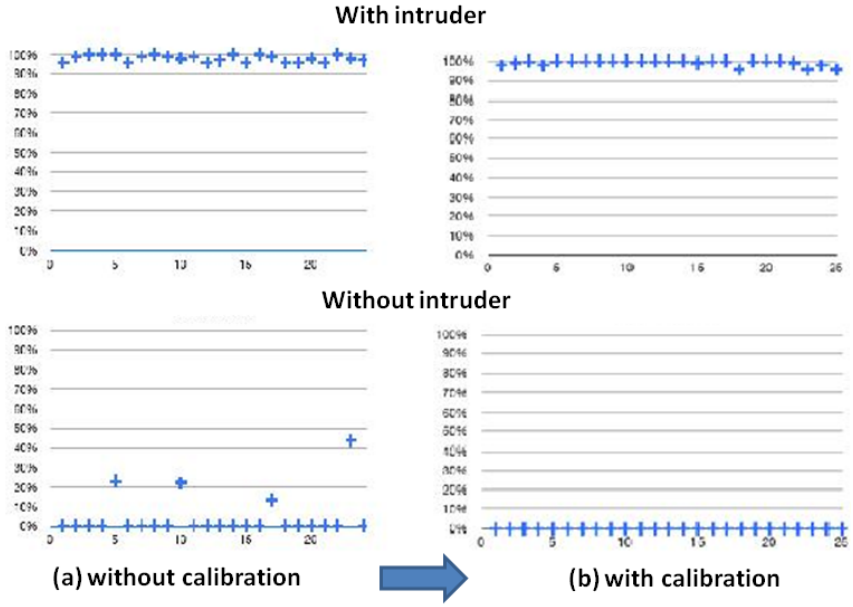

Figure 9: Outdoor scenario

Distance	Average LQI	Absolute variance	Variance (%)
0,8 m	140	0,0	0,000%
1,2 m	132	8,1	6,121%
1,5 m	140	0,9	0,644%
2 m	133	2,4	1,805%
2,5 m	123	0,0	0,000%
3 m	123	0,0	0,000%
4 m	120	0,0	0,000%
6 m	79	17,6	22,373%
9 m	75	12,2	16,267%
12 m	64	22,4	34,879%

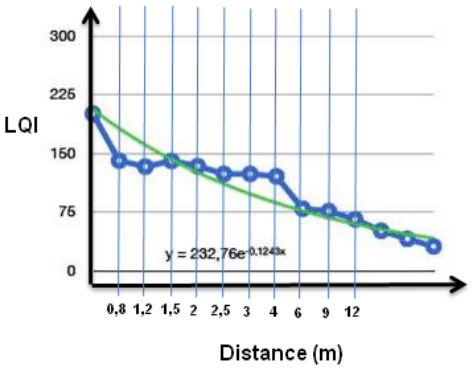

Figure 10: LQI versus distance between coordinator and router ZigBee modules

- the system includes just a coordinator and a router with a camera; adding other components, e.g. smoke and gas sensors, household appliances etc., allows to build a system that guarantees security and safety in a global fashion as electronic keys or biometric sensors to improve the system's capability.

- the algorithm we used to evaluate intrusions was quite simple, and it should be intended as the best choice for first experiments (it indeed did not affect on the overall system performance); better yet slower algorithm should be tested.

6 Acknowledgement

This paper was inspired by a work developed in cooperation with ValueTeam IT consulting and solutions (http://www.valueteam.com). Particularly, we thank Francesco Consoli as Senior Manager at Security Division, for his guidance and support to this work. We also thank Danilo Torrisi for the implementation and test of the system.

Bibliography

[1] Jacobson, J., Understanding Home Automation, *Electronic House*, 14(6):18-21, 2001

[2] Rodden, Tom and Benford, Steve, The evolution of buildings and implications for the design of ubiquitous domestic environments, *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, New York, USA, pp. 9-16,2003

[3] Lorente, S., Key issues regarding Domotic applications, *International Conference on Information and Communication Technologies: From Theory to Applications*, pp. 121 - 122, 2004

[4] Park, Sang Hyun and Won, So Hee and Lee, Jong Bong and Kim, Sung Woo, Smart home-digitally engineered domestic life, *Personal and Ubiquitous Computing*, Springer London, 7(3):189-196, 2003

[5] Luo, R.C. and Hsu, T.Y. and Lin, T.Y. and Su, K.L., The development of intelligent home security robot, *Mechatronics, 2005. ICM '05. IEEE International Conference on*, pp. 422 -427, 2005

[6] European Institute for Corporate Security, *http://www.eicsm.org/index.html*

[7] Chunduru, V. and Subramanian, N., Perimeter-Based High Performance Home Security System, *Consumer Electronics, 2007. ISCE 2007. IEEE International Symposium on*, pp. 1 -7, 2007

[8] Russell, Stuart J. and Norvig, Peter, Artificial intelligence: a modern approach, Prentice-Hall, Inc., 2010

[9] Nieuwdorp, Eva, The pervasive discourse: an analysis, *Comput. Entertain.*, ACM, New York, NY, USA, vol. 5(2):13-13, 2007

[10] Bell, Genevieve and Dourish, Paul, Yesterday's tomorrows: notes on ubiquitous computing's dominant vision, *Personal Ubiquitous Comput.*, Springer-Verlag, London, UK, 11(2):133-143, 2007

[11] Jun Hou and Chengdong Wu and Zhongjia Yuan and Jiyuan Tan and Qiaoqiao Wang and Yun Zhou, Research of Intelligent Home Security Surveillance System Based on ZigBee, *Intelligent Information Technology Application Workshops, International Symposium on*, IEEE Computer Society, Los Alamitos, CA, USA, pp. 554-557, 2008

[12] Krausz, Barbara and Herpers, Rainer, Event detection for video surveillance using an expert system, *AREA '08: Proceeding of the 1st ACM workshop on Analysis and retrieval of events/actions and workflows in video streams*, Vancouver, British Columbia, Canada, ACM, New York, NY, USA, pp. 49-56,2008

[13] Want, Roy and Pering, Trevor, System challenges for ubiquitous & pervasive computing, *ICSE '05: Proceedings of the 27th international conference on Software engineering*, St. Louis, MO, USA, ACM, New York, NY, USA, pp. 9-14,2005

[14] McCullough, Malcolm, Digital Ground: Architecture, Pervasive Computing, and Environmental Knowing, MIT Press, Cambridge, MA, USA, 2004

[15] Weiser, M. and Gold, R. and Brown, J. S., The origins of ubiquitous computing research at PARC in the late 1980s, *IBM Syst. Journal*,IBM Corp., Riverton, NJ, USA, 38(4):693-696, 1999

[16] Weiser, Mark, The computer for the 21st century, *SIGMOBILE Mob. Comput. Commun. Rev.*, ACM, New York, NY, USA, 3(3):3-11, 1999

[17] Hansmann, Uwe and Nicklous, Martin S. and Stober, Thomas, Pervasive computing handbook, Springer-Verlag New York, Inc., New York, NY, USA, 2011

[18] Miori, Vittorio and Russo, Dario and Aliberti, Massimo, Domotic technologies incompatibility becomes user transparent, *Commun. ACM*, New York, NY, USA, 53(1):153-157, 2010

[19] Egan, D., The emergence of ZigBee in building automation and industrial control, *Computing & Control Engineering Journal*, 16(2):14-19, 2005

[20] Fukui Kiyoshi and Tanimoto Akira and Fukunaga Shigeru, ZigBee Technology for Low-Cost and Low-Power Radio Communication Systems, *Journal of the Institute of Electronics, Information and Communication Engineers*, vol. 88(1):40-45, 2005

[21] Wang Dong and Zhang Jin-rong and Wei Yan, Building Wireless Sensor Networks (WSNs) by Zigbee Technology, *Journal of Chongqing University (Natural Science Edition)*, 29(8):95-98, 2006

[22] Li Cai and Nina Dai, The Home Security System Based on ZigBee Technology, *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, pp. 1-2, 2010

[23] The ZigBee Protocol OB/EL, http://www.digi.com/technology/rf-articles/wireless-zigbee.jsp, *http://www.digi.com/technology/rf-articles/wireless-zigbee.jsp*, 2009

[24] Carchiolo V. et al, Pervasive home security: an intelligent domotics application, *Intelligent Distributed Computing IV (IDC 2010) conference*, pp. 145-154, 2010

[25] Danilo Torrisi, Sistema di sicurezza perimetrale con allarmistica e controllo a distanza, 2009, *Tech. Report - Dipartimento di Ingegneria Informatica e delle Telecomunicazioni - Facolta' di Ingegneria - Universita' di Catania*

[26] Qt Nokia Framework, *http://qt.nokia.com/about/news/nokia-releases-qt-4.6.2*

[27] QextSerialPort, *http://qextserialport.sourceforge.net/*

[28] OpenCV libraries, *http://sourceforge.net/projects/opencvlibrary/*

[29] Brian R. Williams and Ming Zhang, Multiple Dimension Chrominance Model for Background Subtraction, *Computational Intelligence*, pp. 438-443, 2005

[30] Lee, Dar-Shyang, Effective Gaussian Mixture Learning for Video Background Subtraction, *IEEE Trans. Pattern Anal. Mach. Intell.*, IEEE Computer Society, Washington, DC, USA, 27(5):827-832, 2005

[31] Eugster, Patrick Th. and Felber, Pascal A. and Guerraoui, Rachid and Kermarrec, Anne-Marie, The many faces of publish/subscribe, *ACM Computing Survey*, ACM, New York, NY, USA, 35(2):114-131, 2003

[32] Pohja, Mikko, Server push with instant messaging, *SAC '09: Proceedings of the 2009 ACM symposium on Applied Computing*, Honolulu, Hawaii, ACM, New York, NY, USA, pp. 653-658, 2009

[33] Apple Push Notification Service, *http://developer.apple.com/ IPhone/ library/ documentation/ NetworkingInternet/ Conceptual/ RemoteNotificationsPG/ ApplePushService/ ApplePushService.html*

[34] Introducing JSON, 2011, *http://www.json.org*

[35] Cocoa Touch Framework, *http://developer.apple.com/iphone*

[36] Philip Orlik and Jinyun Zhang and Bharat Bhargava and Gang Ding and Gang Ding and Zafer Sahinoglu and Zafer Sahinoglu, Reliable Broadcast in ZigBee Networks, *In Proceedings of SECON IEEE Conference*, 2005