AN RF-BASED SURVEILLANCE SYSTEM USING COMMERCIAL OFF-THE-SHELF WIRELESS LAN COMPONENTS

Jianjun Chen, Zoltan Safar, John Aa. Sørensen and Kåre J. Kristoffersen

Department of Innovation, IT University of Copenhagen, Copenhagen, Denmark
E-mail: {chen, safar, jaas, kjk}@itu.dk

ABSTRACT

In this paper, we present an indoor surveillance system implemented using the commercially available wireless LAN infrastructure. Taking advantage of the programmability of the wireless network nodes, we have built a wireless network that has two modes of operation: the communication mode, when the network is used as a traditional wireless LAN, and the surveillance mode, when the network is used as a distributed sensor network that can detect illegal intrusion by detecting changes in the propagation environment caused by the intruder. The experimental results show promising through-the-wall intrusion detection capabilities in an office environment.

1. INTRODUCTION

There has been an abundance of activities focusing on the development and deployment of wireless networks. Results of these efforts so far are the widely utilized IEEE 802.11a, b, and g Wireless Local Area Network (WLAN) standard, the Bluetooth technology, and the 2G and 3G cellular data networks. These wireless data networks will be pervasive and will be accessed from virtually anywhere, offering a wide range of services. Moreover, the programmability of such network nodes has been continuously increasing. The latest products of this tendency are the ROSE development environment [1], which is an access point development kit for WLANs, the Host AP software package [2], which allows a computer and an 802.11 WLAN card to be used as a programmable access point, and the programmable smart phones, which are expected to be widespread in the next 3-5 years. Therefore, the future wireless communication infrastructure is likely to consist of highly programmable wireless network nodes that will provide flexible platforms for both communication and computation.

In this paper, we present an indoor surveillance system implemented using the existing WLAN infrastructure. Taking advantage of the programmability of the wireless network nodes, we have built a wireless network that has two modes of operation: the communication mode, when the network is used as a traditional WLAN, and the surveillance mode, when the network is used as a distributed sensor network that can detect illegal intrusion. In surveillance mode, the nodes constantly monitor the environment by analyzing the properties of the received signals, such as the received signal strength. When entering a site covered by such a network, the intruder will disturb the physical propagation environment, causing change in the characteristics of the received signals, and this change is used for intrusion detection. Due to the multiple modes of operation, this kind of network has been coined as a multimodal wireless network.

So far, the problems of wireless communication and ”physical” intrusion detection (i.e. detecting a person or persons entering private/corporate premises illegally) have been considered as two separate issues, and two different infrastructures have been deployed: one for communication and one for surveillance/security. However, if the communication infrastructure could also be used for security purposes, the deployment of the additional infrastructure could be avoided or reduced, resulting in a considerably more cost-effective solution. As a consequence, our objective was to develop a surveillance system with real-time detection capabilities based on the existing WLAN infrastructure, possibly already deployed, and to achieve this by changing only software components.

2. RELATED WORK

Previous experiments investigating the impact of moving objects/humans on the propagation environment [3], [4] have shown that significant variations can be observed in the received signal strength and the rms delay spread. However, those measurements were carried out using specialized equipment, and not low-cost, commercial off-the-shelf devices, such as a WLAN card, and the authors did not propose any signal processing architectures or systems for intrusion detection.

The idea of using WLAN for surveillance was first proposed in [5], and a single-receiver (or single-node) model was described for the received signal parameter of interest. In [6], we considered the problem of distributed surveillance with multiple wireless network nodes. We developed a multi-sensor model for the received signal parameters and derived a parameter change detector based on the generalized likelihood ratio test (GLRT) for a multiple transmitter/multiple receiver scenario. However, that work focused on the algorithmic and signal processing aspects of the problem, ignoring the implementation, integration and system design issues such as how to map the data fusion and detection algorithms onto the layers of the protocol stack, how to organize the communication among the network nodes and how to write application software and modify the system software to realize the desired functionality. The solution to these problems is the main contribution of this paper.

3. THE SURVEILLANCE SYSTEM

In communication mode, the wireless transceiver nodes (access points, or nodes in an ad-hoc network) implement the functionality of a traditional wireless network. In surveillance mode, the nodes go through the following two phases: the training phase and the detection phase.

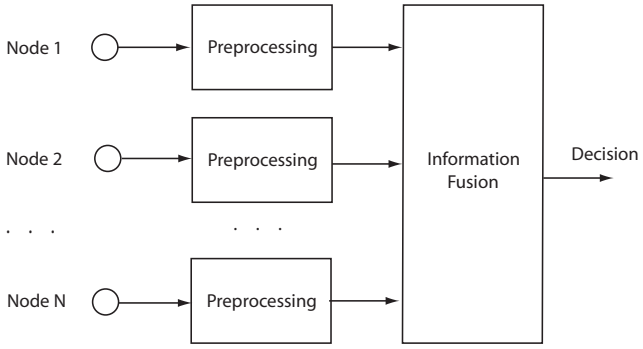

Figure 1: The signal flow diagram of the surveillance system

During the training phase, the steady state of the propagation environment is estimated when it is ensured that no intruder is present. First, the signal acquisition takes place: the nodes scan the environment by transmitting, for example, one by one in a round robin fashion, while the rest of the nodes receive the transmitted signal. Then, based on the received observations, the nodes estimate the signal (and noise) parameters corresponding to each transmitter-receiver pair. The training phase before activating the surveillance system is necessary since the propagation environment may change significantly during an inactive period of the surveillance system. For example, workers at a company may move furniture or leave doors open or closed during daytime.

The second phase is the detection phase, when the surveillance system detects the changes in the signal parameters of interest compared to the steady state. The environment is scanned similarly to the training phase, and after each scanning cycle, the received signal is processed as depicted in Figure 1. The observations are processed by a preprocessor to extract the relevant characteristics of the propagation environment, called features, which may include time of arrival, angle of arrival, the strength of the received signal, channel impulse response, or a combination of these. In case of the 802.11 WLANs, the only observations related to the propagation environment are the received signal strength values corresponding to each received frame, so they will be the basis for feature extraction. The obtained set of such features is then combined by an information fusion function, which produces a single output that is used to decide whether an intruder is present in the system or not.

3.1 System Architecture

The network nodes in the implemented prototype of the surveillance system were IBM Laptops with 802.11b ZyAIR B-100 WLAN cards. The installed software was the Host AP package [2], running under the Linux Redhat 9.0 operating system. We chose the classical master-slave architecture: one network node was designated as a master, and the rest of the nodes acted as slaves. The master node controlled the scanning and data acquisition process by sending commands to the slave nodes via a wired 100 MBit/s Ethernet LAN. The control channel was implemented as a wired network for its high data rate and reliability, but the system can easily be configured to use the wireless medium to exchange control information. The master node also gathered the signal strength information from the slave nodes, ran the estimation and detection algorithms, and decided whether to generate an alarm signal or not. One by one, each slave node received commands from the master via the wired network to transmit through its wireless link, while the rest of the nodes became receivers. Upon the reception of the transmitted wireless frames, the slave nodes sent the corresponding signal strength information to the master node via the wired network and waited for the next command.

3.2 Scanning and Signal Acquisition

The control channel between the master node and the slave nodes was implemented using standard UDP sockets, and the master node performed point-to-point communication with each slave node. Since on the wired LAN the packet loss rate was very small, the overhead of the TCP protocol was undesirable. However, packet loss did occur on rare occasions, so some extra measures were included in the communication protocol to be able to recover from packet loss/corruption.

For the wireless broadcast channel, neither TCP nor UDP were found suitable. These protocols cannot make the received signal strength values available for the application layer, and changing the protocols such that all TCP/UDP sockets would forward this information was undesirable since wired TCP/UDP packets do not have this information, and one of the objectives was to be able to use the system in communication mode as a regular data network. Thus, we used raw sockets to send and receive our own user-defined MAC frames. Moreover, the Host AP driver was modified in such a way that when a wireless frame was received, the driver copied the signal strength value from the driver-level Rx descriptor into a specific field in the user-defined MAC frame, and the slave node process could read this information from the packet received through the raw socket. Since raw sockets are ”shortcuts” between the application layer and the logical link control layer, and all wireless frames were broadcast (the broadcast destination MAC address was used), the error-free reception of all wireless frames was not guaranteed. During one scanning cycle, each slave node transmitted 50 wireless frames in a ”burst”, and the detection algorithm was designed to be able to work even if there were gaps in the received signal strength values due to lost or corrupted wireless frames.

The communication protocol realizing the scanning cycles was implemented according to the timing diagram shown in Figure 2. The rectangles above the time axes represent packets sent over the wired communication link (UDP packets), while the rectangles below the axes represent wireless transmission and reception (raw socket frames). In the beginning of the cycle, the master node selected one slave and sent it a command to become the transmitter for the next burst. Upon the reception of this command, the addressed node sent a burst of 50 wireless broadcast frames, and the rest of the nodes received some of these frames and saved the corresponding signal strength information. After finishing with the wireless transmission, the transmitter slave node sent back a ”Burst Done” message to the master. Thus, the master node knew that one burst of the scanning cycle had been completed, so the master selected the next node to become the transmitter and sent a command to it. Since these commands were sent over a UDP socket, some of the messages were lost or corrupted, so after transmitting each command, the master started a timer and waited for the ”Burst Done” message. If timeout occurred, the master node attempted a number of retransmissions before declaring system failure.

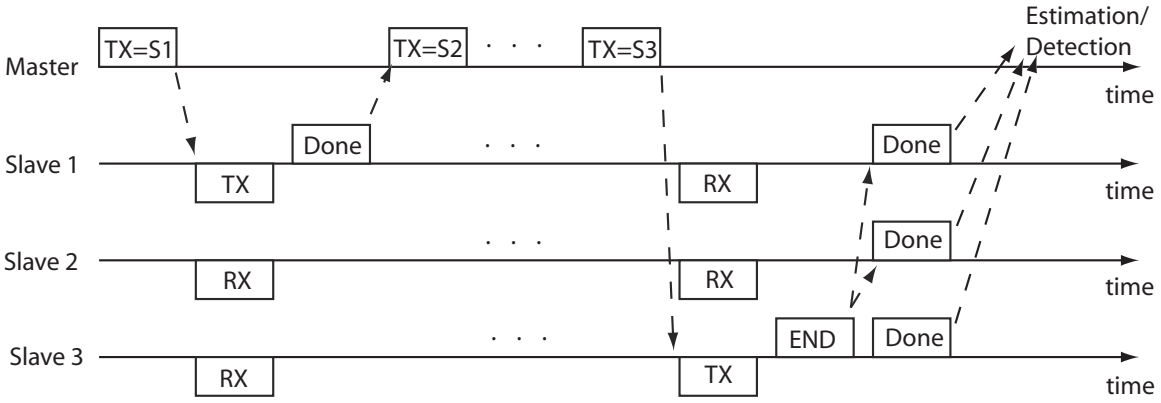

Figure 2: The timing diagram of one scanning cycle

The last burst in the scanning cycle was special because this was when the slave nodes sent the collected signal strength information to the master node. The last slave node was informed that this was the last burst in this scanning cycle by a special field in the command from the master node, and after finishing the wireless burst, the slave node sent "Cycle End" messages through a the wired network to the rest of the slave nodes. Then, all nodes (including the last transmitter) sent "Burst Done" messages to the master node, and these messages contained the received signal strength values collected during the scanning cycle. The master node received these messages (again, a timer was started and stopped to be able to recover from packet loss/corruption), and if enough observation data had been collected, it executed the estimation and detection algorithms. Otherwise, a new scanning cycle was started.

3.3 Signal Processing

The estimation and detection algorithms were implemented following the description in [6]. These algorithms were based on the received signal model depicted in Figure 3. The true value of the received signal parameter (denoted by $l^0_{m,n}$, when the m-th slave node was the transmitter and the n-th slave node was the receiver) was shifted by an unknown bias ($B_{m,n}$), resulting in the biased parameter ($l_{m,n}$). The bias represents measurement inaccuracy due to transmitted and received signal features that are not calibrated and/or not standardized, such as the 802.11 transmit power and signal strength value calculation. The true signal parameter was further disturbed with zero-mean, white Gaussian noise $z_{m,n}(t)$ with variance σ^2_n at time t. The observations are usually only available in quantized form (for example, 802.11 signal strength values), so the observed values were $\{y_{m,n}(t)\}$, the quantizer indices corresponding to the noisy and biased signal parameter values $\{x_{m,n}(t)\}$.

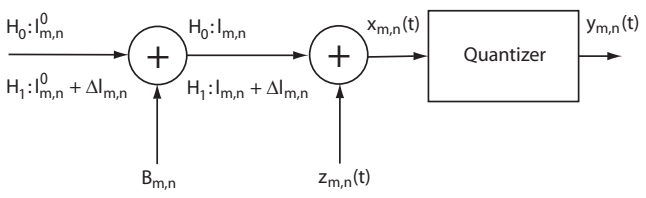

Figure 3: The received signal model

During the training phase, the signal and noise parameters $l_{m,n}$ and σ_n were estimated by an approximate and an iterative maximum likelihood (ML) estimator using a larger number (1000) of training samples received from each slave node. The approximate ML estimates provided the initial values for the iterative estimator. The observations were the signal strength values corresponding to the received wireless frames, and the true value of the (biased) signal strength was the signal parameter $l_{m,n}$ to be estimated.

In the detection phase, the surveillance system detected the changes $l_{m,n}$ in the parameters $l_{m,n}$ based on a small number (50) of observations. The values of $l_{m,n}$ were estimated by an approximate ML estimator. Since the objective was to maximize the probability of detection for a given probability of false alarm, the detector performed a GLRT. As a natural consequence of the multi-node GLRT formulation, the preprocessor calculated the features, which were functions of the estimated signal and noise parameters, and the feature-in-decision-out information fusion was performed by a simple summation and threshold comparison.

4. EXPERIMENTAL RESULTS

To illustrate the performance of the implemented surveillance system, we performed some experiments. The experimental site, depicted in Figure 4, was the D section on the 5th floor of the IT University of Copenhagen building. The walls were made of plaster and concrete, and the doors were made of wood. At the time of the experiments, the 5th floor was empty, so there were no furniture or any equipment present. Three slave nodes were placed in different rooms, shown by the numbered triangles in the figure. The directions of the triangles indicate the orientation of the WLAN cards.

The event to be detected was the opening of a door by a person, which modeled the intrusion into a secure area. Before the event, the person stood beside the closed door in one of the rooms. Then, he opened the door (to about $40-45^\circ$), moved through the door opening into the corridor and closed the door behind him. Three experiments were conducted at three different locations. The corresponding rooms and doors are labeled with "A", "B" and "C" in Figure 4. The performance of the system was evaluated by the traditional detection performance metric: the probability of detection (the probability that an event is detected if the event occurs) versus the probability of false alarm (the probability that an event is detected provided that the event does not occur).

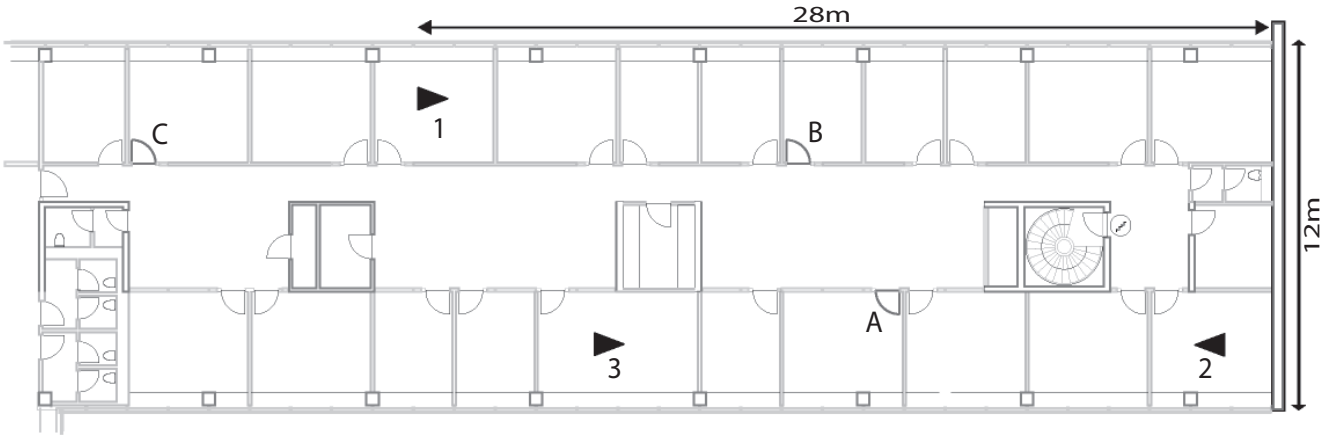

Figure 4: The experimental site

The slave nodes transmitted the wireless frames in bursts of 50, and each slave node transmitted one burst in one scanning cycle. The training phase consisted of 20 scanning cycles. In the detection phase, a decision regarding the detection of an event was made after each scanning cycle.

To calculate the average probability of detection, we performed 200 door-opening events according to the description above. In order to average over the estimation error incurred by the training phase, the system was periodically retrained after 20 event detection attempts (detection phases). To calculate the average probability of false alarm, the door was closed, and 10000 scanning cycles and detection decisions were made. The system was periodically retrained after 1000 scanning cycles (detection phases).

The experimental results are depicted in Figure 5, which shows the obtained average probability of detection versus average probability of false alarm curves. The curves "A", "B" and "C" correspond to locations "A", "B" and "C", respectively. As can be seen form the figure, the door-opening events at locations "A" and "B" could be perfectly detected, without any false alarms. This is not surprising since these locations are in the vicinity of the direct line-of-sight paths between some of the slave nodes. Thus, moving objects and human body caused several dB change in the received signal strength values, which could be detected with high accuracy. Experiment "C" was a more challenging case because the door-opening event only caused changes in the reflected component of the received signals (the direct paths were not disturbed), and the event and the slave nodes were separated by multiple walls. As a consequence, the detection performance degraded significantly: at 10^{-2} false alarm probability, the door-opening event was detected only about 80% of the time.

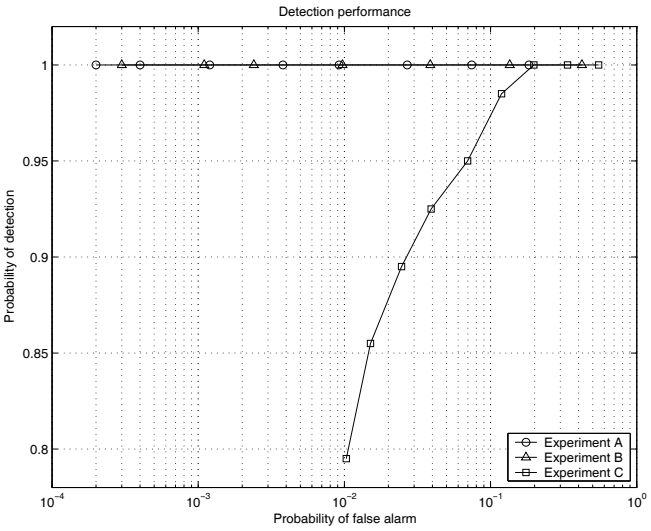

Figure 5: The detector performance

5. CONCLUSION

We described an implementation of an indoor surveillance system built from commercially available, off-the-shelf WLAN components. The surveillance system performed continuous environment scanning and was capable of giving real-time alarm signals based on detected changes in the received signal strength values. Even though the exact limits on the performance and the robustness of the system are yet to be investigated, the experimental results showed promising intrusion detection capabilities.

REFERENCES

[1] http://rose.radionet.fi.

[2] http://hostap.epitest.fi.

[3] A. Kara and H. L. Bertoni, "Blockage/Shadowing Polarization Measurements at 2.4GHz for Interference Evaluation between Bluetooth and IEEE 802.11 WLAN", *IEEE Antennas and Propagation Society International Symposium*, Vol. 3, pp. 376–379, 2001.

[4] K. Pahlavan, *Wireless Information Networks*, Wiley & Sons, 1995.

[5] J. Sørensen, Z. Safar, J. Chen, K. Kristoffersen and M. Schiøtz, "Indoor Surveillance with Multimodal Wireless Networks", *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 242-245, December 2004.

[6] Z. Safar, J. Sørensen, J. Chen and K. Kristoffersen, "Multimodal Wireless Networks: Distributed Surveillance with Multiple Nodes", to appear, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2005.