CEPIS UPGRADE is the European Journal for the Informatics Professional, published bi-monthly at <http://cepis.org/upgrade>

Publisher
CEPIS UPGRADE is published by CEPIS (Council of European Professional Informatics Societies, <http://www.cepis.org/>), in cooperation with the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*, <http://www.ati.es/>) and its journal Novática

CEPIS UPGRADE monographs are published jointly with Novática, that publishes them in Spanish (full version printed; summary, abstracts and some articles online)

CEPIS UPGRADE was created in October 2000 by CEPIS and was first published by Novática and INFORMATIK/INFORMATIQUE, bimonthly journal of SVI/FSI (Swiss Federation of Professional Informatics Societies)

CEPIS UPGRADE is the anchor point for UPENET (UPGRADE European NETwork), the network of CEPIS member societies' publications, that currently includes the following ones:
- inforewiew, magazine from the Serbian CEPIS society JISA
- Informatica, journal from the Slovenian CEPIS society SDI
- Informatik-Spektrum, journal published by Springer Verlag on behalf of the CEPIS societies GI, Germany, and SI, Switzerland
- ITNOW, magazine published by Oxford University Press on behalf of the British CEPIS society BCS
- Mondo Digitale, digital journal from the Italian CEPIS society AICA
- Novática, journal from the Spanish CEPIS society ATI
- OCG Journal, journal from the Austrian CEPIS society OCG
- Pliroforiki, journal from the Cyprus CEPIS society CCS
- Tölvumál, journal from the Icelandic CEPIS society ISIP

Editorial correspondence: Llorenç Pagés-Casas <pages@ati.es>
Advertising correspondence: <info@cepis.org>

Subscriptions
If you wish to subscribe to CEPIS UPGRADE please send an email to info@cepis.org with 'Subscribe to UPGRADE' as the subject of the email or follow the link 'Subscribe to UPGRADE' at <http://www.cepis.org/upgrade>

ISSN 1684-5285

CEPIS UPGRADE
The European Journal for the Informatics Professional
http://cepis.org/upgrade

Vol. XII, issue No. 5, December 2011

Farewell Edition

./..

* This monograph will be also published in Spanish (full version printed; summary, abstracts, and some articles online) by **Novática**, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) at <http://www.ati.es/novatica/>.

CEPIS UPGRADE
The European Journal for the Informatics Professional
http://cepis.org/upgrade
Vol. XII, issue No. 5, December 2011

Farewell Edition

Cont.

UPENET (UPGRADE European NETwork)

CEPIS NEWS

Surveillance Systems

An Intelligent Indoor Surveillance System

Rok Piltaver, Erik Dovgan, and Matjaz Gams

This paper was first published, in English, by ***Informatica*** (Vol. 35, issue no. 3, 2011, pp. 383-390). ***Informatica***, <http://www.informatica.si/> is a quarterly journal published, in English, by the Slovenian CEPIS society SDI (*Slovensko Drustvo Informatika* – Slovenian Society Informatika, <http://www.drustvo-informatika.si/>).

The development of commercial real-time location system (RTLS) enables new ICT solutions. This paper presents an intelligent surveillance system for indoor high-security environments based on RTLS and artificial intelligence methods. The system consists of several software modules each specialized for detection of specific security risks. The validation shows that the system is capable of detecting a broad range of security risks with high accuracy.

Keywords: Expert System, Fuzzy Logic, Intelligent System, Real-Time Locating System, Surveillance.

1 Introduction

Security of people, property, and data is becoming increasingly important in today's world. Security is ensured by physical protection and technology, such as movement detection, biometric sensors, surveillance cameras, and smart cards. However, the crucial factor of most security systems is still a human [7], providing the intelligence to the system. The security personnel has to be trustworthy, trained and motivated, and in good psychically and physical shape. Nevertheless, they are still human and as such tend to make mistakes, are subjective and biased, get tired, and can be bribed. For example, it is well known that a person watching live surveillance video often becomes tired and may therefore overlook a security risk. Another problem is finding trustworthy security personnel in foreign countries where locals are the only candidates for the job.

With that in mind there is an opportunity of using the modern information-communication technology in conjunction with methods of artificial intelligence to mitigate or even eliminate the human shortcomings and increase the level of security while lowering the overall security costs. Our

Authors

Rok Piltaver received his B.Sc. degree in Computer Science from the University of Ljubljana, Slovenia, in 2008. He is a research assistant at the Department of Intelligent Systems of the Jozef Stefan Institute, Ljubljana, and a Ph.D. student of New media and e-science at the Jozef Stefan International Postgraduate School where he is working on his dissertation on combining accurate and understandable classifiers. His research interests are in artificial intelligence and machine learning with applications in ambient intelligence and ambient assisted living. He published two papers in international scientific journals and eight papers in international conferences and was awarded for the best innovation in Slovenia in 2009 and for the best joint project between business and academia in 2011. <rok.piltaver@ijs.si>

Erik Dovgan received his B.Sc. degree in Computer Science from the University of Ljubljana, Slovenia, in 2008. He is a research assistant at the Department of Intelligent Systems of the Jozef Stefan Institute, Ljubljana, and a Ph.D. student of New media and e-science at the Jozef Stefan International Postgraduate School where he is working on his dissertation on multiobjective optimization of vehicle control strategies. His research interests are in evolutionary algorithms, stochastic multiobjective optimization, classification algorithms, clustering and application of these techniques in energy efficiency, transportation, security systems and ambient assisted living. <erik.dovgan@ijs.si>

Matjaz Gams is Head of Department of Intelligent Systems at the Jozef Stefan Institute and professor of computer science at the University of Ljubljana and MPS, Slovenia. He received his degrees at the University of Ljubljana and MPS. He is or was teaching at 10 Faculties in Slovenia and Germany. His professional interest includes intelligent systems, artificial intelligence, cognitive science, intelligent agents, business intelligence and information society. He is member of numerous international program committees of scientific meetings, national strategic boards and institutions, editorial boards of 11 journals and is managing director of the Informatica journal. He was co-founder of various societies in Slovenia, e.g. the Engineering Academy, AI Society, Cognitive Society, and was president and/or secretary of various societies including ACM Slovenia. He is president of institute and faculty union members in Slovenia. He headed several national and international projects including the major national employment agent on the Internet first to present over 90% of all available jobs in a country. His major scientific achievement is the discovery of the principle of multiple knowledge. In 2009 his team was awarded for the best innovation in Slovenia and in 2011 for the best joint project between business and academia. <matjaz.gams@ijs.si>

> “ This paper presents an intelligent surveillance system for indoor high-security environments ”

first intelligent security system that is focused on the entry control is described in [5]. In this paper we present a prototype of an intelligent indoor-surveillance system (i.e. it works in the whole indoor area and not only at the entry control) that automatically detects security risks.

The prototype of an intelligent security system, called "Poveljnikova desna roka" (PDR, eng. commander's right hand), is specialized for surveillance of personnel, data containers, and important equipment in indoor high-security areas (e.g., an archive of classified data with several rooms). The system is focused on the internal threats; nevertheless it also detects external security threats. It detects any unusual behaviour based on user-defined rules and automatically extracted models of the usual behaviour. The artificial intelligence methods enable the PDR system to model usual and to recognize unusual behaviour. The system is capable of autonomous learning, reasoning and adaptation. The PDR system alarms the supervisor about unusual and forbidden activities, enables an overview of the monitored environment, and offers simple and effective analysis of the past events. Tagging all personnel, data containers, and important equipment is required as it enables real-time localization and integration with automatic video surveillance. The PDR system notifies the supervisor with an alarm of appropriate level and an easily comprehensible explanation in the form of natural language sentences, tagged video recordings and graphical animations. The PDR system detects intrusions of unidentified persons, forbidden actions of known and unknown persons and unusual activities of tagged entities. The concrete scenarios detected by the system include thefts, sabotages, staff negligence and insubordination, unauthorised entry, unusual employee behaviour and similar incidents.

The rest of the paper is structured as follows. Section 2 summarizes the related work. An overview of software modules and a brief description of used sensors are given in Section 3. Section 4 describes the five PDR modules, including the Expert System Module and Fuzzy Logic Module in more detail. Section 5 presents system verification while Section 6 provides conclusions.

2 Related Work

There has been a lot of research in the field of automatic surveillance based on video recordings. The research ranges from extracting low level features and modelling of the usual optical flow to methods for optimal camera positioning and evaluating of automatic video surveillance systems [8]. There are many operational implementations of such system increasing the security in public places (subway stations, airports, parking lots).

On the other hand, there has not been much research in the field of automatic surveillance systems based on real-time locating systems (RTLS), due to the novelty of sensory equipment. Nevertheless, there are already some simple commercial systems with so called room accuracy RTLS [20] that enable tracking of objects and basic alarms based on if-then rules [18]. Some of them work outdoors using GPS (e.g., for tracking vehicles [21]) while others use radio systems for indoor tracking (e.g., in hospitals and warehouses). Some systems allow video monitoring in combination with RTLS tracking [19].

Our work is novel as it uses several complex artificial intelligence methods to extract models of the usual behaviour and detect the unusual behaviour based on an indoor RTLS. In addition, our work also presents the benefits of combining video and RTLS surveillance.

3 Overview of the PDR System

This section presents a short overview of the PDR system. The first subsection presents the sensors and hardware used by the system. The second subsection introduces software modules. Subsection 3.3 describes RTLS data pre-processing and primitive routines.

3.1 Sensors and other Hardware

The PDR system hardware includes a real-time locating system (RTLS), several IP video cameras (Figure 1), a processing server, network infrastructure, and optionally one or more workstations, such as personal computers, handheld devices, and mobile phones with internet access, which are used for alerting the security personnel.

RTLS provides the PDR system with information about locations of all personnel and important objects (e.g. container with classified documents) in the monitored area. RTLS consists of sensors, tags, and a processing unit (Figure 1). The sensors detect the distance and the angle at which the tags are positioned. The processing unit uses these measurements to calculate the 3D coordinates of the tags. Commercially available RTLS use various technologies: infrared, optical, ultrasound, inertial sensors, Wi-Fi, or ultra-wideband radio. The technology determines RTLS accuracy (1 mm – 10 m), update frequency (0.1 Hz – 120 Hz), covered area (6 – 2500 m^2), size and weight of tags and sensors, various limitations (e.g., required line of sight between sensors and tags), reliability, and price (2.000 – 150.000 •) [13]. PDR uses Ubisense RTLS [15] that is based on the ultra-wide band technology and is among the more affordable RTLSs. It uses relatively small and energy efficient active tags, has an update rate of up to 9 Hz and accuracy of ±20 cm in 3D space given good conditions. It covers areas of up to 900 m^2 and does not require line of sight.

The advantages of a RTLS are that people feel more comfortable being

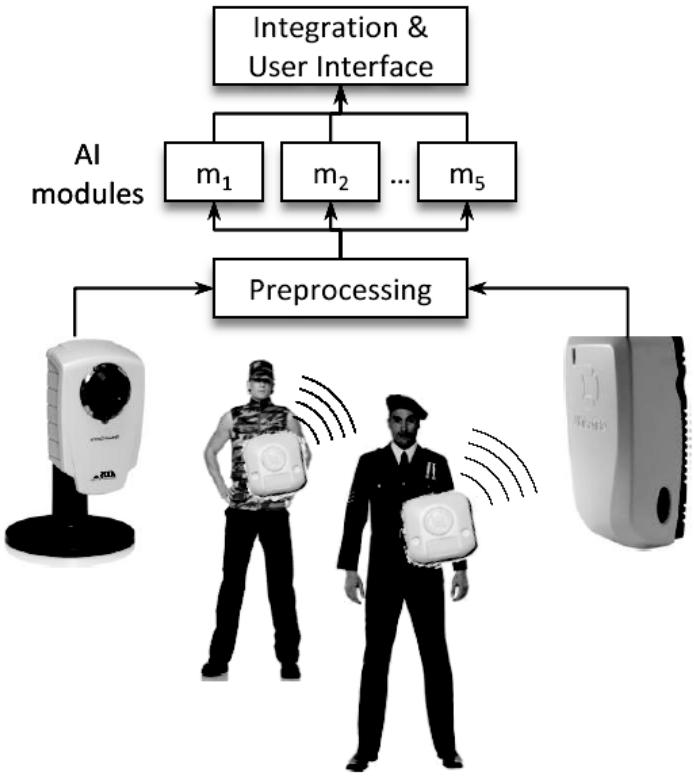

Figure 1: Overview of the PDR System.

tracked by it than being filmed by video cameras and that localization with a RTLS is simpler, more accurate, and more robust than localization from video streams. On the other hand, RTLS is not able to locate objects that are not marked with tags. Therefore, the most vital areas need to be monitored by video cameras also in order to detect intruders that do not wear RTLS tags. However, only one PDR module requires video cameras, while the other four depend on RTLS alone. Moreover, the cameras enable on-camera processing, therefore only extracted features are sent over the network.

3.2 Software Structure

The PDR software is divided into five modules. Each of them is specialized for detecting a certain kind of abnormal behaviour (i.e., a possible security risk) and uses an appropriate artificial intelligence method for detecting it. The modules reason in real time independently of each other and asynchronically trigger alarms about detected anomalies. Three of the PDR modules are able to learn automatically while the other two use predefined knowledge and knowledge entered by the supervisor. The Video Module detects persons without tags and is the only module that needs video cameras. The Expert System Module is customisable by the supervisor, who enters information about forbidden events and actions in the form of simple rules, thus enabling automatic rule checking. The three learning modules that automatically extract models of the usual behaviour for each monitored entity and compare current behaviour with it in order to detect abnormalities are Statistic, Macro and Fuzzy Logic Modules. The Statistic Module collects statistic information about entity movement such as time spent walking, sitting, lying etc. The Macro model is based on macroscopic properties such as the usual time of entry in certain room, day of the week etc. Both modules analyse relatively long time intervals while the Fuzzy Logic Module analyses short intervals. It uses fuzzy discretization to represent short actions and fuzzy logic to infer whether they are usual or not.

3.3 RTLS Data Pre-processing and Primitive Routines

Since the used RTLS has relatively low accuracy and relatively high update rate, a two-stage data filtering is used to increase the reliability and to mitigate the negative effect of the noisy location measurements. In the first stage, median filter [1] with window size 20 is used to filter sequences of *x*, *y*, and *z* coordinates of tags. Equation (1) gives the median filter equation for direction *x*. The median filter is used to correct the RTLS measurements that differ from the true locations by more than ~1.5 m and occur in up to 2.5 % of measurements. Such false measurements are relatively rear and occur only in short sequences (e.g., probability of more than 5 consecutive measurements having a high error is very low) therefore the median filter corrects these errors well.

$$\tilde{x}_n = med\{x_{n-10}, x_{n-9}, \ldots, x_{n+8}, x_{n+9}\} \quad (1)$$

The second stage uses a Kalman filter [6] that performs the following three tasks: smoothing of the RTLS measurements, estimating the velocities of tags, and predicting the missing measurements. Kalman filter state is a six dimensional vector that includes positions and velocities in each of the three dimensions. The new state is cal-

> “The system is capable of detecting a broad range of security risks with high accuracy”

> “The crucial factor of most security systems is still a human providing the intelligence to the system”

culated as a sum of the previous position (e.g. x_n) and a product between the previous velocity (e.g. $v_{x,n}$) and the time between the consecutive measurements $Ä_t$ for each direction separately. The velocities remain constant. Equation (2) gives the exact vector formula used to calculate the next state of the Kalman filter. The measurement noise covariance matrix was set based on RTLS system specification, while the process noise covariance matrix was fine-tuned experimentally.

Once the measurements are filtered, primitive routines can be applied. They are a set of basic preprocessing methods used by all the PDR modules and are robust to noise in 3D location measurements. They take short intervals of RTLS data as input and output a symbolic representation of the processed RTLS data.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \\ \hdashline v_{x,n+1} \\ v_{y,n+1} \\ v_{z,n+1} \end{bmatrix} = \left[\begin{array}{ccc:ccc} 1 & 0 & 0 & \Delta_t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta_t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta_t \\ \hdashline 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \begin{bmatrix} x_n \\ y_n \\ z_n \\ \hdashline v_{x,n} \\ v_{y,n} \\ v_{z,n} \end{bmatrix} \quad (2)$$

The first primitive routine detects in which area (e.g., a room or a user-defined area) a given tag is located, when it has entered, and when it has exited from the area. The routine takes into account the positions of walls and doors. A special method is used to handle the situations when a tag moves along the boundary between two areas that are not separated by a wall.

The second primitive routine classifies the posture of a person wearing a tag into: standing, sitting, or lying. A parameterized classifier, trained on pre-recorded and hand-labelled training data, is used to classify the sequences of tag heights into the three postures. The algorithm has three parameters: the first two are thresholds t_{lo} and t_{hi} dividing the height of a tag into the three states, while the third parameter is tolerance d. The algorithm stores the previous posture and adjusts the boundaries between the postures according to it (Figure 2). If the current state is below the threshold t_i, it is increased by d, otherwise it is decreased by d. The new posture is set to the posture that occurs most often in the window of consecutive tag heights according to the dynamically set thresholds. The thresholds t_{lo} and t_{hi} were obtained from the classification tree that classifies the posture of a person based on the height of a tag. It was trained on half an hour long manually labelled recording of lying, sitting and standing.

The third group of primitive routines is a set of routines that detect whether a tag is moving or not. This is not a trivial task due to the considerable amount of noise in the 3D location data. There are separate routines for detecting movement of persons, movable objects (e.g., a laptop) and objects that are considered stationary. The routines include hardcoded, handcrafted, common sense algorithms and a classifier trained on extensive, pre-recorded, hand labelled training set. The classifier uses the following attributes calculated in a sliding window with size 20: the average speed, the approximate distance travelled, sum of consecutive position distances, and the standard deviation of moving direction. The classifier was trained on more than two hours long hand-labelled recording of consecutive moving and standing still. Despite the noise in the RTLS measurements the classification accuracy of 95 % per single classification was achieved. [12] describes the classifier in more detail.

The final group of routines detects if two tags (or a tag and a given 3D position) are close together by comparing the short sequences of tags' positions. There are separate methods used for detecting distances between two persons (e.g., used to detect if a visitor is too far away from its host), between

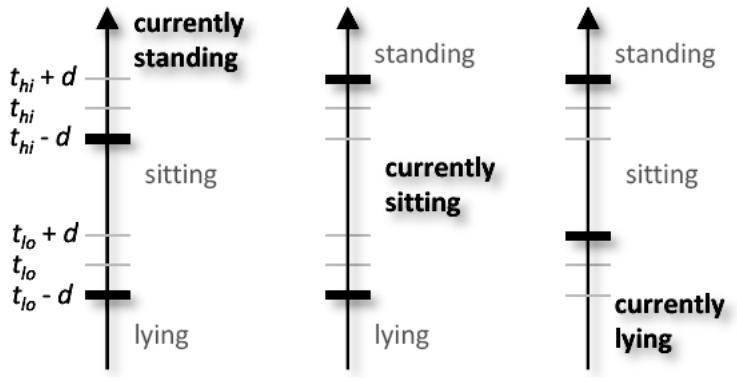

Figure 2: Dynamic Thresholds.

> “The PDR system detects intrusions of unidentified persons, forbidden actions of known and unknown persons and unusual activities of tagged entities”

a person and an object, and between a person and a given 3D location (e.g., used to assign tags of moving persons to locations of moving objects detected by video processing).

All of the described primitive routines are robust to the noise in RTLS measurements and are specialized for the PDR's RTLS. Primitive routines' parameters were tuned according to the noise of the RTLS and using data mining tools Orange [4] and Weka [16]. In case of more accurate RTLS, the primitive routines could be simpler and more accurate. Nevertheless, the presented primitive routines perform well despite the considerable amount of noise. This is possible because of the relatively high update rate. If it was significantly lower, the primitive routines would not work as well. Therefore, the accuracy, reliability and update rate of RTLS are crucial for the performance of the entire PDR system.

4 PDR Modules

4.1 Expert System Module

The Expert System Module enables the supervisor to customize the PDR system according to his/her needs by setting simple rules that must not be violated. It is the simplest and the most reliable module of the PDR system [11]. It is capable of detecting a vast majority of the predictable security risks, enables simple customization, is reliable, robust to noise, raises almost no false alarms, and offers comprehensible explanation for the raised alarms. In addition, it does not suffer from the typical problems common to the learning modules/algorithms, such as long learning curve, difficulty to learn from unlabeled data, relatively high probability of false alarms, and the elusive balance between false negative and false positive classifications. The expert system consists of three parts described in the following subsections.

4.1.1 Knowledge Base

Knowledge base of an expert system contains the currently available knowledge about the state of the world. The knowledge base of PDR expert system consists of RTLS data, predefined rules, and user-defined rules. The first type of knowledge is in form of data stream, while the latter two are in form of if-then rules.

The expert system gets the knowledge about objects' positions from the RTLS data stream. Each unit of the data stream is a filtered RTLS measurement that contains a 3D location with a time stamp and a RTLS tag ID.

User-defined rules enable simple customization of the expert system according to specific supervisor's needs by specifying prohibited and obligatory behaviour. Supervisor can add, edit, view, and delete the rules at any time using an intuitive graphic user interface. There are several rule templates available. The supervisor has to specify only the missing parameters of the rules, such as for which entities (tags), in which room(s) or user-defined areas(s), and at which time the rules apply.

For instance, a supervisor can choose to add a rule based on the following template: "Person *P* must be in the room *R* from time T_{min} to time T_{max}." and set *P* to John Smith, *R* to the hallway H, T_{min} to 7 am, and T_{max} to 11 am. Now the expert system knows that John must be in the hallway from 7 am to 11 am. If he leaves the hallway during that period or if he does not enter it before 7 am, the PDR supervisor will be notified.

Some of the most often used rule templates are listed below:

- Object O_i is not allowed to enter area A_i.
- Object O_i can only be moved by object O_j.
- Object O_i must always be close to object O_j.

The predefined rules are a set of rules that are valid in any application where PDR might be used. Nevertheless, the supervisor has an option to turn them on or off. Predefined rules define when alarms about hardware failures should be triggered.

4.1.2 Inference Engine

The inference engine is the part of the PDR expert system that deduces conclusions about security risks from the knowledge stored in the knowledge base. The inference process is done in real-time. First, the RTLS data stream is processed using the primitive routines. Second, all the rules related to a given object (e.g., a person) are checked. If a rule fires, an alarm is raised and an explanation for the raised alarm is generated. An example is presented in the next paragraph.

Suppose that the most recent 3D location of John Smith's tag (from the previous example) has just been received at 8:32 am. The inference engine checks all the rules concerning John Smith. Among them is the rule R_i that says: "John Smith must be in the hallway H from 7 am to 11 am." The inference engine calls the primitive routine that checks whether John is in the hallway H. There are two possible outcomes. In the first outcome, he is in the hallway H, therefore, the rule R_i is not violated. If John was not in the hallway H in the previous instant, there is an ongoing alarm that is now ended by the inference engine. In the second outcome, John is not in the hallway H; hence the rule R_i is violated at this moment. In this case the inference engine checks if there is an ongoing alarm about John not being in the hallway H. If there is no such ongoing alarm the inference engine triggers a new alarm. On the other hand, if there is such an alarm, the inference engine knows that the PDR supervisor was already notified about it.

> “Our work is novel as it uses several complex artificial intelligence methods to extract models of the usual behaviour and detect the unusual behaviour based on an indoor RTLS”

If an alarm was raised every time a rule was violated, the supervisors would get flooded with alarm messages. Therefore, the inference engine automatically decreases the number of alarm messages and groups alarm messages about the same incident together so that they are easier to handle by the PDR supervisor. The method will be illustrated with an example. Because of the noise in 3D location measurements the inference engine does not trigger or end an alarm immediately after the status of rule R_i (violated/not violated) changes. Instead it waits for more RTLS measurements and checks the trend in the given time window: if there are only few instances when the rule was violated they are considered as noise. On the other hand, if there are many (over the global threshold set by the supervisor) such instances, then the instances when rule was not violated are treated as noise. Two consecutive alarms that are interrupted by a short period of time will therefore result in a single alarm message. A short period in which a rule seems to be violated because of the noise in RTLS data, however, will not trigger an alarm. The grouping of alarms works in the following way: the inference engine groups the alarm messages based on the two rules R_i and R_j together if at the time when rule R_i is violated another rule R_j concerning John Smith or hallway H is violated too. As a result, the supervisor has to deal with fewer alarm messages.

4.1.3 Generating Alarm Explanations

The Expert System Module also provides the supervisor with an explanation of the alarm. It consists of three parts: explanation in natural language, graphical explanation, and video recording of the event.

Each alarm is a result of a particular rule violation. Since each rule is an instance of a certain rule template, explanations are partially prepared in advance. Each rule template has an assigned pattern in the form of a sentence in natural language with some objects and subjects missing. In order to generate the full explanation, the inference engine fills in the missing parts of the sentence with details about the objects (e.g., person names, areas, times, etc.) related to the alarm.

Graphical explanation is given in form of a ground plan animation and can be played upon supervisors' request. The inference engine determines the start and the end times of an alarm and sets the animation to begin slightly before the alarm was caused and to end slightly after the causes for the alarm are no longer present. The animation is generated from the recorded RTLS data and the ground plan of the building under surveillance. The animated objects (e.g., persons, objects, areas) that are relevant to the alarm are highlighted with red colour.

If a video recording of the incident that caused an alarm is available it is added to the alarm explanation. Based on the location of the person that caused the alarm, the person in the video recording is marked with a bounding rectangle (Figure 3). The video explanation is especially important if an alarm is caused by a person or object without a tag.

The natural language explanation, ground plan animation, and video recordings with embedded bounding rectangles produced by the PDR expert system efficiently indicate when and to which events the security personnel should pay attention.

Figure 3: Video Explanation of an Alarm.

4.2 Video Module

The video Module periodically checks if the movement detected by the video cameras is caused by people marked with tags. If it detects movement in an area where no authorised humans are located, it triggers an alarm. It combines the data about tag locations and visible movement to reason about unauthorised entry.

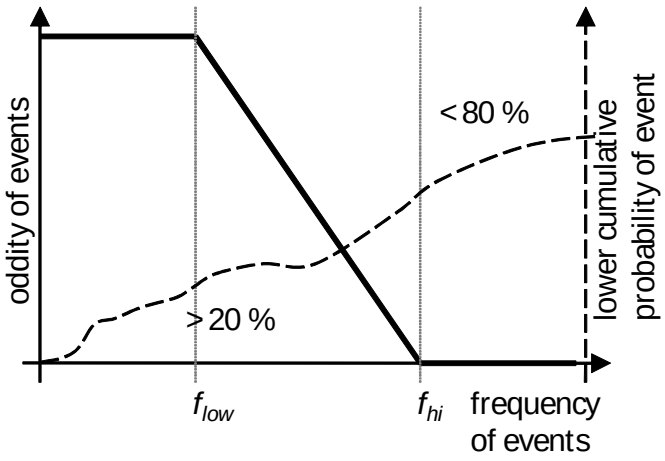

Figure 4: Calculating the Oddity of Events.

Data about visible moving objects (with or without tags) is available as the output of video pre-processing. Moving objects are described with their 3D locations in the same coordinate system as RTLS data, sizes of their bounding boxes, similarity of the moving object with a human, and a time stamp. The detailed description of the algorithm that processes the video data (developed at the Faculty of Electrical Engineering, University of Ljubljana, Slovenia) can be found in [9] and [10].

The Video Module determines the pairing between the locations of tagged personnel and the detected movement locations. If it determines that there is movement in a location that is far enough from all the tagged personnel, it raises an alarm. In this case the module reports moving of an unauthorised person or an unknown object (e.g., a robot) based on the similarity between the moving object and a person. The probability of false alarms can be reduced if several cameras are used to monitor the area from various angles. It also enables more accurate localization of moving objects.

Whenever the Video Module triggers an alarm it also offers an explanation for it in form of video recordings with embedded bounding boxes highlighting the critical areas (Figure 3). The supervisor of the PDR system can quickly determine whether the alarm is true or false by checking the supplied video recording.

The video pre-processing algorithm is also capable of detecting if a certain camera is blocked (e.g. covered with a piece of fabric). Such information is forwarded to the Video Module that triggers an alarm.

4.3 Fuzzy Logic Module

The Fuzzy Logic Module is based on the following presumption: frequent behaviour is usual and therefore uninteresting while rare behaviour is interesting as it is highly possible that it is unwanted or at least unusual. Therefore the module counts the number of actions done by the object under surveillance and reasons about oddity of the observed behaviour based on the counters. If it detects a high number of odd events (i.e., events that rarely took place in the past) in a short period of time, it triggers an alarm.

The knowledge of the module is stored in two four- dimensional arrays of counters for each object under surveillance (implemented as red-black trees [2]). Events are split into two categories, hence the two arrays: events caused by movement and stationary events. A moving event is characterised by its location, direction, and the speed of movement. A stationary event, on the other hand, is characterised by location, duration and posture (lying, sitting, or standing). When an event is characterised, fuzzy discretization [17] is used, hence the name of the module. The location of an event in the floor plane is determined using the RTLS system and discretized in classes with size 50 cm, therefore the module considers the area under surveillance as a grid of 50 by 50 cm squares. The speed of movement is estimated by the Kalman filter. It is used to calculate the direction which is discretized in the 8 classes (N, NE, E, SE, S, SW, W, and NW). The scalar velocity is discretized in the following four classes: very slow, slow, normal, and fast. The posture is determined by a primitive routine (see Section 3.3). The duration of an event is discretized in the following classes: 1, 2, 4, 8, 15, 30, seconds, minutes or hours.

The fuzzy discretization has four major advantages. The first is a smaller amount of memory needed to store the counters, as there is only one counter for a whole group of similar events. Note that the accuracy of the stored knowledge is not significantly decreased because the discrete classes are relatively small. The second advantage is the time complexity of counting the events that are similar to a given event, which is constant instead of being dependent on the number of events seen in the past. The third advantage is the linear interpolation implicitly introduced by fuzzy discretization, which enables a more accurate estimation of the rare events' frequencies. The fourth advantage is the low time complexity of updating the counters' values compared to the time complexity of adding a new counter with value 1 for each

"The advantages of a RTLS are that people feel more comfortable being tracked by it than being filmed by video cameras"

> “The PDR software is divided into five modules: Video, Expert System, Statistic, Macro and Fuzzy Logic”

new event.

The oddity of the observed behaviour is calculated using a sliding window over which the average oddity of events is calculated. Averaging the oddity over time intervals prevents the false alarms that would be triggered if the oddity of single events was used whenever RTLS data noise or short sequences of uncommon events would occur. The oddity of a single event is calculated by comparing the frequency of events similar to the given event with the frequencies of the other events. For this purpose the supervisor sets the two relative frequencies f_{low} and f_{hi}. The threshold f_{low} determines the share of the rarest events that are treated as completely unusual and therefore they get assigned the maximum level of oddity. On the other hand, f_{hi} determines the share of the most frequent events that are treated as completely usual and therefore they get assigned 0 as the level of oddity. The oddity of an event whose frequency is between the thresholds f_{low} and f_{hi} is linearly decreasing with the increasing share of the events that are rarer than the given event (Figure 4).

The drawback of the described method is a relatively long learning period which is needed before the module starts to perform well. On the other hand, the module discards the outdated knowledge and emphasizes the new data, which enables adapting to the gradual changes in observed person's behaviour. The module is also highly responsive: it takes only about 3 seconds to detect the unusual behaviour. The module autonomously learns the model of usual behaviour which enables the detection of the unusual behaviour. It can detect events such as an unconscious person lying on the floor, running in a room where people usually do not run, a person sitting at the table at which he usually does not sit etc. The module also triggers an alarm when a long sequence of events happens for the first time. If such false alarm is triggered, the supervisor can mark it as false. Consequently, the module will increase the appropriate counters and will not raise an alarm for that kind of behaviour in the future.

When the Fuzzy Logic Module triggers an alarm, it also provides a graphical explanation for it. It draws a target-like graph in each square of the mesh dividing the observed area. The colour of a sector of the target represents the frequency of a given group of similar events. The concentric circles represent the speed of movement, e.g., a small radius represents a low speed. The triangles, on the other hand, represent the direction of movement. The location of a target on the mesh represents the location in the physical area. White colour depicts the lowest frequency, black colour depicts the highest frequency while the shades of grey depict the frequencies in between. The events that caused an alarm are highlighted with a scale ranging from green to red. For stationary events, tables are used instead of the targets. The row of the table represents the posture while the column represents the duration. A supervisor can read the graphical explanations quickly and effectively. The visualization is also used for the general analysis of the behaviour in the observed area.

4.4 Macro and Statistic Modules

Macro and Statistic modules analyse persons' behaviour and trigger alarms if it significantly deviates from the usual behaviour. In order to do that, several statistics about the movement of each tagged person are collected, calculated, and averaged over various time periods. Afterwards, these statistics are compared to the previously stored statistics of the same person and the deviation factor is calculated. If it exceeds the predefined bound, the modules trigger an alarm.

The Statistic Module collects data over time periods from one minute to several hours regardless of person's location or context. On the other hand, the Macro Module collects data regarding behaviour in certain areas (e.g. room), i.e. the behaviour collection starts when a person enters the area and ends when he/she leaves it.

Both modules use behaviour attributes such as: the percentage of the time the person spent lying, sitting, standing, or walking during the observed time period, the average walking speed. Additionally, Macro module uses the following attributes: area id, day of the week, length of stay, entrance time, and exit time.

The behaviours are classified with the LOF algorithm [3], a density-based kNN algorithm, which calculates the local outlier factor of the tested instance with respect to the learning instances. The LOF algorithm was chosen based on the study [14]. Bias towards false positives or false negatives can be adjusted by setting the alarm threshold.

The modules show a graphical explanation for each alarm in form of parallel coordinates plot. Each attribute is represented with one of the parallel vertical axes, while statistics about given time periods are represented by a zigzag line connecting values of each attribute from the leftmost to the rightmost one. Past behaviour is represented with green zigzag lines, while the zigzag line portending to the behaviour that triggered the alarm is collared red. The visualisation offers a quick and simple way of establishing the cause of alarm and often indicates more specific reason for it.

5 Verification

Due to the complexity of the PDR system and the diverse tasks that it performs it is difficult to verify its quality with a single test or to summarize it in a single number such as true positive rate. Therefore, validation was done on

Module	TP	TN	FP	FN	N
Expert Sys.	197	199	2	2	400
Video	30	30	0	0	60
Fuzzy Logic	47	42	8	3	100
Macro	9	10	1	0	20
Statistic	9	10	1	0	20
Total	292	291	12	5	600
Percentage (%)	**48.7**	**48.5**	**2**	**0.8**	

Table 1: Evaluation of PDR System.

more subjective and qualitative level with several scenarios for each of the individual modules. Four demonstration videos of the PDR tests are available at http://www.youtube.com/user/ijsdis. A single test case or a scenario is a sequence of actions and events including a security risk that should be detected by the system. "A person enters a room without the permission" is an example of scenario. Each scenario has a complement pair: a similar sequence of actions which, on the contrary, must not trigger an alarm. "A person with permission enters the room" is the complement scenario for the above example. The scenarios and their complements were carefully defined in cooperation and under supervision of security experts from the Slovenian Ministry of Defence .

The Expert System Module was tested with two to three scenarios per expert rule template. Each scenario was performed ten times with various persons and objects. The module has perfect accuracy (no false positives and no false negatives) in cases when the RTLS noise was within the normal limits. When the noise was extremely large, the system occasionally triggered false alarms or overlooked security risks. However, in those cases even human experts were not able to tell if the observed behaviour should trigger an alarm or not based on the noisy RTLS measurements alone. Furthermore, the extreme RTLS noise occurred in less than 2 % of the scenario repetitions and the system made an error in less than 50 % of those cases.

The Video Module was tested using the following three scenarios: "a person enters the area under surveillance without the RTLS tag", "a robot is moving without authorised person's presence", and "a security camera is intentionally obscured". Scenarios were repeated ten times with different people as actors. The module detected the security risks in all of the scenario repetitions with movement and distinguished between a human and a robot perfectly. It failed to detect the obscured camera in one out of 10 repetitions. The module also did not trigger any false alarms.

The Fuzzy Logic Module was tested with several scenarios while the fuzzy knowledge was gathered over two weeks. The module successfully detected a person lying on the floor, sitting on colleagues chair for a while, running in a room, walking on a table, crawling under a table, squeezing behind a wardrobe, standing on the same spot for extended period of time, and similar unusual events. However, the experts' opinion was that some of the alarms should not have been triggered. Indeed we expect that in more extensive tests the modules supervised learning capabilities would prevent further repetitions of unnecessary alarms.

The test of the Macro and Statistic Modules included the simulation of a usual day at work condensed into one hour. The statistic time periods were 2 minutes long. Since the modules require a collection of persons' past behaviour, two usual days of work were recorded by a person constituting of two hours of past behaviour data. Afterwards, the following activities were performed 10 times by the same person and classified: performing a normal day of work, stealing a container with classified data, acting agitated as under the effect of drugs and running. The classification accuracy was 90 %. This was due to the low amount of past behaviour data. Therefore, the modules did not learn the usual behaviour of the test person but only a condensed (simulated) behaviour in a limited learning time. We expect that the classification accuracy would be even higher, if the learning time was extended and if the person would act as usual instead of simulating the condensed day of work.

The overall system performance was tested on a single scenario: "stealing a container with classified documents". In the test five persons tried to steal the container from a cabinet in a small room under surveillance. Each person tried to steal the container five times with and without a tag. All the attempts were successfully detected by the system that reported the alarm and provided an explanation for it.

The validation test data is summarized in Table 1. It gives the number of true positive (TP), true negative (TN), false positive (FP), and false negative alarms (FN), and total number (N) of scenario repetitions. Each row gives the results for one of the five modules. The bottom two rows give the total sum for each column and the relative percentage.

> "The system is customizable and can be used in a range of security applications such as confidential data archives and banks"

The system received the award for the best innovation among research groups in Slovenia for 2009 at the Fourth Slovenian Forum of Innovations.

6 Conclusion

This paper presents an intelligent surveillance system utilizing a real-time location system (RTLS), video cameras, and artificial intelligence methods. It is designed for surveillance of high security indoor environments and is focused on internal security threats. The data about movement of personnel and important equipment is gathered by RTLS and video cameras. After basic pre-processing with filters and primitive routines the data is sent to the five independent software modules. Each of them is specialized for detecting specific security risk. The Expert System Module detects suspicious situations that can be described by location of a person or other tagged objects in space and time. It detects many different scenarios with high accuracy. The Video Module automatically detects movement of persons and objects without tags, which is not allowed inside the surveillance area. Fuzzy Logic, Macro, and Statistics Modules automatically extract the usual movement patterns of personnel and equipment and detect deviations from the usual behaviour. Fuzzy Logic is focused on short-term anomalous behaviour such as entering an area for the first time, lying on the ground or walking on the table. Macro and Statistic Modules, on the other hand, are focused on mid- and long-term behaviour such as deviations in daily work routine.

The validation of the system shows that it is able to detect all the security scenarios it was designed for and that it does not raise too many false alarms even in more challenging situations. In addition, the system is customizable and can be used in a range of security applications such as confidential data archives and banks.

Acknowledgement

Research presented in this paper was financed by the Republic of Slovenia, Ministry of Defence. We would like to thank the colleges from the Machine Vision Laboratory, Faculty of Electrical Engineering, University of Ljubljana, Slovenia and Spica International, d.o.o. for fruitful cooperation on the project. Thanks also to Boštjan Kaluza, Mitja Lustrek, and Bogdan Pogorelc for help regarding the RTLS and discussions, and Anze Rode for discussions about security systems, expert system rules templates and specification of scenarios.

References

[1] G. R. Arce. "Nonlinear Signal Processing: A Statistical Approach", Wiley: New Jersey, USA, 2005.

[2] R. Bayer. "Symmetric Binary B-Trees: Data Structures and Maintenance Algorithms", Acta Informática, 1, pp. 290–306, 1972.

[3] M. M. Breunig, H. P. Kriegel, R. T. Ng, J. Sander. "LOF: Identifying densitybased local outliers," Proceedings of the International Conference on Management of Data – SIGMOD '00, pp. 93–104, Dallas, Texas, 2000.

[4] J. Demsar, B. Zupan, G. Leban. "Orange: From Experimental Machine Learning to Interactive Data Mining," White Paper (www. ailab.si/orange), Faculty of computer and information science, University of Ljubljana, Slovenia, 2004.

[5] M. Gams, T. Tusar. (2007), "Intelligent High-Security Access Control", Informatica, vol 31(4), pp. 469-477.

[6] R.E. Kalman. "A new approach to linear filtering and prediction problems". Journal of Basic Engineering, 82 (1), pp. 35–45, 1960.

[7] M. Kolbe, M. Gams. "Towards an intelligent biometric system for access control," Proceedings of the 9th International Multiconference Information Society - IS 2006, Ljubljana, Slovenia, 2006, pp. 118-122.

[8] B. Krausz, R. Herpers. 'Event detection for video surveillance using an expert system', Proceedings of the 1st ACM Workshop on Analysis and Retrieval of Events/Actions and Workflows in Video Streams - AREA 2008, Vancouver, Canada, pp. 49-56.

[9] M. Kristan, J. Pers, M. Perse, S. Kovaèiè. "Closed-world tracking of multiple interacting targets for indoor-sports applications", Computer Vision and Image Understanding, vol 113, 5, pp. 598-611, 2009.

[10] M. Perše, M. Kristan, S. Kovaèiè, G. Vuèkoviæ, J. Pers. "A trajectory-based analysis of coordinated team activity in a basketball game", Computer Vision and Image Understanding, vol 113, 5, pp. 612-621, 2009.

[11] R. Piltaver, G. Matjas. "Expert system as a part of intelligent surveillance system", Proceedings of the 18th International Electrotechnical and Computer Science Conference - ERK 2009, vol. B, pp. 191–194, 2009.

[12] R. Piltaver. "Strojno uèenje pri naèrtovanju algoritmov za razpoznavanje tipov gibanja", Proceedings of the 11th International Multiconference Information Society - IS 2008, str. 13–17, 2008.

[13] V. Schwarz, A. Huber, M. Tüchler. "Accuracy of a Commercial UWB 3D Location Tracking System and its Impact on LT Application Scenarios," Proceedings of the IEEE International Conference on Ultra-Wideband, Zürich, Switzerland, 2005.

[14] T. Tusar, M. Gams. "Odkrivanje izjem na primeru inteligentnega sistema za kontrolo pristopa," Proceedings of the 9th International Multiconference Information Society - IS 2006, Ljubljana, Slovenia, 2006, pp. 136-139.

[15] Ubisense: available at: http://www.ubisense.net/

[16] H. Witten, E. Frank. Data Mining. "Practical Machine Learning Tools and Techniques" (2nd edition), Morgan Kaufmann, 2005.

[17] L. A. Zadeh. "Fuzzy sets", Information and Control 8 (3), pp. 338–353, 1965.

[18] http://www.pervcomconsulting.com/secure.html

[19] http://www.visonictech.com/Active-RFID-RTLS-Tracking-and-Mangement-Software-Eiris.html

[20] http://www.aeroscout.com/content/healthcare

[21] http://www.telargo.com/solutions/track_trace.asp

