

2010 IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE

2010 July 18-23, Barcelona, Spain



WCCI 2010

[Preface](#)

[Program at a glance](#)

[Organizing Committee](#)

[Welcome Message](#)

[Plenary Sessions](#)

[Announcements](#)

[Detailed Program](#)

[Detailed Program](#)

[Detailed Program](#)

[Invited Sessions](#)

[Invited Sessions](#)

[Invited Sessions](#)

[Author Index](#)

[Paper Index](#)

[Author Index](#)

[Paper Index](#)

[Author Index](#)

[Paper Index](#)

[Program Committee](#)

[Program Committee](#)

[Program Committee](#)

[Hybrid Program](#)

IJCNN 2010

IEEE Catalog Number: CFP10US-DVD

ISBN: 978-1-4244-6917-8

FUZZ-IEEE 2010

IEEE Catalog Number: CFP10FUZ-DVD

ISBN: 978-1-4244-6920-8

IEEE CEC 2010

IEEE Catalog Number: CFP10ICE-DVD

ISBN: 978-1-4244-6910-9

© 2010 IEEE



INTERNATIONAL NEURAL NETWORK SOCIETY



Human Activity Recognition applying Computational Intelligence techniques for fusing information related to WiFi positioning and body posture

A. Alvarez-Alvarez, J. M. Alonso, G. Trivino, N. Hernández, F. Herranz, A. Llamazares and M. Ocaña

Abstract—This work presents a general framework for people indoor activity recognition. Firstly, a Wireless Fidelity (WiFi) localization system implemented as a Fuzzy Rule-based Classifier (FRBC) is used to obtain an approximate position at the level of discrete zones (office, corridor, meeting room, etc). Secondly, a Fuzzy Finite State Machine (FFSM) is used for human body posture recognition (seated, standing upright or walking). Finally, another FFSM combines both WiFi localization and posture recognition to obtain a robust, reliable, and easily understandable activity recognition system (working in the desk room, crossing the corridor, having a meeting, etc). Each user carries with a personal digital agenda (PDA) or smart-phone equipped with a WiFi interface for localization task and accelerometers for posture recognition. Our approach does not require adding new hardware to the experimental environment. It relies on the WiFi access points (APs) widely available in most public and private buildings. We include a practical experimentation where good results were achieved.

I. INTRODUCTION

People activity recognition provides interesting applications in many areas, e.g., to filter the phone calls depending on different circumstances, personal navigation assistance, personal security, etc. We are mainly interested in indoor security applications (for instance sending warnings when someone gets into a dangerous area in order to reduce the occupational health and safety risk) and/or people assistance (for instance helping elderly or handicapped people).

Our activity recognition system is mainly based on Fuzzy Logic (FL) [1] because it allows to combine several heterogeneous sources of knowledge (mainly expert knowledge and knowledge automatically extracted from experimental data provided by sensors), dealing with vague information, and its interaction with humans demands the design of an easily understandable system. FL is widely recognized for its ability for linguistic concept modeling and its use in system identification. On the one hand, FL semantic expressivity, using linguistic variables [2] and linguistic rules [3], is quite close to natural language what reduces the effort of expert knowledge extraction. On the other hand, being universal approximators [4] fuzzy inference systems (FIS) are able to perform nonlinear mappings between inputs and outputs. Thus, there are lots of fuzzy machine learning methods for knowledge induction from experimental data [5].

A. Alvarez-Alvarez, J. M. Alonso and G. Trivino are with the European Centre for Soft Computing (ECSC), Mieres, Asturias, Spain (email: {alberto.alvarez,jose.alonso,gracian.trivino}@softcomputing.es).

N. Hernández, F. Herranz, A. Llamazares and M. Ocaña are with the Department of Electronics, University of Alcalá (UAH), Madrid, Spain (email: {nhernandez,fherranz,allamazares,mocana}@depeca.uah.es).

There are other recent works [6], [7] that show the advantages of using FL for modeling and monitoring human activity. They are mainly based on fusing video sensors what means installing additional hardware (HW) like video cameras in the environment under study. On the contrary, our approach takes profit from pre-existent HW and avoids adding new devices to the environment.

In indoor environments, the use of the network infrastructure to estimate user's location is quite common. Local network based systems are sometimes based on pre-existing networks like ZigBee networks designed for home control applications [8]. However, the most used systems are based on WiFi networks which are able to provide indoor absolute localization. In contrast, the main drawback is the need of a complete network infrastructure in the whole building where we want to localize a person. Luckily, this technology is quickly growing of coverage. Currently, there are WiFi Access Points (APs) in most public buildings like hospitals, libraries, universities, museums, etc. Moreover, measuring the WiFi signal level is free even for private WiFi networks. As a result, WiFi technology is a good choice for indoor global localization systems yielding a good accuracy-cost trade-off [9].

Regarding human activity recognition it is important to know the place where a person is located but it is not enough. We propose taking into account also information related to the human body posture. It can be estimated by means of sensor based systems which provide absolute information (e.g., magnetic compass, ultrasonic or infrared sensors) or relative information (e.g., inertial measurement units or pressure sensors). One low-cost inertial sensor is the accelerometer, based on the Micro Electro Mechanical Systems (MEMS) technology that has allowed its integration in small and low energy consumption devices. Accelerometers can be used as step length estimators; furthermore they let us to obtain some information about body posture [10]. In previous works we have already shown how human activity can be analyzed in terms of combining one accelerometer with a skin conductivity meter [11]. This work focuses on exploiting the fusion of WiFi signal and accelerations.

This paper is organized as follows. Section II describes how to design a Fuzzy Rule-based Classifier (FRBC) while Section III formalizes the notion of the Fuzzy Finite State Machine (FFSM). Afterwards, Section IV explains our proposal related to people activity recognition. It combines one FRBC and two FFSMs. Then, Section V shows the

experimental results. And finally, Section VI expounds the conclusions and future works.

II. FUZZY RULE BASE CLASSIFIERS

A FRBC is a fuzzy system able to select one output class from a pre-defined set of classes $C = \{C^1, C^2, \dots, C^{NC}\}$. Given an n -dimensional input space ($X \subseteq R^n$), a fuzzy inference mechanism yields an activation degree associated to each class C^i . Of course, several classes can be activated at the same time with activation degree greater than zero.

Our FRBC is designed following the fuzzy modeling methodology called HILK (Highly Interpretable Linguistic Knowledge) [12]. It focuses on building comprehensible fuzzy classifiers, i.e., classifiers easily understandable by human beings. Useful pieces of knowledge are automatically extracted from experimental data and represented by means of linguistic variables and rules under the FL formalism. The whole modeling process is made up of three steps:

- **Partition design.** The readability of fuzzy partitioning is a prerequisite to build interpretable FRBCs. Therefore, it is based on the use of Strong Fuzzy Partitions (SFPs) which are the best ones from the comprehensibility point of view.
- **Rule base learning.** Linguistic rules are automatically extracted from data keeping in mind the comprehensibility goal. Therefore, we have chosen Fuzzy Decision Tree (FDT) [13] as rule induction method. It generates a neuro-fuzzy decision tree which is translated into quite general incomplete rules where only a subset of input variables is considered. In addition, inputs are sorted according to their importance (minimizing the entropy). FDT is a fuzzy version of the popular decision trees defined by Quinlan [14]. Rules are of form **If Premise Then Conclusion**, where both *Premise* and *Conclusion* use linguistic terms previously defined for expressing linguistic propositions that describes the system behavior.

$$R: \text{If } \underbrace{I_1 \text{ is } A_1^i}_{\text{Premise } P_1} \text{ AND } \dots \text{ AND } \underbrace{I_{NI} \text{ is } A_{NI}^j}_{\text{Premise } P_{NI}} \text{ Then } \underbrace{Y_R \text{ is } C^i}_{\text{Conclusion}}$$

Premise

where given a rule R , rule premises are made up of tuples (*input variable, linguistic term*) where I_a is the name of the input variable a , while A_a^i represents the label i of such variable, with a belonging to $\{1, \dots, NI\}$ and being NI the number of inputs. In the conclusion part, C^i represents one of the possible output classes.

- **Knowledge base improvement.** It is an iterative refinement process that comprises both rule base simplification and fuzzy partition optimization. The former increases interpretability while keeping high accuracy. The later gets higher accuracy without penalizing the high interpretability previously achieved.

Designed FRBCs are endowed with the usual fuzzy classification structure based on the Max-Min inference scheme, and the winner rule fuzzy reasoning mechanism:

$$y_{FRBC}(x^p) = C^i \Leftrightarrow \mu_{C^i}(x^p) = \max_{k=1, \dots, NC} \mu_{C^k}(x^p)$$

$$\mu_{C^k}(x^p) = \max_{R=1, \dots, NR} \mu_R(x^p) \Leftrightarrow Y_R \text{ is } C^k$$

$$\mu_R(x^p) = \min_{i=1, \dots, NI} \mu_{A_i^j}(x_i^p)$$

where given an input vector $x^p = \{x_1^p, \dots, x_{NI}^p\}$, the output class C^i is derived from the highest $\mu_{C^i}(x^p)$ which is the membership degree of x^p to the class C^i . It is computed as the maximum firing degree of all rules yielding C^i as output class. For each rule, the firing degree is computed as the minimum membership degree of x^p to all the attached A_i^j fuzzy set, for all the NI inputs.

III. FUZZY FINITE STATE MACHINES

In previous studies, we have showed that FFSM are suitable tools for modeling phenomena that follow an approximately repetitive pattern [15], [16], [17], [18]. During the development of these works, the concept of FFSM has grown up in clarity and usability. In the following, we will introduce the current version of this paradigm for system modeling.

A FFSM is a tuple $\{Q, S, S_0, U, Y, f, g\}$. We will describe each one of its components in the next subsections.

A. Fuzzy States

Every state represents the pattern of a repetitive situation. The fuzzy state of the system (Q) is a linguistic variable [2] that takes its values in the set of linguistic labels $\{Q_1, Q_2, \dots, Q_n\}$, where n is the number of states. Numerically, the fuzzy state of the FFSM is represented with a state activation vector:

$$S[t] = (s_1[t], s_2[t], \dots, s_n[t]), \text{ where } s_i \in [0, 1].$$

Moreover, the FFSM implementation verifies $\sum_{i=1}^n s_i = 1$.

We require the state activation vector to fulfill the previous relation for two reasons; first, we want that the degree of activation works like a quantity that is distributed among the different states keeping the total degree as a constant equal to one; second, the state activation vector may be used as input of a second FFSM (serial connection), so we want that these input values are normalized in the interval $[0, 1]$ in such a way that we do not need to renormalize them.

We define, S_0 as the initial value of the state activation vector at $t = 0$, i.e., $S_0 = S[t = 0]$.

B. Input variables

U is the input vector $(u_1, u_2, \dots, u_{NI})$. Typically, U is a set of linguistic variables obtained after fuzzification of numerical measures obtained from sensors. Moreover, u_i can be directly obtained from sensor data and also applying some calculations, e.g., the derivative or integral of the signal, or by combination of several signals.

The designer summarizes the domain of the possible numerical values provided by sensors representing them by a small set of fuzzy intervals.

$A_{u_i} = \{A_{u_i}^1, A_{u_i}^2, \dots, A_{u_i}^{n_a}\}$ is the set of all the possible values that u_i can take, being n_a the number of linguistic labels of the linguistic variable u_i .

C. Transition function f

The next value of the state activation vector is obtained by means of the transition function f :

$$S[t+1] = f(U[t], S[t]).$$

This function is implemented by means of a set of expert fuzzy rules. Once the designer has identified the relevant states in the model, he/she must define the rules that govern the temporal evolution of the system among these states (e.g., see Figure 2).

We can distinguish between rules R_{ii} to remain in a state Q_i , and rules R_{ij} to change from the state Q_i to the state Q_j . To design the allowed transitions and the forbidden ones, we follow a simple procedure: the allowed transitions have explicitly associated fuzzy rules while there are not rules associated with the forbidden transitions.

1) *Rules to remain in a state:* The designer uses these rules to express the conditions of the system to remain in a specific state. The generic expression of R_{ii} is formulated as follows:

$$R_{ii}: \text{If } S[t] \text{ is } Q_i \text{ AND } C_{ii} \text{ Then } S[t+1] \text{ is } Q_i$$

where:

- The antecedent ($S[t]$ is Q_i) calculates the degree of activation of the state Q_i in the instant of time t , i.e., $s_i(t)$. Note that the FFSM cannot remain in the state Q_i if it is not in this state previously.
- The antecedent C_{ii} describes the constraints over the input variables to remain in the state Q_i . For example: $C_{ii} = (u_1 \text{ is } A_{u_1}^3) \text{ AND } (u_2 \text{ is } A_{u_2}^1) \text{ OR } (u_3 \text{ is } A_{u_3}^5)$.
- Finally, the consequent of the rule is the next value of the state activation vector $S[t+1]$. It consists of a vector with a zero in all of its components except in s_i , where it has a one.

2) *Rules to change of state:* The designer uses these rules to express the conditions that make the system change from state Q_i to state Q_j . Here, the generic expression of R_{ij} is formulated as follows:

$$R_{ij}: \text{If } S[t] \text{ is } Q_i \text{ AND } C_{ij} \text{ Then } S[t+1] \text{ is } Q_j$$

where:

- The antecedent ($S[t]$ is Q_i) calculates the degree of activation of the state Q_i in the instant of time t , i.e., $s_i(t)$. Note that the FFSM cannot change from the state Q_i to the state Q_j if it is not in Q_i previously.
- The antecedent C_{ij} describes the constraints over the input variables to change from the state Q_i to the state Q_j . In a first approach, these conditions could coincide with the amplitude conditions to remain in the destination state of the transition, i.e., $C_{ij} = C_{jj}$.

Then, some tuning could be needed to express different conditions to change.

- Finally, the consequent of the rule is the next value of the state activation vector $S[t+1]$. It consists of a vector with a zero in all of its components except in s_j , where it has a one.

D. Output variables

Y is the output vector $(y_1, y_2, \dots, y_{n_y})$, where n_y is the number of output variables. Y is a summary of the characteristics of the system evolution that are relevant for the application, e.g., each y_i can be obtained as the average of certain parameters of the system while the model remained in the state Q_i .

E. Output function g

The output function $g(U[t], S[t])$ calculates the value of the output vector $Y(t)$. E.g., a possible implementation of g is doing $Y[t] = S[t] = (s_1[t], s_2[t], \dots, s_n[t])$. Here, the output is the current fuzzy state of the system represented by the state activation vector.

F. Computational implementation

In order to implement the transition function, we use the Takagi-Sugeno-Kang (TSK) approach [19]. The advantage of using TSK is that it provides directly the numerical values of $S[t]$.

Using the TSK implementation, the transition function (f) of the FFSM is formulated as follows:

$$R_{ii}^1: \text{If } S[t] \text{ is } Q_i \text{ AND } C_{ii}^1 \\ \text{Then } S[t+1]^1 = (0, \dots, s_i = 1, \dots, 0)$$

...

$$R_{ij}^r: \text{If } S[t] \text{ is } Q_i \text{ AND } C_{ij}^r \\ \text{Then } S[t+1]^r = (0, \dots, s_j = 1, \dots, 0)$$

The state activation vector ($S[t+1]$) will be the weighted average of the individual outputs:

$$S[t+1] = \begin{cases} \frac{\sum_{k=1}^r \omega_k \cdot S[t+1]^k}{\sum_{k=1}^r \omega_k} & \text{if } \sum_{k=1}^r \omega_k \neq 0 \\ S[t] & \text{if } \sum_{k=1}^r \omega_k = 0 \end{cases}$$

where ω_k is the degree of firing of the rule k using the minimum for the AND operator. This formulation keeps the system in its previous state when no rule is fired. Moreover, it makes $s_i \in [0, 1]$ and $\sum_{i=1}^n s_i = 1$.

IV. PROPOSAL

This section introduces the proposed fusion framework for human activity recognition. It is made up of three main modules as illustrated in Figure 1. Each block will be described in the following subsections. First, subsection IV-A focuses on building a FRBC devoted to estimate the location of a person

in an indoor environment by means of processing WiFi strength signal levels (SLi). Then, subsection IV-B describes the FFSM1 in charge of human body posture recognition. Finally, subsection IV-C gives the details related to the FFSM2 that combines both WiFi positioning and posture recognition yielding the desired human activity recognition.

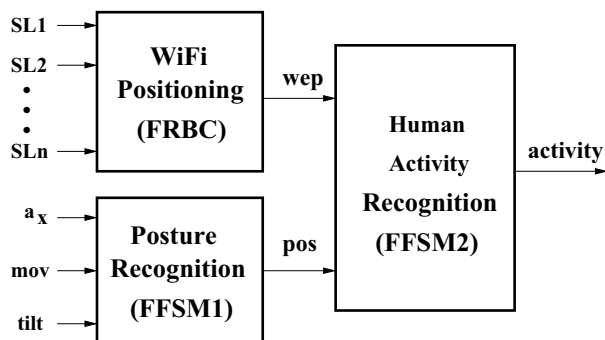


Fig. 1. Scheme of the proposed fusion framework

A. WiFi positioning module (FRBC)

WiFi localization systems use 802.11b/g network infrastructure to estimate a device position without using additional hardware. The received SL from each AP depends on the distance but also on the obstacles between the emitter and the receiver. Therefore, the simplest method for estimating the device position consists of applying a triangulation algorithm. Unfortunately, in indoor environments SL is strongly affected by the well-known multipath effect that comprises reflection, refraction and diffraction. Thus, SL becomes a complex function of the distance that dynamically changes with time because it is affected by every modification made in the environment [20].

Only approximate solutions are able to get nice results. Authors of [21] propose the use of a priori radio map storing the received SL of each AP belonging to an interest region. The radio map is built during the training stage. Then, in the estimation stage, a vector with received SL of each AP is created and compared with the radio map to obtain the estimated position. We have previously proposed the use of fuzzy classification for WiFi localization inspired on the radio map method, handling the signal measure uncertainty and getting small localization errors [9]. In this contribution we propose the use of an enhanced version of such WiFi localization system, yielding room-level localization. Notice that, the output of the FRBC will be one zone of the environment along with an activation degree which is understood as a degree of confidence on the system output. Of course, the interpolation ability of fuzzy systems makes possible to define a hierarchical localization system where the position may be refined as much as desired. In a first level it is possible to identify the floor of the building, in a second level it points out the room where the person is located, but in a third level (depending on the application) it

may be interesting giving also the position inside the room. Thus, thanks to this approach, a FRBC made up of a small number of rules is used for each level, keeping a good trade-off between accuracy and interpretability. Although we have measured the SL in many points of each room, we will only consider the second level of this hierarchy, i.e., we work at the room level with all rooms located at the same floor of the building.

As an illustrative example, let us suppose that two zones A and B are one close to the other (with a common wall) and one person is inside zone A but near the wall. The FRBC is made up of rules like **If** *Signal received from AP_i is High AND Signal received from AP_j is Low* **Then** *The person is close to Position P which belongs to zone Z*. In this example, at least two rules may be fired yielding as output an activation degree of 0.7 related to zone A and 0.3 regarding zone B. Output is computed as the result of a fuzzy inference that takes into account all defined variables and rules.

First of all, we need to identify the zones of interest in a map of the environment under analysis. The number of zones determines the number of classes of the FRBC. Second, we have to find out the APs visible in such environment. The number of APs determines the number of input variables of the FRBC. Then, in the training stage we build the radio map of the environment. To do so, we collect a training data set (LRN) with the SL measures (from all visible APs) carried out in several locations for each of the zones of interest. Then, HILK methodology [12] is applied (as it was introduced in section II) on LRN in order to automatically generate a FRBC with a good accuracy-interpretability trade-off. All input variables (one per each AP visible in the environment) are characterized by strong fuzzy partitions of nine linguistic terms (*extremely low, very low, low, etc*). In addition, linguistic rules are automatically generated from data by means of the algorithm FDT. Finally, the simplification procedure provided by HILK is run getting a more compact and general FRBC, keeping high accuracy while increasing even more its interpretability. Notice that, the FRBC follows the usual fuzzy classification structure and the winner rule fuzzy reasoning mechanism. For further information the interested reader is referred to the cited papers.

Thanks to its flexibility and adaptability the designed FRBC can be used whenever the environment does not suffer a great modification, i.e., when some access points are switched off. In such case, the system should be re-adjusted, but usually these things do not happen and the fuzzy system is able to deal with slight modifications like people moving in the environment or changes in the state of the doors.

B. Posture recognition module (FFSM1)

In previous works, we have shown how a FFSM is able to synchronize with the acceleration signal produced during the human gait and to extract the relevant characteristics suitable for our purpose [16]. In the following, we explain how to design a FFSM for body posture recognition:

1) *Fuzzy States*: Here, the fuzzy states are defined to recognize different body postures and human activity. So, we have identified three fuzzy states: $\{Q_1$: Seated, Q_2 : Upright, Q_3 : Walking $\}$.

2) *Input variables*: The set of linguistic variables U , as stated in the definition of the FFSM, can be directly obtained from sensors. In this case, we have used a three-axial accelerometer tight with a belt in the middle of the back, therefore, the numerical values that we obtain from the sensor are: the dorso-ventral acceleration (a_x), the medio-lateral acceleration (a_y) and the antero-posterior acceleration (a_z). With these numerical values, and in order to distinguish between the three different states, we have created three linguistic variables $\{a_x, mov, tilt\}$:

- a_x is the dorso-ventral acceleration as it was obtained from the sensor.
- mov measures the movement, it is the sum of the difference between the maximum and minimum of a_x , a_y and a_z respectively contained in a interval of 1 second.
- $tilt$ is a variable that measures the tilt of the body, it is calculated as the sum of the absolute value of the medio-lateral acceleration (a_y) and the absolute value of the antero-posterior acceleration (a_z), i.e., $|a_y| + |a_z|$.

The linguistic labels, that summarize the domain of each linguistic variable, are uniform strong fuzzy partitions based on trapezoidal or triangular membership functions in order to achieve a good interpretability, satisfying semantic constraints on membership functions in order to respect semantic integrity within the partitions. They are defined for each linguistic variable in the intervals defined by their maximum and minimum values taken by their numeric values, i.e., they are adapted for each user in an off-line process. The possible values of the three linguistic variables are summarized as follows:

- $a_x = \{L_{a_x}, H_{a_x}\}$ which corresponds to the terms *Low* and *High* respectively.
- $mov = \{L_{mov}, M_{mov}, H_{mov}\}$ which corresponds to the terms *Low*, *Medium* and *High* respectively.
- $tilt = \{L_{tilt}, H_{tilt}\}$ which corresponds to the terms *Low* and *High* respectively.

The input vector (U), with the set of its possible values, represents the system input with lower granularity than the domain of numerical values directly obtained from sensors.

3) *Transition function f* : As we have stated previously, we will obtain the next value of the activation vector using the transition function: $S[t + 1] = f(U[t], S[t])$.

Figure 2 shows how we use the FFSM to define constraints on the possibilities to change of state. More specifically, we force the model to pass by the state Upright (Q_2) when the subject passes from Seated (Q_1) to Walking (Q_3). The subject cannot be seated and start walking, first he/she must get upright. This restriction makes the system more robust.

- *Rules to remain in a state*. Using the generic expression of R_{ii} explained in section III-C.1, we can define the

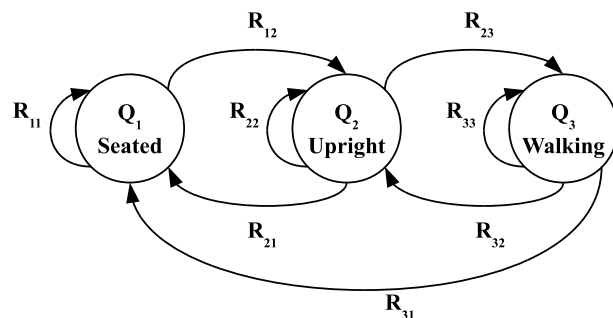


Fig. 2. Diagram of states of the FFSM1 (body posture recognition)

three constraints over the input variables to remain in each state:

$$C_{11} = a_x \text{ is } L_{a_x} \text{ AND } mov \text{ is } L_{mov} \text{ AND } tilt \text{ is } H_{tilt}$$

$$C_{22} = a_x \text{ is } H_{a_x} \text{ AND } mov \text{ is } L_{mov} \text{ AND } tilt \text{ is } L_{tilt}$$

$$C_{33} = a_x \text{ is } H_{a_x} \text{ AND } mov \text{ is } (M_{mov} \text{ OR } H_{mov})$$

- *Rules to change of state*. In a first approach, the constraints over the input variables to change of state could be the same as the constraints over the input variables to remain in the destination state of the transition, i.e., $C_{ij} = C_{jj}$. But, as we have stated, some tuning is needed to express different conditions to change:

$$C_{12} = a_x \text{ is } H_{a_x} \text{ AND } mov \text{ is } L_{mov} \text{ AND } tilt \text{ is } L_{tilt}$$

$$C_{21} = a_x \text{ is } L_{a_x} \text{ AND } mov \text{ is } L_{mov} \text{ AND } tilt \text{ is } H_{tilt}$$

$$C_{23} = a_x \text{ is } H_{a_x} \text{ AND } mov \text{ is } H_{mov} \text{ AND } tilt \text{ is } L_{tilt}$$

$$C_{32} = a_x \text{ is } H_{a_x} \text{ AND } mov \text{ is } L_{mov} \text{ AND } tilt \text{ is } L_{tilt}$$

$$C_{31} = a_x \text{ is } L_{a_x} \text{ AND } mov \text{ is } H_{mov} \text{ AND } tilt \text{ is } H_{tilt}$$

4) *Output variables*: Since we are going to use the output of this FFSM1 as input in the FFSM2, we can use as output variable the state activation vector, i.e., $Y[t] = S[t]$.

5) *Output function g* : The output function, as we have stated above, is simply: $g = S[t]$.

C. Human activity recognition module (FFSM2)

Currently, a new generation of smart-phones and PDAs including capabilities for WiFi communications and accelerometers is available. We use a PDA to obtain the information that our system requires for inferring the user activity. In the following, we explain how to design a FFSM for combining the WiFi Positioning module and the Posture Recognition module to achieve a Human Activity Recognition system:

1) *Fuzzy States*: The system must be adapted to each specific user. We manage linguistic descriptions of the different activities daily performed by the user. For example we distinguish among the following fuzzy states of activity in an office:

- Q_1 : Walking. Typical body movement detected by accelerometers
- Q_2 : Working at his/her desk. Usually, the user is seated, in specific WiFi coordinates, the most of time.
- Q_3 : Visiting a colleague. Seated or standing upright, in several possible WiFi coordinates, for little time.
- Q_4 : Having coffee. Seated or standing upright, in specific WiFi coordinates, some time.

- Q_5 : Having a meeting. Seated in specific WiFi coordinates for some time.

2) *Input variables*: In order to distinguish among the different states, we have created two linguistic variables $\{wep, pos\}$ that characterize the outputs of the two previous modules (FRBC and FFSM1):

- wep is the WiFi estimated position (FRBC computed it as explained in section IV-A).
- pos is the posture estimation obtained from the posture recognition module (FFSM1 described in section IV-B).

These variables are characterized by the following linguistic labels which are defined in the interval $[0,1]$:

- $wep = \{WAA, MC, WAB, WO, CA, MR\}$, which are the zones of our experimental scenario (see later section V).
- $pos = \{Seated, Upright, Walking\}$ which corresponds to the three different states of the FFSM1.

3) *Transition function f* : As in the FFSM1, we will obtain the next value of the activation vector using the transition function $S[t+1] = f(U[t], S[t])$.

Since we have already identified the relevant states in the model, we can represent the fuzzy rules that govern the temporal evolution of the system among these states. Figure 3 shows the transition diagram of the FFSM2. There are five rules to remain in a state (R_{ii}) and eight rules to change of state (R_{ij}). In this application not all the possible transitions are allowed, the majority of the states are connected to the state Q_1 (*Walking*).

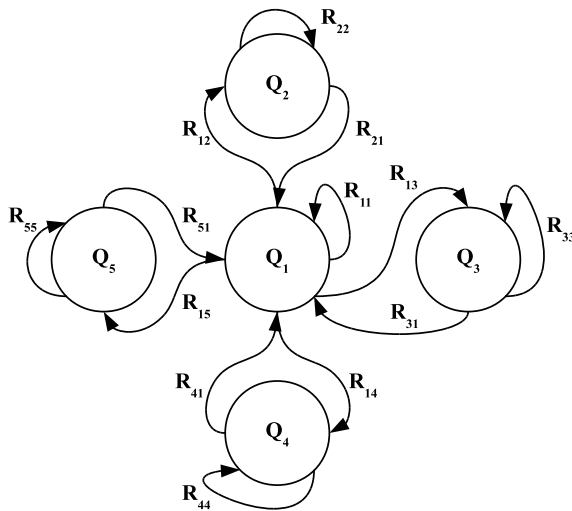


Fig. 3. Diagram of states of the FFSM2 (human activity recognition)

- *Rules to remain in a state*: Using the generic expression of R_{ii} explained in section III-C.1, we can define the constraints over the input variables to remain in each state.

$$\begin{aligned}
 C_{11} &= pos \text{ is } Walking \\
 C_{22} &= pos \text{ is } Seated \text{ AND } wep \text{ is } WAA \\
 C_{33} &= pos \text{ is } (Seated \text{ OR } Upright) \text{ AND } wep \text{ is } (WO \text{ OR } WAB) \\
 C_{44} &= pos \text{ is } (Seated \text{ OR } Upright) \text{ AND } wep \text{ is } CA
 \end{aligned}$$

$$C_{55} = pos \text{ is } Seated \text{ AND } wep \text{ is } MR$$

- *Rules to change of state*: Using the generic expression of R_{ij} explained in section III-C.2, we can define the constraints over the input variables to change of each state.

$$\begin{aligned}
 C_{12} &= pos \text{ is } Seated \text{ AND } wep \text{ is } WAA \\
 C_{13} &= pos \text{ is } (Seated \text{ OR } Upright) \text{ AND } wep \text{ is } (WO \text{ OR } WAB) \\
 C_{14} &= pos \text{ is } (Seated \text{ OR } Upright) \text{ AND } wep \text{ is } CA \\
 C_{15} &= pos \text{ is } Seated \text{ AND } wep \text{ is } MR \\
 C_{21} &= pos \text{ is } Walking \text{ AND } wep \text{ is } (WAA \text{ OR } MC) \\
 C_{31} &= pos \text{ is } Walking \text{ AND } wep \text{ is } WAB \\
 C_{41} &= pos \text{ is } Walking \text{ AND } wep \text{ is } MC \\
 C_{51} &= pos \text{ is } Walking \text{ AND } wep \text{ is } MR
 \end{aligned}$$

4) *Output variables*: We can use as output variable the state activation vector, i.e., $Y[t] = S[t]$. But we have to give a crisp description of the activity of the person. Therefore, we can consider as output the state with the maximum degree of activation at each instant of time t . However, this selection will make the FFSM very sensitive to noise and spurious in the signal, and that is precisely what we want to avoid. Therefore, the output is designed as the state which has had the maximum average degree of activation over the last second.

5) *Output function g* : The output function $g(U[t], S[t])$ that calculates the value of the output variables is, as we have stated above, the average operator in an interval of one second combined with the maximum operator to make the decision.

V. EXPERIMENTS

The experimentation took place at the premises of the European Centre for Soft Computing (ECSC). The layout of ECSC environment is shown in Figure 4. It has a surface of 440 m² illustrated on the top picture. We have identified six zones (look at the bottom picture): WAA (working area A), MC (main corridor), WAB (working area B), WO (working office), CA (coffee area), and MR (meeting room).

The user carried a HP iPAQ hw6910/hw6915 PDA. It has a WiFi interface with a maximum acquisition frequency of 4Hz, i.e., it is able to capture up to four samples per second. In addition, an external accelerometer (WiTilt v2.5) with acquisition frequency of 100Hz was connected to our PDA through Bluetooth. The user wore the accelerometer tight with a belt in the middle of his back. Our program measures both WiFi signal and accelerations in the same cycle with the aim of keeping synchronization. Notice that, each 25 measures provided by the accelerometer correspond to only one WiFi measure.

As it can be seen at the top picture in Figure 4, there are four APs in the environment covering most of the zones. Inside each zone we have set several training fixed positions. They are represented by filled circles at the bottom picture in the Figure. For each of them, we collected 100 samples from all the four APs. The resultant data set was taken as LRN for training the WiFi positioning module as explained in section IV-A. The FRBC contains four inputs (one per AP). First, we set strong fuzzy partitions with nine linguistic terms per input. Second, linguistic rules were induced with FDT. Third,



Fig. 4. Discretization of the ECSC environment

simplification was carried out. As a result, the final FRBC is made up of 14 rules with a total of 41 premises.

Table I gives the description of our experimental scenario that tries to summarize a normal day at the work. Of course, this is a simplified scenario where we have set a reduced time for the different tasks. For example, *Having a meeting* lasts less than 2 minutes. Notice that we wanted to test how our system is able to recognize all defined states of activity. The whole experiment takes about 9 minutes because the time walking is approximated. Furthermore, the same user has repeated eleven times the same experiment yielding more than one hour and a half of experimentation. There may be a slight time lag between different repetitions of the experiment when the user is walking. The first trial was used for tuning the FFSMs. Then, another different day, we run in a row the rest of ten executions which have been used for testing the previously designed system.

Table II includes the test averaged results for the ten repetitions of the experiment. We have reported results (in terms of misclassified samples) for all the three modules that constitute the system (look at Figure 1). The first row shows the percentage of error (about 14%) for the FRBC module. The two last rows are related to the two FFSMs. In both cases, the percentage of misclassified samples is very small (1.2 and 1.5% respectively). This is due to the characteristic memory effect of FFSMs which define the new state taking into account the transition conditions but also the previous state. In addition, output of FFSM2 is averaged in an interval of only one second what makes feasible the use of our system in real-time applications. It is also important to remark how

TABLE I
DESCRIPTION OF THE EXPERIMENTAL SCENARIO

Length (s)	Description	Activity
60	Seated and typing	Working at the desk
30	Standing up and walking towards the coffee area	Walking
75	Staying up in front of the coffee machine. Sitting and having the coffee	Having a coffee
25	Standing up and walking until the office of a colleague	Walking
50	Staying up and waiting for a colleague	Visiting a colleague
30	Walking towards the meeting room	Walking
100	Seated in the meeting room	Having a meeting
40	Standing up and walking back to the work-desk	Walking
100	Seated and typing	Working at the desk

TABLE II
PERCENTAGE OF MISCLASSIFIED SAMPLES

	Mean (%)	St. Deviation (%)
FRBC	13.8	4.7
FFSM1	1.2	0.2
FFSM2	1.5	0.7

FFSM2 is able to absorb and correct the errors produced by the FRBC due to the high variability in the WiFi signal. Note how the error is dramatically reduced from 14 to 1.5%.

Figure 5 plots the system output for the worse trial of our experiments, the one yielding the largest percentage of error. The figure is made up of three pictures. The first one (at the top) illustrates the state activation vector $S[t]$, while its activation value is printed by means of the gray intensity (black means one and white means zero). The picture below plots the output vector $Y[t]$ obtained from the output function g , i.e. the state which has had the maximum average degree of activation over the last one second. Finally, the picture at the bottom represents the expected output vector $Y[t]$, i.e., the right output of the FFSM at each instant of time defined in our experiment (as detailed in Table I). As it can be appreciated, most errors correspond to the situations of *Having a coffee* and *Visiting a colleague*. Indeed, it seems that the user was slightly moving while waiting for the coffee and for his colleague. In consequence, the state *Walking* is activated for a few seconds.

VI. CONCLUSIONS

In this paper we have described a system able to detect some simple tasks carried out by a human in a usual working day. The main contributions can be summarized as follows:

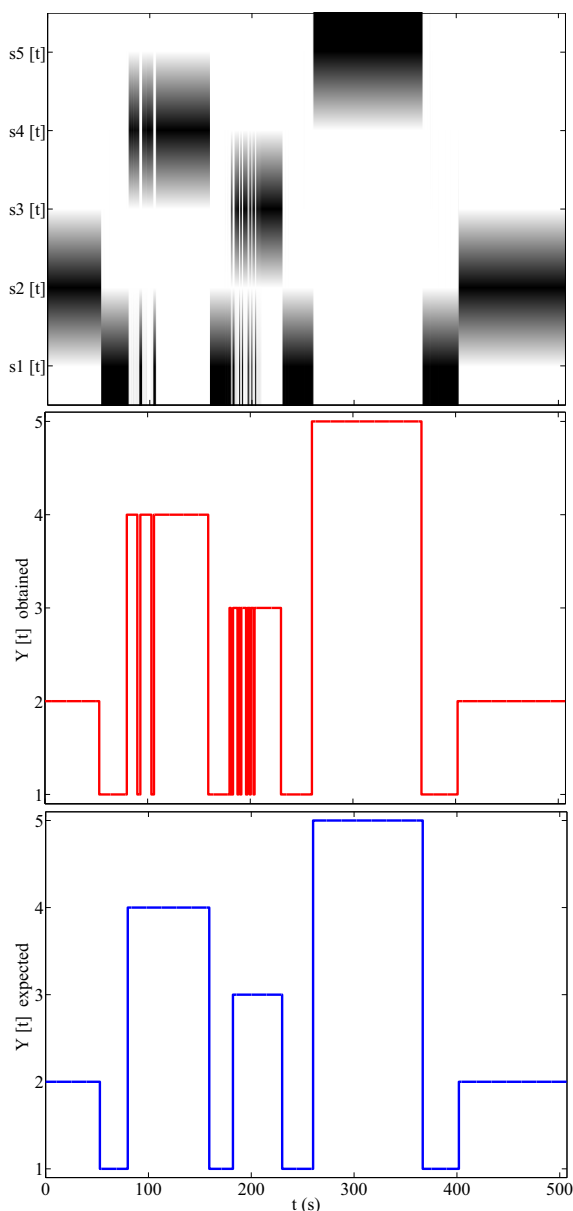


Fig. 5. System output (human activity recognition)

(1) the FRBC is mainly based on automatically induced knowledge yielding the approximated location of the user; (2) the FFMSs allow the designer to introduce constraints in the model based on the available expert knowledge about the activity states; (3) they also allow to fuse data from different sources and to compensate the partial errors, with the aim of producing a robust, reliable and easily understandable activity recognition module; and (4) the whole system is quite interpretable because it comprises several sets of linguistic variables and rules. In the future we will extend our model with the aim of detecting more complex human activities.

ACKNOWLEDGMENT

This work has been funded by the Spanish government (CICYT) under project TIN2008-06890-C02-01 and by S2009/DPI-1559 (Robocity 2030 II-CM Project) from the Science Department of Community of Madrid.

REFERENCES

- [1] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338–353, 1965.
- [2] —, "The concept of a linguistic variable and its application to approximate reasoning," *Parts I, II, and III. Information Sciences*, vol. 8, no. 8, pp. 199–249, 301–357, 43–80, 1975.
- [3] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic systems," *IEEE Transactions on Computers*, vol. 26(12), pp. 1182–1191, 1977.
- [4] J. L. Castro, "Fuzzy logic controllers are universal approximators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25, no. 4, pp. 629–635, 1995.
- [5] E. Hüllermeier, "Fuzzy methods in machine learning and data mining: Status and prospects," *Fuzzy Sets and Systems*, vol. 156, pp. 387–406, 2005.
- [6] D. Anderson et al., "Modeling human activity from voxel person using fuzzy logic," vol. 17, no. 1, pp. 39–49, 2009.
- [7] C. S. Chan and H. Liu, "Fuzzy qualitative human motion analysis," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 4, pp. 851–862, 2009.
- [8] R. Blasco et al., "Indoor localization based on neural networks for non-dedicated zigbee networks in AAL," *5517 Lecture Notes in Computer Science*, pp. 1113–1120, 2009.
- [9] J. M. Alonso, M. Ocaña, M. A. Sotelo, L. M. Bergasa, and L. Magdalena, "WiFi localization system using fuzzy rule-based classification," *Computer Aided Systems Theory - LNCS 5717 - EUROCAST 2009*, pp. 383–390, 2009.
- [10] G. Trivino and G. Bailador, "Linguistic description of human body posture using fuzzy logic and several levels of abstraction," in *Proceedings of the IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMS)*, 2007, pp. 105–109.
- [11] G. Trivino and A. van der Heide, "Linguistic summarization of the human activity using skin conductivity and accelerometers," in *Proceedings of the Conference Information Processing and Management of Uncertainty in Knowledge Based Systems (IPMU)*, 2008, pp. 1583–1589.
- [12] J. M. Alonso, L. Magdalena, and S. Guillaume, "HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism," *International Journal of Intelligent Systems*, vol. 23(7), pp. 761–794, 2008.
- [13] H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, "Neuro-fuzzy ID3: A method of inducing fuzzy decision trees with linear programming for maximizing entropy and an algebraic method for incremental learning," *Fuzzy Sets and Systems*, vol. 81, pp. 157–167, 1996.
- [14] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986.
- [15] G. Trivino, A. Alvarez-Alvarez, and G. Bailador, "Application of the computational theory of perceptions to human gait pattern recognition," *Pattern Recognition*, vol. 43, no. 7, pp. 2572–2581, 2010.
- [16] A. Alvarez and G. Trivino, "Comprehensible model of a quasi-periodic signal," in *Proc. of the 9th International Conference on Intelligent Systems Design and Applications (ISDA)*, 2009, pp. 450–455.
- [17] G. Bailador and G. Trivino, "Pattern recognition using temporal fuzzy automata," *Fuzzy Sets and Systems*, vol. 161, no. 1, pp. 37–55, 2010.
- [18] G. Trivino and A. van der Heide, "An experiment on the description of sequences of fuzzy perceptions," in *Proceedings of the 2008 8th International Conference on Hybrid Intelligent Systems (HIS)*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 228–233.
- [19] M. Sugeno, *Industrial Applications of Fuzzy Control*. New York, NY, USA: Elsevier Science Inc., 1985.
- [20] V. Matellán, J. M. Cañas, and O. Serrano, "WiFi localization methods for autonomous robots," *Robotica*, vol. 24, no. 4, pp. 455–461, 2006.
- [21] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. of the IEEE Infocom*, 2000, pp. 775–784.