Body Posture Recognition By Means Of A Genetic Fuzzy Finite State Machine

Alberto Alvarez-Alvarez, Gracian Trivino and Oscar Cordón
European Centre for Soft Computing (ECSC)
Mieres, Asturias, Spain
Email: {alberto.alvarez, gracian.trivino, oscar.cordon}@softcomputing.es

***Abstract*—Body posture recognition is a very important issue as a basis for the detection of user's behavior. In this paper, we propose the use of a genetic fuzzy finite state machine for this real-world application.**

Fuzzy finite state machines (FFSMs) are an extension of classical finite state machines where the states and inputs are defined and calculated by means of a fuzzy inference system, allowing them to handle imprecise and uncertain data. Since the definition of the knowledge base of the fuzzy inference system is a complex task for experts, we use an automatic method for learning this component based on the hybridization of FFSMs and genetic algorithms (GAs). This genetic fuzzy system learns automatically the fuzzy rules and membership functions of the FFSM devoted to body posture recognition while an expert defines the possible states and allowed transitions.

We aim to obtain a specific model (FFSM) with the capability of generalizing well under different subject's situations. The obtained model must become an accurate and human friendly linguistic description of this phenomenon, with the capability of identifying the posture of the user. A complete experimentation is developed to test the performance of the new proposal, comprising a detailed analysis of results which shows the advantages of our proposal in comparison with another classical technique.

I. Introduction

Body posture recognition consists of identifying the different poses of a human being. This research field has attracted considerable attention as a basis for the detection of user's behavior, which could provide new context aware services. Example of applications range from proactive care for elderly people to safety applications based on fall detection.

There are two well distinguished approaches to tackle this problem: the sensor-based and the computer vision approach. The sensor-based approach consists of using small sensors (usually accelerometers) placed in the body of the person. In [1], the authors showed how acceleration data can aid the recognition of pace and incline. The main advantages of this approach are the possibilities of embedding these sensors into clothes or electronic devices such as mobile phones due to the advances in miniaturization, the capabilities of communication between sensors through wireless connections, and the low cost and energy consumption thanks to to the Micro Electro Mechanical Systems (MEMS) technology. Its principal drawback is the user's need of wearing the sensors.

The computer vision approach is based on the use of video cameras installed in the scenario under study [2]. While the sensor-based approach made the user to wear sensors, in this case, the additional hardware must be installed in each room of the environment. This approach usually works in lab but fails in real world scenarios due to clutter, variable light intensity, and contrast. Moreover, the video cameras are sometimes perceived as invasive and threatening by some people. Another important drawback is the computational cost of working with video signals.

In this work, we propose the use of fuzzy finite state machines (FFSMs) for body posture recognition within the sensor-based approach. FFSMs are specially useful tools for modeling dynamical processes which change in time, becoming an extension of classical finite state machines. The main advantage of FFSMs is the use of Fuzzy Logic (FL), which provides semantic expressiveness by the use of linguistic variables [3] and rules [4] close to natural language (NL). Moreover, being universal approximators [5], fuzzy inference systems are able to perform nonlinear mappings between inputs and outputs, allowing FFSMs to handle imprecise and uncertain data, which is inherent to real-world phenomena, in the form of fuzzy states and transitions.

In previous studies [6], [7], we have learnt that FFSMs are suitable tools for modeling signals that follow an approximately repetitive pattern. As any fuzzy system, FFSMs require the definition of a knowledge base (KB). It is well known that this is a complex task for experts as it was the case in these previous works. In addition, the dynamic nature of FFSMs increases the complexity of the process. For that reason, in [8], we proposed a new automatic learning method for the fuzzy KB of FFSMs based on the use of genetic algorithms (GAs) [9]. GAs have proven largely their effectiveness and efficiency for the latter task in the last two decades in the so-called genetic fuzzy systems (GFSs) area [10], [11], [12]. In our approach, the fuzzy states and transitions are defined by the expert in order to keep the knowledge that she/he has over the whole system while the fuzzy rules and membership functions regulating the state changes will be derived automatically by the GFS. This combined action thus results in a robust, accurate, and human friendly model called genetic fuzzy finite state machine (GFFSM) [8].

In this contribution, we propose the use of a GFFSM for the body posture recognition problem. Our final goal is to obtain a specific model (FFSM) in such way that this FFSM can generalize well under different subject's situations. Moreover, the obtained FFSM will result in an accurate and human friendly linguistic description of this phenomenon, with the capability of identifying the posture of the user. A complete experimentation is developed to test the performance of the new proposal, comprising a detailed analysis of results which shows the advantages of our proposal in comparison with another classical technique. Furthermore, we will also compare this new proposal against a FFSM previously developed for body posture recognition, whose KB had been defined by the expert in [13].

978-1-61284-048-2/11/$26.00 ©2011 IEEE

The remainder of this paper is organized as follows. Section II presents how to use FFSMs for modeling the temporal evolution of a phenomenon. Section III explains how to build FFSMs for recognizing the body posture. The automatic method for learning the KB of these FFSMs based on GAs is introduced in Section IV. Section V presents the experimentation carried out, comparing the obtained results with another system identification tool. Finally, Section VI draws some conclusions and future research works.

II. Fuzzy Finite State Machines

In this section, we introduce the main concepts and elements of our paradigm for system modeling allowing experts to build comprehensible fuzzy linguistic models in an easier way. In our framework, a FFSM is a tuple $\{Q, U, f, Y, g\}$, where:

- Q is the state of the system.
- U is the input vector of the system.
- f is the transition function which calculates the state of the system.
- Y is the output vector of the system.
- g is the output function which calculates the output vector.

Each of these components are described in the following subsections. Furthermore, the interested reader can refer to [6], [7], [8], [13] for a more detailed description.

A. Fuzzy States (Q)

The state of the system (Q) is defined as a linguistic variable [3] that takes its values in the set of linguistic labels $\{q_1, q_2, \ldots, q_n\}$, with n being the number of fuzzy states. Every fuzzy state represents the pattern of a repetitive situation and it is represented numerically by a state activation vector:

$S[t] = (s_1[t], s_2[t], \ldots, s_n[t])$, where $s_i[t] \in [0, 1]$ and $\sum_{i=1}^{n} s_i[t] = 1$.

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

B. Input Vector (U)

U is the input vector $(u_1, u_2, ..., u_{n_u})$, with n_u being the number of input variables. U is a set of linguistic variables obtained after fuzzification of numerical data. Typically, u_i can be directly obtained from sensor data or by applying some calculations to the raw measures, e.g., the derivative or integral of the signal, or the combination of several signals. The domain of numerical values that u_i can take is represented by a set of linguistic labels, $A_{u_i} = \{A^1_{u_i}, A^2_{u_i}, \ldots, A^{n_i}_{u_i}\}$, with n_i being the number of linguistic labels of the linguistic variable u_i.

C. Transition Function (f)

The transition function (f) calculates, at each time instant, the next value of the state activation vector: $S[t+1] = f(U[t], S[t])$. It is implemented by means of a fuzzy KB.

Once the expert has identified the relevant states in the model, she/he must define the allowed transitions among states. There are rules R_{ii} to remain in a state q_i, and rules R_{ij} to change from state q_i to state q_j. If a transition is forbidden in the FFSM, it will have no fuzzy rules associated.

A generic expression of a rule is of the form:

R_{ij}: **IF** ($S[t]$ is q_i) **AND** C_{ij} **THEN** $S[t+1]$ is q_j

where:

- The first term in the antecedent ($S[t]$ is q_i) computes the degree of activation of the state q_i in the time instant t, i.e., $s_i[t]$. With this mechanism, we only allow the FFSM to change from the state q_i to the state q_j (or to remain in state q_i, when $i = j$).
- The second term in the antecedent C_{ij} describes the constraints imposed on the input variables in disjunctive normal form (DNF) [10]. For example: $C_{ij} = (u_1[t]$ is $A^3_{u_1})$ **AND** ($u_2[t]$ is $A^4_{u_2}$ **OR** $A^5_{u_2}$).
- Finally, the consequent of the rule defines the next value of the state activation vector $S[t+1]$. It consists of a vector with a zero value in all of its components but in $s_j[t]$, where it takes value one.

To calculate the next value of the state activation vector ($S[t+1]$), a weighted average using the firing degree of each rule k (ω_k) is computed as defined in Equation 1:

$$S[t+1] = \begin{cases} \dfrac{\sum_{k=1}^{\#Rules} \omega_k \cdot (s_1, \ldots, s_n)_k}{\sum_{k=1}^{\#Rules} \omega_k} & \text{if} \quad \sum_{k=1}^{\#Rules} \omega_k \neq 0 \\ S[t] & \text{if} \quad \sum_{k=1}^{\#Rules} \omega_k = 0 \end{cases} \quad (1)$$

where (ω_k) is calculated using the minimum for the AND operator and the bounded sum of Łukasiewicz [14] for the OR operator.

D. Output Vector (Y)

Y is the output vector: $(y_1, y_2, ..., y_{n_y})$, with n_y being the number of output variables. Y is a summary of the characteristics of the system evolution that are relevant for the application.

E. Output Function (g)

The output function (g) calculates, at each time instant, the next value of the output vector: $Y[t] = f(U[t], S[t])$. The most simple implementation of g is $Y[t] = S[t]$.

III. Fuzzy Finite State Machine for Body Posture Recognition

This section presents the design of the main elements needed to build a FFSM for body posture recognition [13].

A. Fuzzy States

In this application, we have defined three different fuzzy states which directly describe the body posture:

$$\{q_1 \rightarrow \text{Seated},\ q_2 \rightarrow \text{Upright},\ q_3 \rightarrow \text{Walking}\}$$

B. Input Vector

In our experiments, 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). In order to distinguish between the three different states, we have created three linguistic variables $\{a_x, mov, tilt\}$ with these numerical values:

- a_x is the dorso-ventral acceleration as it was obtained from the sensor.

- mov measures the amount of movement. It is the sum of the difference between the maximum and minimum of a_x, a_y, and a_z, respectively, contained in an 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 term sets for each linguistic variable are: $\{S_{a_x}, M_{a_x}, B_{a_x}\}$, $\{S_{mov}, M_{mov}, B_{mov}\}$, and $\{S_{tilt}, M_{tilt}, B_{tilt}\}$, where S, M, and B are linguistic terms representing small, medium, and big, respectively.

C. Transition Function

The definition of which transitions are allowed and which are not can be easily represented by means of the state diagram. Figure 1 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). This restriction makes the system more robust.

From the state diagram represented in Figure 1, it can be recognized that there are 8 fuzzy rules overall in the system: 3 rules to remain in each state and other 5 to change between states.

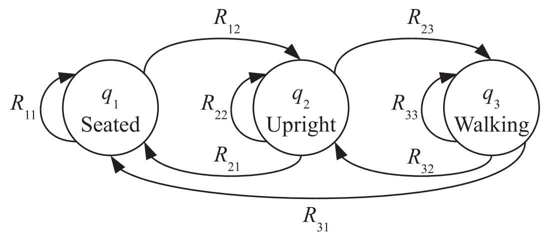

Fig. 1. State diagram of the FFSM for body posture recognition.

Therefore, the RB will have the following structure:

R_{11}: **IF** ($S[t]$ is q_1) **AND** C_{11} **THEN** $S[t+1]$ is q_1
R_{22}: **IF** ($S[t]$ is q_2) **AND** C_{22} **THEN** $S[t+1]$ is q_2
R_{33}: **IF** ($S[t]$ is q_3) **AND** C_{33} **THEN** $S[t+1]$ is q_3
R_{12}: **IF** ($S[t]$ is q_1) **AND** C_{12} **THEN** $S[t+1]$ is q_2
R_{21}: **IF** ($S[t]$ is q_2) **AND** C_{21} **THEN** $S[t+1]$ is q_1
R_{23}: **IF** ($S[t]$ is q_2) **AND** C_{23} **THEN** $S[t+1]$ is q_3
R_{32}: **IF** ($S[t]$ is q_3) **AND** C_{32} **THEN** $S[t+1]$ is q_2
R_{31}: **IF** ($S[t]$ is q_3) **AND** C_{31} **THEN** $S[t+1]$ is q_1

where C_{ij} could be: (a_x is S_{a_x}) **AND** (mov is M_{mov}) **AND** ($tilt$ is M_{tilt} **OR** B_{tilt}).

D. Output Vector and Output Function

In this contribution, we simply consider $Y[t] = S[t]$, i.e., the output of the FFSM is the degree of activation of each state.

IV. Genetic Fuzzy System

The current section reviews the fusion framework between FFSMs and GAs developed in [8], which will be considered to solve a new application in the current contribution. In this case, we will learn the KB of the FFSM designed for body posture recognition. The KB is comprised by the data base (DB), which contains the linguistic labels' membership functions (MFs); and the RB, which collects the fuzzy if-then rules. The following subsections describe the structure of the different components of this GFS.

A. Representation Scheme and Initial Population Generation

We have divided the representation scheme into two parts: the RB part and the DB part. In the following, we explain each of these representations.

1) RB part: We codify the whole rule set in a chromosome following the Pittsburgh approach [15]. For each of the three input variables a_x, mov, and $tilt$, the rule representation consists of a binary sub-string of length 3 that refers to its linguistic term set $\{S_{a_x}, M_{a_x}, B_{a_x}\}$, $\{S_{mov}, M_{mov}, B_{mov}\}$, and $\{S_{tilt}, M_{tilt}, B_{tilt}\}$, respectively. Only the non-fixed part of the DNF rule antecedent (see Section III-C) is encoded. Each bit has a one (zero) which denotes the presence (absence) of each linguistic term in the rule. Moreover, the feature selection capability of this representation is used since an input variable is omitted in the rule if all of its bits in the representation become zeros or ones. The RB part of the chromosome will thus be composed of 8 rules $\times$ 9 linguistic terms (3 per input variable) = 72 binary-coded genes.

2) DB part: We have considered the use of trapezoidal strong fuzzy partitions (SFPs) [16] because they allow us to reduce the number of parameters to tune, in such way that the normalization constraint is easily satisfied by only coding the two modal points of each MF. Therefore, we have to code 12 real parameters, 4 per input variable where one parameter is enough to codify the first and third linguistic label and two parameters are needed to codify the second linguistic label. Therefore, the DB part of the chromosome will be composed of 12 real-coded genes:

$$\{a^1_{a_x}, a^2_{a_x}, b^2_{a_x}, a^3_{a_x}, a^1_{mov}, a^2_{mov}, b^2_{mov}, a^3_{mov}, a^1_{tilt}, a^2_{tilt}, b^2_{tilt}, a^3_{tilt}\}$$

We use a real-coded representation. The variation interval of each allele is defined within the interval defined by its previous and next parameter. Figure 2 shows the graphical representation of the fuzzy partition related with the linguistic input variable mov. Notice that, the parameters a^1_{mov} and a^3_{mov} are enough to codify the first and third linguistic labels, S_{mov} and B_{mov} respectively, while two parameters a^2_{mov} and b^2_{mov} are needed to codify the intermediate linguistic label M_{mov}.

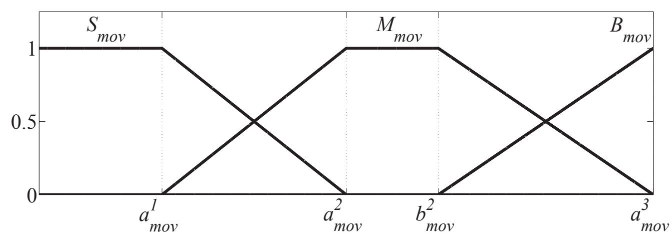

Fig. 2. Parameters that form all the trapezoidal linguistic labels of the linguistic variable mov.

Hence, the final chromosome encoding a candidate problem solution will be comprised by $72 + 12 = 84$ genes, with the first 72 being binary-coded genes corresponding to the RB part, and the last 12 being real-coded genes associated to the DB part. We have initialized the first population by generating all the individuals at random, except the DB part

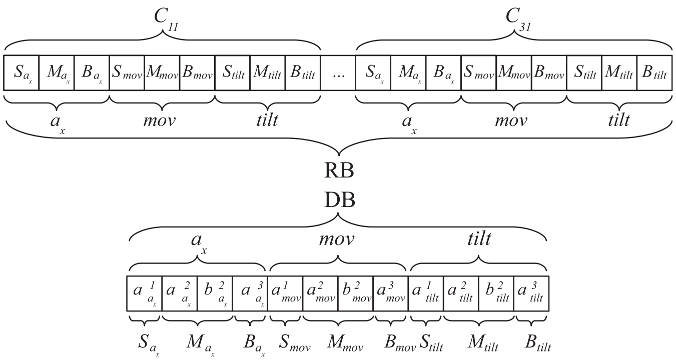

Fig. 3. Chromosome structure encoding the RB and DB part.

of the first individual of the population that encodes uniform fuzzy partitions for each linguistic variable. Figure 3 shows the structure of the complete chromosome encoding the RB and DB part.

B. Fitness Function

Since the computation of the next state is based on the previous state, we need to evaluate the tentative FFSM definition encoded in each chromosome over the whole data set. We have chosen as fitness function the mean absolute error (MAE) measure, defined in Equation 2:

$$\text{MAE} = \frac{1}{n} \cdot \frac{1}{T} \cdot \sum_{i=1}^{n} \sum_{j=0}^{T} |s_i[j] - s_i^*[j]| \tag{2}$$

where:

- n is the number of states, i.e., $n = 3$.
- T is the dataset size (i.e., the considered time interval duration).
- $s_i[j]$ is the degree of activation of state q_i at time $t = j$.
- $s_i^*[j]$ is the expected degree of activation of state q_i at time $t = j$.

The MAE directly measures the difference between the actual state activation vector ($S^*[t]$) and the obtained one ($S[t]$). However, we need to define $S^*[t]$ for each input data set that we want to learn. This definition could be problematic and must be done carefully because, more than one state can be defined at each time instant, each of those states activated with certain degree in the interval $[0, 1]$. In the following subsection, this issue is explained in detail.

C. Defining the Training Data Set

We have to create a training vector which consists of $a_x[t]$, $a_y[t]$, $a_z[t]$ and $S^*[t]$, i.e., $(a_x[t], a_y[t], a_z[t], s_1^*[t], s_2^*[t], s_3^*[t])$. To define it, we have developed a user-friendly graphical interface that allows the expert to select manually the relevant points where each state starts and ends using her/his knowledge about body posture and the duration of each part of the experiment. The fuzzy definition of the states is based on the imprecision of the expert when defining those relevant points. For each state q_i, there are different points comprising the beginning (b_i^j) and the end of each state (e_i^j), with $j \in \mathbb{N}$.

As an example, let us consider the definition of the actual degree of activation of state q_3 when there is a transition from state q_2 to state q_3 and then from state q_3 to state q_1. The actual value of $s_3^*[t]$ is then specified by Equation 3. Between the end time of q_2 (e_2^j) and the start time of q_3 (b_3^j), the activation of the state q_3 is rising from 0 to 1. Between the start (b_3^j) and the end time (e_3^j) of q_3, defined by the user, the activation has the maximum of 1. Afterwards, the activation drops till zero at the start of q_1 (b_1^j). Otherwise, the activation is zero.

$$s_3^*[t] = \begin{cases} \frac{t - e_2^j}{b_3^j - e_2^j} & \text{if} \quad e_2^j < t < b_3^j \\ 1 & \text{if} \quad b_3^j \leq t \leq e_3^j \\ \frac{b_1^j - t}{b_1^j - e_3^j} & \text{if} \quad e_3^j < t < b_1^j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The interested reader is referred to [8] for a deeper description on the human definition of the activation degrees for the fuzzy states in FFSMs.

D. Genetic Operators

A binary tournament selection and a generational replacement with elitism are considered. The classical two-point crossover has been used for the RB (binary-coded) part of the chromosome and BLX-α crossover [17] for the DB (real-coded) part. The BLX-α crossover is applied twice over a pair of parents in order to obtain a new pair of chromosomes. The classical bitwise mutation has been selected for the binary-coded RB part, while uniform mutation has been chosen for the real-coded DB part.

In this contribution, we have implemented three different termination conditions. First, the search is stopped when the algorithm has obtained a fitness value equal to zero, which is the best value that the fitness function can take. Moreover, we have decided to set a maximum number of generations and also to stop the search when, for a certain number of generations, the fitness value of the best individual is not improved.

V. Experimentation

This section presents the experimentation carried out. First, the experimental setup, which comprises the data acquisition and the parameters of the GA, is explained. The second part contains a brief description of an alternative model used for body posture recognition. Finally, the third part presents and analyzes the results obtained.

A. Experimental Setup

1) Data acquisition: We have used a wireless three-axial accelerometer attached to a belt, centered in the back of the person. It provided measurements of the three orthogonal accelerations with a frequency of 100 Hz. Therefore, every record contained the information: (t, a_x, a_y, a_z) where t is each instant of time, a_x is the dorso-ventral acceleration, a_y is the medio-lateral acceleration, and a_z is the antero-posterior acceleration.

We asked this person to perform a variety of activities, such as sitting at her/his desk, having a coffee, visiting a colleague, having a meeting, etc. In this simplified scenario, we have set a reduced time for the different tasks because we wanted to test how our system is able to recognize all defined states related to body posture. This process was repeated ten times producing ten different datasets. These datasets were then processed as explained in Section IV-C getting the following structure:

$$(a_x[t], a_y[t], a_z[t], s_1^*[t], s_2^*[t], s_3^*[t])$$

where:

- $a_x[t]$ is the dorso-ventral acceleration at time instant t.
- $a_y[t]$ is the medio-lateral acceleration at time instant t.
- $a_z[t]$ is the antero-posterior acceleration at time instant t.
- $s_1^*[t]$ is the expected degree of activation of state q_1 at time instant t.
- $s_2^*[t]$ is the expected degree of activation of state q_2 at time instant t.
- $s_3^*[t]$ is the expected degree of activation of state q_3 at time instant t.

2) Parameters of the GA:

- Population size $\rightarrow$ 30 individuals.
- Crossover probability $\rightarrow p_c = 0.8$.
- Value of alpha (BLX-α parameter) $\rightarrow \alpha = 0.3$.
- Mutation probability per bit $\rightarrow p_m = 0.02$.
- Termination conditions:
  - Fitness value reached $\rightarrow$ MAE = 0.
  - Maximum number of generations $\rightarrow$ 200.
  - Generations without improvement of the fitness function $\rightarrow$ 50.

B. Autoregressive Linear Models

In order to benchmark the GFFSM results, we have considered another technique commonly used in system modeling of time-dependent systems: autoregressive linear models (ARX) [18]. We have defined a multiple-input multiple-output (MIMO) ARX model with the structure defined by Equation 4:

$$\begin{aligned} Y[t] &= A_1 \cdot Y[t-1] + \ldots + A_{n_A} \cdot Y[t-n_A] \\ &+ B_0 \cdot U[t] + \ldots + B_{n_B} \cdot U[t-n_B] \end{aligned} \quad (4)$$

where:

- $Y[t] = (s_1[t], s_2[t], s_3[t])$ is the current output vector.
- $Y[t-1], \ldots, Y[t-n_A]$ are the previous output vectors on which the current output vector depends.
- $U[t] = (a_x[t], mov[t], tilt[t]), \ldots, U[t-n_B]$ are the current and delayed input vectors on which the current output vector depends.
- n_A is the number of previous output vectors on which the current output vector depends.
- n_B is the number of previous input vectors on which the current output vector depends.
- $A_1, \ldots, A_{n_A}$ and $B_0, \ldots, B_{n_B}$ are the matrices that define the models. They are estimated using the least squares method.

The performance of this model has been tested with values of $n_A = n_B = 20$, resulting in the ARX model defined by Equation 5:

$$\begin{aligned} Y[t] &= A_1 \cdot Y[t-1] + \ldots + A_{20} \cdot Y[t-20] \\ &+ B_0 \cdot U[t] + \ldots + B_{19} \cdot U[t-19] \end{aligned} \quad (5)$$

C. Results

To test the performance of the GFFSM and the ARX model, we have done a leave-one-out cross validation for each of the 10 datasets. Table I shows the MAE obtained for each fold of the leave-one-out in training and test. It also depicts the average value of the MAE (MEAN) and its standard deviation (STD) for the ten results of the procedure.

TABLE I
MAE FOR EACH DATASET OF THE LEAVE-ONE-OUT.

FOLD	GFFSM		ARX	
	TRAIN	TEST	TRAIN	TEST
1	0.010	0.016	0.071	0.083
2	0.009	0.007	0.072	0.093
3	0.010	0.009	0.076	0.064
4	0.009	0.010	0.078	0.059
5	0.010	0.013	0.076	0.072
6	0.009	0.012	0.075	0.073
7	0.010	0.010	0.075	0.081
8	0.011	0.009	0.070	0.104
9	0.008	0.010	0.077	0.065
10	0.009	0.009	0.076	0.072
MEAN	0.009	0.011	0.074	0.077
STD	0.001	0.002	0.003	0.014

TABLE II
MAE OBTAINED FOR EACH DATASET BY THE FFSM DEFINED BY THE EXPERT AND OBTAINED IN TEST WITH THE LEAVE-ONE-OUT.

DATASET	FFSM	GFFSM	ARX
1	0.023	0.016	0.083
2	0.027	0.007	0.093
3	0.016	0.009	0.064
4	0.020	0.010	0.059
5	0.022	0.013	0.072
6	0.028	0.012	0.073
7	0.022	0.010	0.081
8	0.030	0.009	0.104
9	0.017	0.010	0.065
10	0.018	0.009	0.072
MEAN	0.022	0.011	0.077
STD	0.005	0.002	0.014

In addition, we have evaluated the FFSM manually defined by the expert in [13] (where no training data has been used) over these ten datasets. Table II shows the MAE obtained for each dataset by the expert FFSM and the MAE obtained in test with the leave-one-out procedure for the GFFSM and the ARX model.

It can be easily observed that our proposal (GFFSM) and the FFSM defined by the expert obtain better results than the autoregressive linear model (ARX). Moreover, ARX models are black-box models not understandable by the human expert while our GFFSM is able to describe and

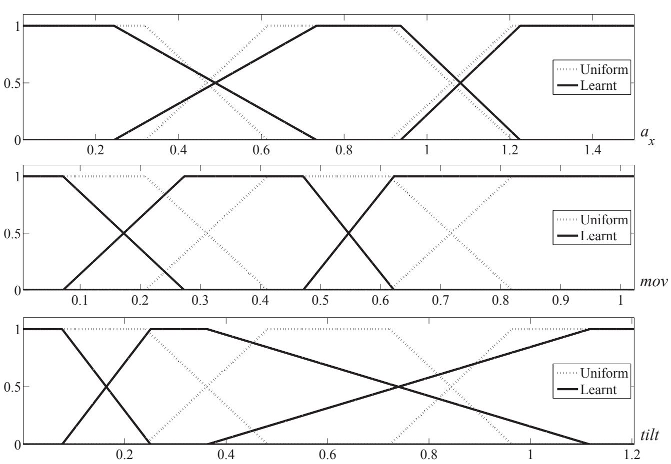

Fig. 4. Linguistic labels' trapezoidal MFs of each linguistic variable which comprise the learnt DB compared with the uniformly distributed original ones.

model the body posture by means of linguistic fuzzy if-then rules achieving a good interpretability-accuracy tradeoff.

Note that, with the application of the genetic learning procedure proposed in [8], we have increased the accuracy of the FFSM defined by the expert (by reducing the average MAE from 0.022 to 0.011) keeping her/his knowledge about this application, and producing a RB and a DB that have the same interpretability as the former.

As an example of how our novel proposal is describing linguistically the temporal evolution of the body posture, a complete set of constraints imposed on the input variables (which forms the RB as explained in III-C) learned for the second fold of the leave-one-out procedure is shown as follows:

$$
\begin{aligned}
C_{11} &= (a_x \text{ is } M_{a_x}) \textbf{ AND } (mov \text{ is } S_{mov} \textbf{ OR } B_{mov}) \textbf{ AND } (tilt \text{ is } B_{tilt}) \\
&= (a_x \text{ is } M_{a_x}) \textbf{ AND } (mov \text{ is } \neg M_{mov})^1 \textbf{ AND } (tilt \text{ is } B_{tilt}) \\
C_{22} &= (a_x \text{ is } B_{a_x}) \\
C_{33} &= (mov \text{ is } M_{mov}) \\
C_{12} &= (a_x \text{ is } \neg S_{a_x}) \textbf{ AND } (mov \text{ is } \neg S_{mov}) \textbf{ AND } (tilt \text{ is } \neg B_{tilt}) \\
C_{21} &= (a_x \text{ is } S_{a_x}) \textbf{ AND } (mov \text{ is } \neg M_{mov}) \textbf{ AND } (tilt \text{ is } B_{tilt}) \\
C_{23} &= (a_x \text{ is } B_{a_x}) \textbf{ AND } (mov \text{ is } \neg S_{mov}) \textbf{ AND } (tilt \text{ is } S_{tilt}) \\
C_{32} &= (mov \text{ is } S_{mov}) \textbf{ AND } (tilt \text{ is } \neg B_{tilt}) \\
C_{31} &= (a_x \text{ is } S_{a_x}) \textbf{ AND } (mov \text{ is } S_{mov}) \textbf{ AND } (tilt \text{ is } M_{tilt})
\end{aligned}
$$

Figure 4 shows the graphical representation of the learnt DB associated with this RB. The initial DB is also plotted, which consists of uniformly distributed MFs.

VI. Concluding Remarks

We have presented a practical application where we described how to build a FFSM to recognize the body posture in a dynamical environment. We defined three different states related to the body posture and applied the FFSM genetic learning procedure proposed in [8] to recognize these states.

This GFS can obtain automatically the fuzzy rules and fuzzy MFs associated to the linguistic terms of the FFSM while the states and transitions are defined by the expert, thus maintaining the knowledge that she/he has about the application. The results obtained by the GFFSM showed the goodness of our proposal. Moreover, its ability to combine the handling of the available expert knowledge with the accuracy achieved by the learning process can be used to study several phenomena where the human interaction is demanded.

Acknowledgment

This work has been funded by the Spanish Ministerio de Ciencia e Innovación (MICINN) under project TIN2008-06890-C02-01; and under project TIN2009-07727, including EDRF fundings.

References

[1] K. Aminian, P. Robert, E. Jequier, and Y. Schutz, "Estimation of speed and incline of walking using neural network," *Instrumentation and Measurement, IEEE Transactions on*, vol. 44, no. 3, pp. 743–746, 1995.

[2] T. Banerjee, J. M. Keller, M. Skubic, and C. Abbott, "Sit-to-stand detection using fuzzy clustering techniques," in *Proceedings of the 2010 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Jul. 18-23, Barcelona, Spain*, 2010, pp. 2836–2843.

[3] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Parts I, II, and III. Information Sciences*, vol. 8, 8, 9, pp. 199–249, 301–357, 43–80, 1975.

[4] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic systems," *Computers, IEEE Transactions on*, vol. 26, no. 12, pp. 1182–1191, Dec. 1977.

[5] J. L. Castro, "Fuzzy logic controllers are universal approximators," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 25, no. 4, pp. 629–635, Apr. 1995.

[6] A. Alvarez-Alvarez and G. Trivino, "Comprehensible model of a quasi-periodic signal," in *Proceedings of the 9^{th} International Conference on Intelligent Systems Design and Applications (ISDA)*. Los Alamitos, CA, USA: IEEE Computer Society, 2009, pp. 450–455.

[7] 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.

[8] A. Alvarez-Alvarez, G. Trivino, and O. Cordón, "Human gait modeling using a genetic fuzzy finite state machine," European Centre for Soft Computing Research Report: AFE 2010-08, CGC 2010-01, Mieres, Spain. 2010. *Submitted*.

[9] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*. Springer-Verlag, 2003.

[10] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems. Evolutionary tuning and learning of fuzzy knowledge bases, Advances in Fuzzy Systems: Applications and Theory*. World Scientific, 2001.

[11] O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: current framework and new trends," *Fuzzy Sets and Systems*, vol. 141, no. 1, pp. 5–31, 2004.

[12] F. Herrera, "Genetic fuzzy systems: taxonomy, current research trends and prospects," *Evolutionary Intelligence*, vol. 1, no. 1, pp. 27–46, Mar. 2008.

[13] A. Alvarez-Alvarez, J. M. Alonso, G. Trivino, N. Hernández, F. Herranz, A. Llamazares, and M. Ocaña, "Human activity recognition applying computational intelligence techniques for fusing information related to WiFi positioning and body posture," in *Proceedings of the 2010 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Jul. 18-23, Barcelona, Spain*, 2010, pp. 1881–1885.

[14] C. Alsina, M. J. Frank, and B. Schweizer, *Associative Functions: Triangular Norms and Copulas*. Singapore: World Scientific Publishing Company, 2006.

[15] K. De Jong, W. M. Spears, and D. F. Gordon, "Using genetic algorithms for concept learning," *Machine Learning*, vol. 13, no. 2-3, pp. 161–188, 1993.

[16] E. H. Ruspini, "A new approach to clustering," *Information and Control*, vol. 15, pp. 22–32, 1969.

[17] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," in *Proceedings of the 2^{nd} Workshop on Foundations of Genetic Algorithms (FOGA)*. Morgan Kaufmann, 1993, pp. 187–202.

[18] L. Ljung, *System identification: theory for the user*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1986.

[1] Notice that, the symbol $\neg$ stands for the negation of the linguistic term M_{mov}, i.e., $\neg M_{mov}(mov) = 1 - M_{mov}(mov)$. With the fuzzy reasoning mechanism defined in II-C and the use of SFPs for the MFs, the antecedent (mov is S_{mov} **OR** B_{mov}) can be replaced by (mov is $\neg M_{mov}$).