Bailador, G., & Guadarrama, S. (2007). Robust Gesture Recognition using a Prediction-Error-Classification Approach. 2007 IEEE International Fuzzy Systems Conference, 1–7. https://doi.org/10.1109/FUZZY.2007.4295635

Robust Gesture Recognition using a Prediction-Error-Classification Approach

Gonzalo Bailador Sergio Guadarrama

***Abstract*— The main idea of this paper consists on doing gesture recognition by means of prediction. Taking into account that a signal predictor will predict accurately future values of gestures of its class and inaccurately the values of the others, we can use the prediction error to classify the gestures. These predictors are implemented using Neuro Fuzzy Systems. We call this approach Prediction-Error-Classification approach (PEC) and this idea represents a different approach to solve the problem of gesture recognition in real time using inexpensive accelerometers.**

To validate this approach we have studied the impact of the number of training samples in the prediction error using cross-validation. We have also studied the impact of the number of training samples in the recognition rate, using again cross-validation. And to test the robustness and applicability in a real situation of this approach, we have repeated all the tests with a more realistic experiment.

I. Introduction

Automatic gesture recognition has the potential to create powerful human computer interfaces hence, a significant effort has been done lately in this topic. Furthermore, advances in technology are producing sensors lighter and with lower consumption. A good example of this kind of sensors are MEMS accelerometers, which can provide proper information to acquire and recognize gestures.

In the last years a variety of applications of gesture recognition have been done using these sensors. For example, in [4] they were used to label sport video sequences, in [8] they were used in video games to replace a regular game-pad (also done in the new Nintendo console Wii), or in [10] they were used to recognize gestures done by workers in a wood shop in order to monitor their work.

Most of the previous works in gesture recognition from acceleration signals are based on discrete [11] or continuous HMM [12](Hidden Markov Models). These methods allow to reach high recognition rates however, they also present a high computational cost especially continuous ones. Nevertheless in [5], approximate classification methods (as Fuzzy c-means, Self Organization Maps and Fuzzy rules system) have been compared with HMM showing a better robustness and a lower computational cost. A different approach to solve this problem was presented in [9] where they transform the accelerometer signals into a 3D trajectory and then comparing it with a pre-defined set of trajectories to classify the gesture.

The prediction approach has been used sucessfully in previous works, for example in [13] they use predictors based on kalman filter. The main drawback of using this kind of methods is the complexity of design the necessary models (arm model,control model,...). On the other hand, other works have used neural networks to create these predictors. However, our previous work [2] using continuous time recurrent neural networks showed us their lower performance with gestures captured in realistic environments.

In the same line that our **neuro-fuzzy** approach there is a work done by Juang and Ku [7], that although is similar in the general approach it is different on the problem tackled, because they want to recognize gestures from video sequences, which are captured by a CCD camera, and they use recurrent fuzzy networks. Due to the different nature of our sensors and of the different kind of experiments, their solution is not directly applicable to our problem. Although it will be interesting try to merge both solutions in the future.

In this paper we have extended and improved our previous work done in [3]. In which we introduced the idea of doing classification based on signal predictors, and showed the suitability of Neuro Fuzzy Systems as signal predictors.

En el siguiente parrafo se explica la ventaja propuesta por el primer review sobre que no hay que reentrenar. Lo unico que no tengo muy claro es si nuestro enfoque es realmente divide y venceras. Estas seguro?
Usually in classification approaches, for each new class the whole system must be retrained but in the Prediction-Error-Classification approach a divide-and-conquer technique is adopted, and any new class only needs to be trained by itself, generating its own predictor that will be later used to compare with the others. In the test phase we use the prediction error as the recognition criterion. So, when an unknown gesture is presented, the predicted mean error of each trained predictor is calculated and compared, and the one with the lowest error is regarded as the class to which the gesture belongs.

The rest of the paper is organized as follows: in section 2 the device used for signal capture is described; Section 3 explains the structure of the Prediction-Error-Classification system and how to use predictors to recognize gestures; In section 4 we describe the experiments performed; In section 5 we analyze the prediction capability of predictors to support the underlying hypothesis; In section 6 we analyze the classification rates for different situations to validate the PEC system. And finally, section 7 highlight the results and

Gonzalo Bailador is with the Department of Tecnología Fotónica, School of Computer Science, Technical University of Madrid, Campus de Montegancedo, Boadilla del Monte, Spain, (email: gonzalo.bailador@upm.es).

Sergio Guadarrama is with the Department of Artificial Intelligence, Technical University of Madrid, Campus de Montegancedo, Boadilla del Monte, Spain (email: sguada@dia.fi.upm.es).

1-4244-1210-2/07/$25.00 ©2007 IEEE.

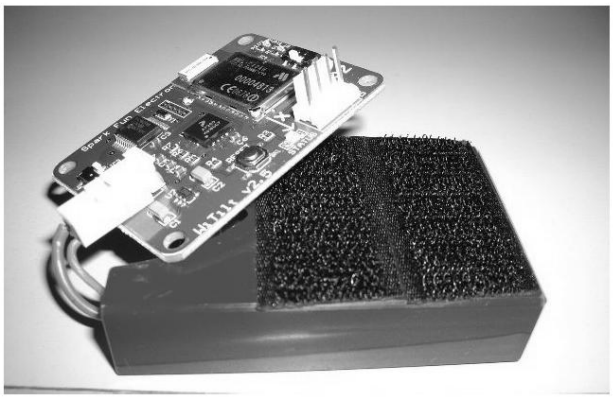

Fig. 1. Tri-Axial Acceleration Sensor 64x40x15 milimeters

concludes this paper.

II. Sensor Hardware

In order to capture the gestures used in the following experiments, a tri-axial accelerometer [1] is used. This device was selected because it has good features for wearable applications. It has a small size, so it can be worn easily (see figure 1), and also has a very low power consumption so it can operate for long periods of time (12 hours with a single battery). This accelerometer module is connected over Bluetooth to the personal computer capturing the data. The sensor was placed in the reverse of the hand using a velcro strap, so the person can use the hand for other purposes during the realistic experiments.

The data provided by this sensor consists in an acceleration vector with three components: one for each axis (A_X, A_Y, A_Z). Their values are measured in gravity units (g) in the range of [-4g,4g] encoded with 10 bits. This vector is captured with a sampling rate of 100 Hz, which is fast enough for our purpose since the maximum frequency of hand gestures is about 10 Hz [14].

During the experiments, this sensor was in one hand in vertical orientation and a wireless mouse was held in the other. The acceleration data together with the data from the wireless mouse was recorded on the personal computer. Mouse button was pressed before starting the gesture and released after finishing it. An example of the acceleration signals for a circular gesture is showed in figure 2.

III. Gesture Recognition using Prediction-Error-Classification

The schematic architecture of Prediction-Error-Classification system is shown in figure 3. This system is composed of N prediction-error blocks, one for each class (as the one represented in figure 4), and a comparison block that compares the prediction errors to decide the type of the gesture. Each block contains: a memory block (in order to delay one time step the input signal), a predictor block (that is the core of the method and can be seen in figure 5), and an error block (that is used to calculate the mean error of each gesture).

[1]The acceleration sensor used in this work is the module Witilt v2.5 provided by Sparkfun electronics http://www.sparkfun.com

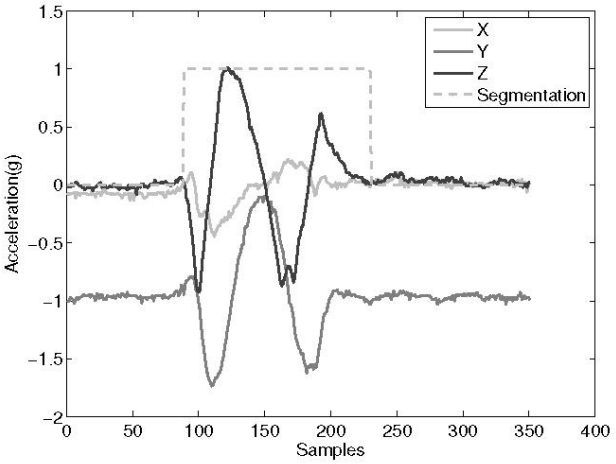

Fig. 2. Acceleration signals recorded at the hand when performing a circular hand motion.

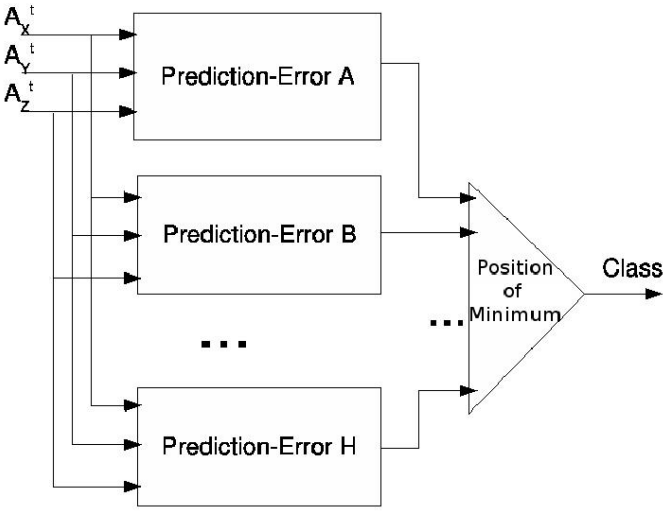

Fig. 3. Prediction-Error-Classification System

To recognize gestures we create one Prediction-Error block for each class, so, if there are N classes, there will be N blocks. During the recognition process the acceleration signal is fed to all the blocks and the gesture prediction error for each class is calculated, and then by comparing the errors the class of the gesture is chosen.

A. Prediction-Error Block

The input to the predictor block are the acceleration values in the previous time step $A^{t-1}_{[X,Y,Z]}$ and the output is the predicted value for next step $P^{t}_{[X,Y,Z]}$. As stated in introduction, this predictor block (see figure 5) is implemented using Neuro Fuzzy Systems. In particular, the Matlab implementation of ANFIS [6] is used to create the predictors. **ANFIS is a well known tool that has been used in many**

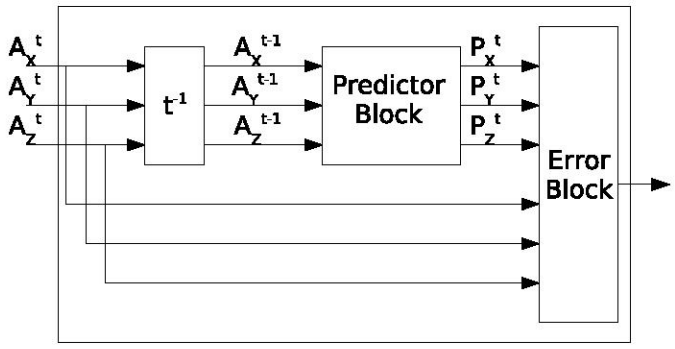

Fig. 4. Prediction-Error Block

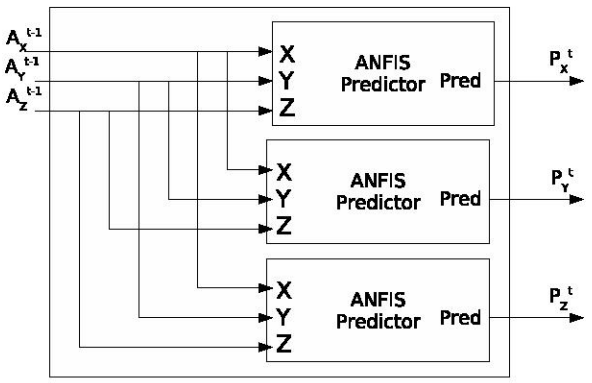

Fig. 5. Predictor Block

prediction applications, even to predict chaotic series [1]. These kind of predictors can produce only one output, so to predict the 3D acceleration vector three fuzzy rule systems are needed.

ANFIS is a neural network implementation of a Takagi-Sugeno (T-S) fuzzy inference system. The antecedents of rules of T-S systems are linguistic variables, while their consequents are functions of the input variables. These functions are usually first-order or zero-order polynomials. In this work we have used first-order so each rule of this predictor has the form:

IF $(A_X^{t-1}$ is $X_i)$ AND $(A_Y^{t-1}$ is $Y_j)$ AND $(A_Z^{t-1}$ is $Z_k)$ THEN $f_n = a_n\ A_X^{t-1} + b_n\ A_Y^{t-1} + c_n\ A_Z^{t-1} + d_n$

Where:

- A_X^{t-1}, A_Y^{t-1} and A_Z^{t-1} are the input values of acceleration signal
- $X_{1..3}$, $Y_{1..3}$, $Z_{1..3}$ are the linguistic labels for each input.
- a_n, b_n, c_n, d_n are coefficients of the linear combination of inputs.

Mirar comentario del segundo reviewer sobre misleading verbatim

For each training set we have trained three fuzzy rule systems using the ANFIS hybrid algorithm which integrates back-propagation and least square estimation. As initial parameters for ANFIS we have set the following: each input has three linguistic labels, which are represented by Gaussian functions (see figure 6). Therefore, each predictor has 27 rules, and the predicted output will be obtained from the weighted average of the consequents of the activated rules. All the parameters of this rules and the linguistic labels will be adjusted by the algorithm during training to minimize the prediction error.

The error block allows us to measure how good is the prediction produced by the predictor. For that, the error block calculates the error between the real signals and the predicted ones. In particular for this work, the error measurement for a sample is the mean absolute error of each axis of the signal, computed with the following equation:

$$Pred\_Error(t) = \sum_{i}^{X,Y,Z} |P(t)_i - V(t)_i|/3 \qquad (1)$$

This is the prediction error for only one sample so, to obtain the prediction error for one gesture we calculate the mean error of all its samples.

$$Pred\_Gesture\_Error = \sum_{t=1}^{T} Pred\_Error(t)/T \qquad (2)$$

Where T is the number of samples of the gesture.

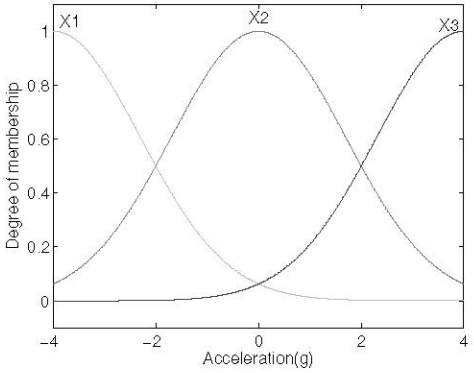

Fig. 6. Membership functions for input X

IV. Experiments

A set of eight different gestures represented in figure 7 has been used to test the accuracy of the signal predictors and the performance of the recognition method. The begin and the end of each gesture is marked with a circle and an arrow, respectively. All gestures were performed vertically.

For this work, two different experiments were done, they were performed by one person that has the sensor on right hand and a wireless mouse held with the left hand to segment the gestures. In each experiment a dataset with twenty instances of each gesture class were recorded, therefore contains 160 gestures.

V. Analysis of the Prediction-Error block for prediction

The objective of the first experiment is to check the accuracy of the signal predictors and the viability of using them

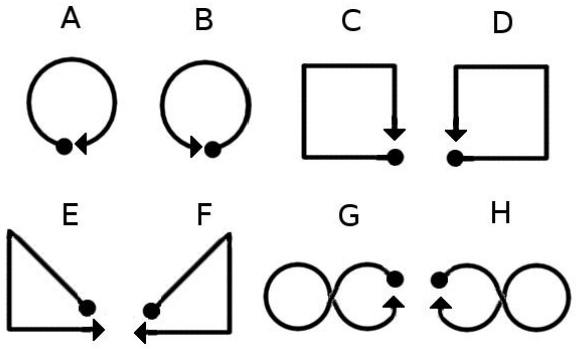

Fig. 7. Gestures used to analized the performance of the method

for gesture recognition. In this experiment the conditions were controlled: the person that performed all gestures was sat during the whole process, and gestures were isolated, that is, between one gesture and the next his hand rested in the same position.

For the training phase some samples of each gesture were chosen and the rest were left for testing. In order to reduce the dependence between results and gestures used for training, we have chosen the training instances randomly and repeated this process 20 times. To check the impact of the size of the training set in the accuracy of the predictors, we have varied the size of the training set from 5% to 50% by increments of 5%, that means, that we have done 200 repetitions for each dataset.

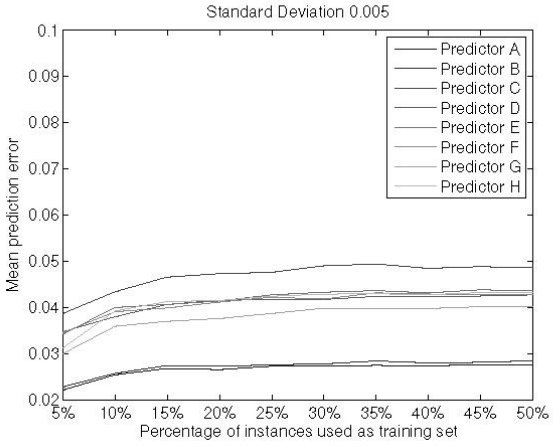

Fig. 8. Mean prediction errors for the training instances using training sets with different number of instances

Using this dataset, we have analyzed how the prediction errors behave for each gesture class when the number of instances of the training set are increased. Figure 8 shows the mean prediction errors produced by each predictor when forecast the next values of training instances. These means are calculated by averaging the errors of the 20 random training sets. On the other hand, figure 9 shows the mean prediction errors when predictors forecast the next values of testing instances.

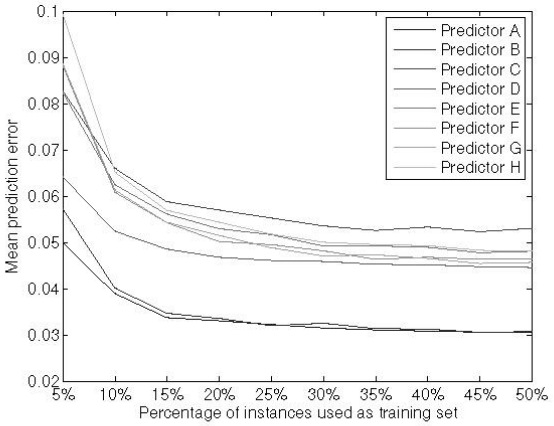

Fig. 9. Mean prediction errors for the testing instances using training sets with different number of instances

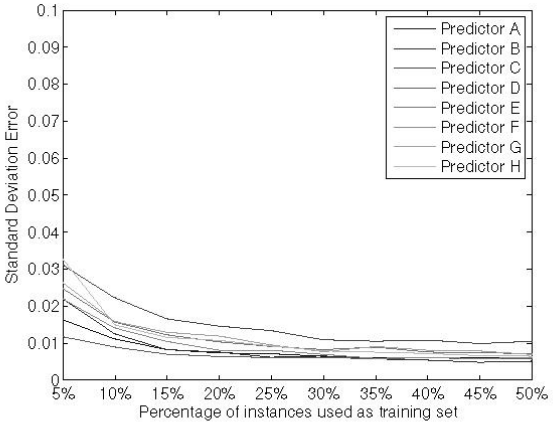

Fig. 10. Standard deviation of prediction errors for the testing instances using training sets with different number of instances

In figure 8, it can be seen that the prediction error for the training instances increases with the size of training sets. This is due to the over-fitting produced on predictors when the number of samples is too small (that is, 1 or 2). The generalization capability of the predictors can be observed in figure 9, in this case the prediction error decreases with the increase of size of training sets.**A low average is important but this only can be seen as a good result if the variability of the prediction error also decreases as it is shown in figure 10**.From these figures we can say that with more than 20% of instances for training the prediction errors tend to stabilize, and therefore 4 samples are enough to create good predictors.

A. Comparison of prediction errors

The next step is to evaluate if these signal predictors can be used to classify gestures. This classification is based on the hypothesis that the predictors trained with gestures of one class will produce better predictions for gestures of this class than other predictors trained with gestures of other classes. For example figure 11 shows how the errors of the predictors

evolve for a specific gesture of class A. It can be seen that although initially is difficult to decide which predictor has the lowest error, after some samples the error of predictor A is clearly smaller than the others.

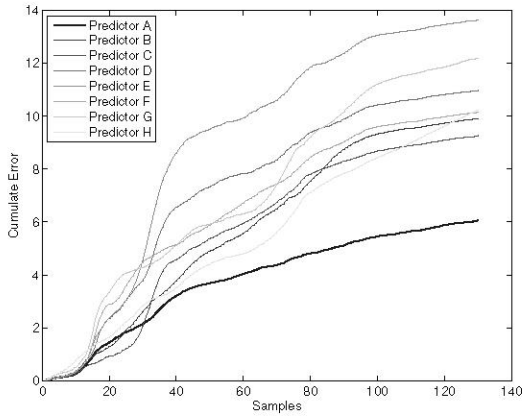

Fig. 11. Cumulate prediction error for a given gesture of class A

To test this hypothesis for all gesture classes, we have compared the prediction errors produced by each predictor for all classes. For this test we are going to set the size of the training set to 20% since in previous section it was shown that this size is good enough. **Explicar mejor los resultados de esta grafica**
Figure 14 shows the results of this comparison, and it can be seen that the predictor of the current class has always the lowest error, and therefore, this way of doing classification is possible.

VI. Analysis of the PEC system for classification

In order to evaluate the performance of the proposed method to classify gestures, we have used the dataset of the first experiment. We have chosen the training instances randomly and repeated the classification process 20 times and average the classification rates. To check the impact of the size of the training set in the accuracy of the classification, we have again varied the size of the training set from 5% to 50% by increments of 5%.

In all cases, independently of the size of the training set and for all the classes, the recognition rate for the training gestures is 100%, so we do not include that figure. Nevertheless for testing gestures the rates vary as we can see in figure 12. It can be observed that average recognition rate is pretty high 96% for a training set of 20% (4 instances) and reaches 99% when using more than 35% (7 instances) for training set. Also it can be observed that the recognition rate varies depending on the class, and class F has initially a lower classification rate than the others but it converges after using more than 35% of training samples.

A. Robustness Analysis

The objective of the second experiment is to test the robustness and the accuracy of this approach in a more realistic environment. In this experiment the person moved around the room in an unconstrained way. He was performing different activities (like sitting, standing up, reading books, opening drawers), while was performing gestures at random instants. Furthermore, there was none rest posture between two following gestures.

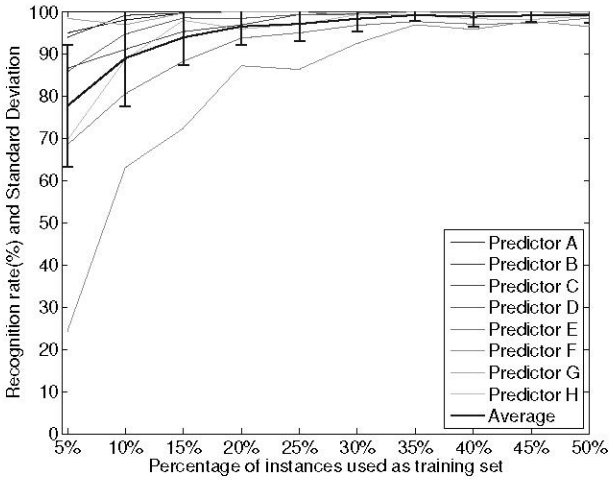

Fig. 12. Recognition rates for testing set of isolated gestures using different number of training instances

Again, the recognition rate for the training gestures is 100% independently of the size of the training set and for all the classes. Now, the recognition rates for the testing gestures is a little bit lower than in the previous one as we can see in figure 13. It can be observed that the average recognition rate is still high 93% for a training set of 20% (4 instances) and reaches 96% when using more than 35% (7 instances) for training set. In this case, the lowest recognition rates are also obtained by the class F and class E that correspond with the triangle gestures. This could mean that these gestures are more difficult to learn than the others.

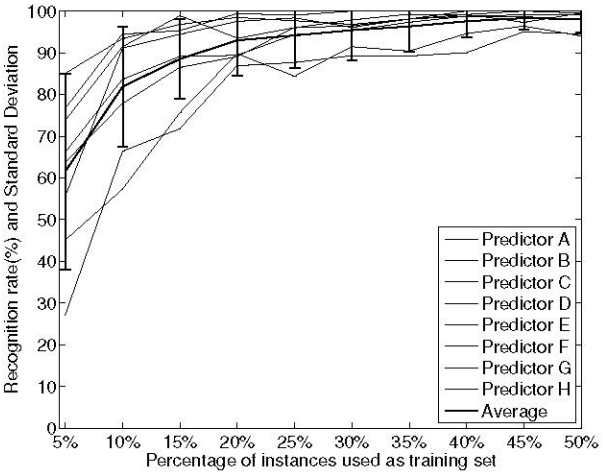

Fig. 13. Recognition rates for testing set of gestures made in a realistic environment using different number of training instances

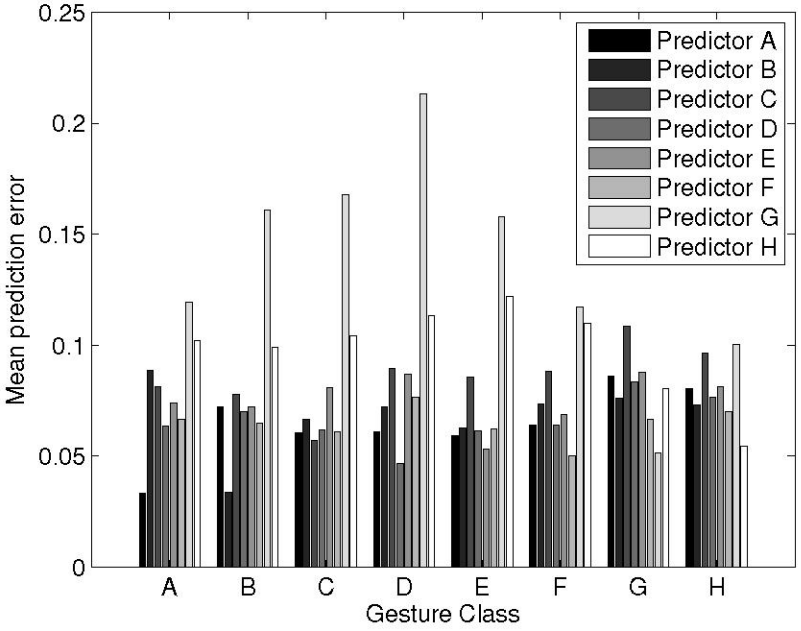

Fig. 14. Mean prediction errors for the testing instances for each predictor

VII. Conclusions and future work

In section V we have checked the accuracy of the signal predictors and proof the viability of this approach for gesture recognition. We have also validate the hypothesis that a predictor of one class predicts more accurately the values of the gestures of its class than the other predictors.

We have also studied the impact of the number of training samples in the prediction error, and the impact of the number of training samples in the recognition rate, using in both cases cross-validation.

As we expected, the recognition rates for realistic gestures are a little bit lower than for isolated gestures. However the recognition rate is still very high and that means that this approach is quite robust and can be used in real situations. Also we want to remark that this method is quite fast, robust and scalable, since it only takes 1 second to learn each class and its execution time is negligible. **(Otra ventaja del primer review) On the other hand, this method presents an interesting advantage against others methods like HMM,... because the raw signal is fed directly to the predictor and therefore it is not necessary any normalization stage of the signal.**

Furthermore, this method shows a better performance than our previous work with neural networks[2]. Using continuous time recurrent neural networks trained with genetic algorithms similar results were obtained for gestures captured in a constrained environment. However the recognition rate for gestures in a realistic environment was quite lower about (65%) using the 20% of instances as training set while in this case is over (90%).

As future work, a study of recognition performance with different people is needed in order to ensure its success as a general gesture interface. **Comentario del segundo reviewer: Que pasa si hacemos gestos con diferentes velocidades. Specially a new dataset with gestures performed at different speeds will be necessary to test how the method deals with this variability. The acceleration signals of a gesture depend directly on the speed of the gesture.**
Furthermore, in these experiments, an automatic segmentation was not used because we could not discriminate between gestures and other activities in the second experiment. And therefore, the gestures were segmented manually by pressing the button of the wireless mouse. But in future work we also plan to study how to do it automatically.

VIII. Acknowledgements

This work has been partially supported by MEC(Spain) under project TIN2005-08943-C02-01

References

[1] *Predicting chaotic time series with fuzzy if-then rules*, 1993.
[2] G. Bailador, D. Roggen, G. Troester, and G. Trivio. Real time gesture recognition using continuous time recurrent neural networks. In *Bodynets and Networks*. ACM, June 2007.
[3] G. Bailador, G. Trivino, and S. Guadarrama. Gesture recognition using a neuro-fuzzy predictor. In *International Conference of Artificial Intelligence and Soft Computing*. Acta press, 2006.
[4] G. S. Chambers, S. Venkatesh, and G. A. W. West. *Automatic Labeling of Sports Video Using Umpire Gesture Recognition*. 2004.
[5] T. Frantti and S. Kallio. Expert system for gesture recognition in terminal's user interface. *Expert Systems with Applications*, 26(2):189–202, February 2004.
[6] J.-S. R. Jang. Anfis: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23:665–684, 1993.

[7] C. F. Juang and K.-C. Ku. A recurrent fuzzy network for fuzzy temporal sequence processing and gesture recognition. *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 35(4):646–658, 2005.
[8] H. Kang, C. W. Lee, and K. Jung. Recognition-based gesture spotting in video games. *Pattern Recognition Letters*, 25(15):1701–1714, 2004.
[9] P. Keir, J. Payne, J. Elgoyhen, M. Horner, M. Naef, and P. Anderson. Gesture-recognition with non-referenced tracking. pages 151–158, 2006.
[10] P. Lukowicz, J. A. Ward, H. Junker, M. Stäger, G. Tröster, A. Atrash, and T. Starner. *Recognizing Workshop Activity Using Body Worn Microphones and Accelerometers*. 2004.
[11] J. Mäntyjärvi, J. Kela, P. Korpipää, and S. Kallio. Enabling fast and effortless customisation in accelerometer based gesture interaction. In *MUM '04: Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia*, pages 25–31, New York, NY, USA, 2004. ACM Press.
[12] T. Pylvänäinen. *Accelerometer Based Gesture Recognition Using Continuous HMMs*. 2005.
[13] G. S. Schmidt and D. H. House. Towards model-based gesture recognition. In *FG '00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, Washington, DC, USA, 2000. IEEE Computer Society.
[14] C. Verplaetse. Inertial proprioceptive devices: self-motion-sensing toys and tools. *IBM Syst. J.*, 35(3-4):639–650, 1996.