Automatic Light Control in Domotics using Artificial Neural Networks

Carlos Machado and José A. Mendes

Abstract— Home Automation is a field that, among other subjects, is concerned with the comfort, security and energy requirements of private homes. The configuration of automatic functions in this type of houses is not always simple to its inhabitants requiring the initial setup and regular adjustments. In this work, the ubiquitous computing system vision is used, where the users' action patterns are captured, recorded and used to create the context-awareness that allows the self-configuration of the home automation system. The system will try to free the users from setup adjustments as the home tries to adapt to its inhabitants' real habits. In this paper it is described a completely automated process to determine the light state and act on them, taking in account the users' daily habits. Artificial Neural Network (ANN) is used as a pattern recognition method, classifying for each moment the light state. The work presented uses data from a real house where a family is actually living.

Keywords—ANN, Home Automation, Neural Systems, Pattern Recognition, Ubiquitous Computing.

I. INTRODUCTION

Automation in houses is becoming more and more available as a standard feature of a normal home. The creation of real smart houses that gets adapted to its inhabitants' usage in an efficient way, is an essential characteristic of the future home automation systems.

The ubiquitous computing vision can be applied in a home automation scenario, offering the ability of the house to react to the users and learn their habits. Ubiquitous computing systems, raised by Mark Weiser, have the purpose of creating applications that react to the situations, help in everyday tasks and make the technology usage effortlessly [1].

In the last years some dedicated smart environments are being developed, by academia and industry, around the world, e.g., AwareHome [2], intending to create a living laboratory for ubiquitous computing research on everyday activities; the MavHome project [3] is a multi-disciplinary research focused on the creation of an intelligent home environment from

Carlos Machado. is with the Department of Industrial Electronics, University of Minho, Campus de Azurém, 4800-058 Guimarães, Portugal (phone: +351 253 040952; fax: +351 253 510189; e-mail: carlos.machado@ dei.uminho.pt; url: http://cmachado.pt.vu/).

José A. Mendes. is with the Department of Industrial Electronics, University of Minho, Campus de Azurém, 4800-058 Guimarães, Portugal (e-mail: jose.mendes@dei.uminho.pt).

University of Texas at Arlington; from industry, the Philips company established HomeLab [4] as a testing ground advanced interaction technology; a collaborative product and technology development, PlaceLab project [5], is joint initiative between Massachusetts Institute of Technology (MIT) and TIAX firm, provides a living laboratory to study human behaviours, the routine activities and interactions of everyday life. The recognition of Activities of Daily Life (ADL), is focused almost by all researches in order to achieve the high context definitions to be used by other high level applications.

In recent years there has been growing the usage of diverse datamining techniques to process data captured from a collection of sensors dispersed in the environment, aiming the determination of pattern in the human activity, e.g., in [6] the usage of a hierarchical hidden semi-Markov model to track the daily activities of residents in an assisted living community; Tapia et al. [7] applied Navie Bayesian classifier to recognize ADL such as bathing, toileting, dressing and preparing lunch, based on analysis of data collected from a set of small and simple state-change sensors; the usage of a temporal neural network-based agent [8] to recognize behaviours and ADL such listening to music, working at computer and sleeping; they utilize a user interface (UI) to help to label the user activities; Lithr et al. [9] uses a data mining approach, intertransaction association rule (IAR), for the detection of new and changing behaviour in people living in a smart home; in [10] it used a self-adaptive neural network (SANN) called Growing Self-Organizing Maps (GSOM), a cluster analysis of human activities of daily living.

In this paper, it is presented a complete autonomous method of using ANN to determine the users' patterns concerning the usage of the lights with the purpose of predicting its state and act on them automatically. The method present will not require any users' direct input by UI. It only uses the patterns acquired from the normal usage of the lights from the users and, therefore, it doesn't require the task labeling step. After ANN training, the weights are small enough to be distributed throughout environment embedded devices. The structure of the ANN was kept as simple as possible so that it can run later in low resource devices.

The entire essential steps to reproduce this work will be detailed as must as possible. However not all available data will be analyzed or processed due to privacy and/or security reasons.

The paper starts by presenting the system and the way the data is acquired, followed by data preprocessing and normalization. The ANN structure and the details of training are also shown as well as the evaluation of the results. Finally, conclusions and future work are presented.

II. 2. DATA ACQUISITION

Data acquisition is the first step of a pattern recognition system as described by Polikar [11].

The data was collected from a real instrumented house, where a young family, composed by a couple and a child, is actually living. The system is monitoring the complete house lights usage, 24 hours a day, since August 2007.
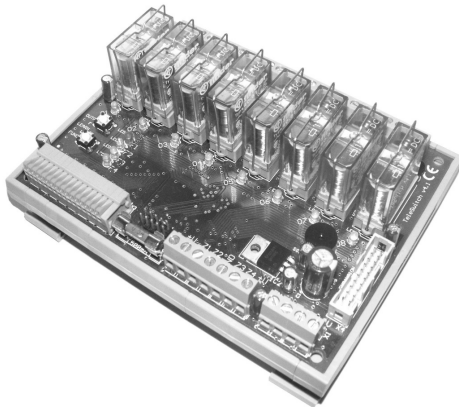

Fig. 1 Switch device with relay output and RS-485 interface.

In this application, it is important that the acquisition of the users' actions is done in an implicit way, i.e., in such a manner that the habitants do not notice this process [12]. This aspect was considered by the hardware, shown in fig. 1, which controls the instrumented house. This device was designed to work in a distributed network and take into account some characteristics that allow it to be included in an autonomic ubiquitous system as described in [13]. It is placed in the home electrical installation between the light switch and the light bulb, reproducing the normal circuit functionality and, at same time, reporting the user actions events. An actuation can also be requested remotely to the device.

TABLE I
EVENTS SAMPLE WHEN ACTING IN A LIGHT BULB

Event Time Stamp	Device Address	IO Id	IO Type	Action
18-03-2008 6:23	18	3	Input	Key Pressed
18-03-2008 6:23	16	130	Output	Relay On
18-03-2008 6:39	18	3	Input	Key Released
18-03-2008 6:39	16	130	Output	Relay Off

The events are captured by the teleswitch device, and then they are transmitted to a communicator device (the gateway of the dedicated home network) where they are time stamped [13]. Finally, the events are stored in a database. The table I shows an example of the events that report the users' actions and the actuator actions. Typically, to every input event, an output event is generated. The first event is the user's input and the second event is the device resulting action.
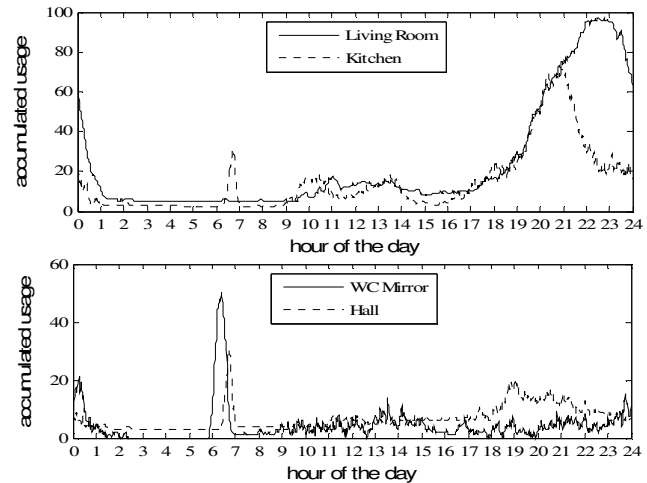

Fig. 2 Light Accumulated Usage with minute resolution (122 days period)

The events in the database can be used to calculate several statistics, as a training set to machine learning or in data-mining algorithms. In fig. 2 it is presented the light accumulated usage charts of four spaces selected from the instrumented house. In the kitchen light usage, we can clearly see that the dinner is served around 8 p.m. (on average) and after that, the family goes to the living-room. The hall light also indicates that its usage increases around 7 p.m., possibly because that is the home arriving hour. Another accumulated value presented in the kitchen, WC mirror and hall around 6 a.m. to 7 a.m. is very noticeable. It indicates that a very precise and timed event occurs almost every day. This feature can be easily identified by the get-up and go-to-work routine. The accumulated value in the WC mirror light starts at 6 a.m. and in the kitchen and Hall it starts around 6:30 a.m. and ends around 7 a.m..
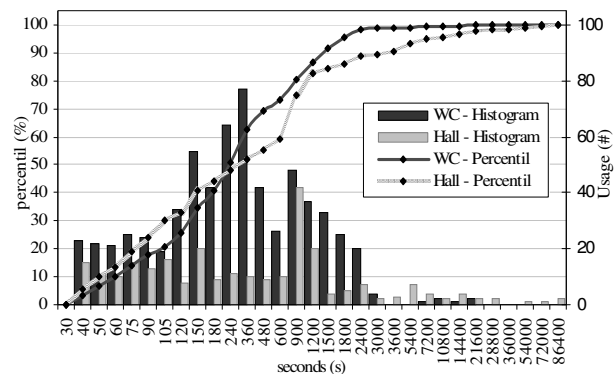

Fig. 3 Histograms for light usage intervals (122 days)

The histograms shown in fig. 3 point out the usage intervals in which a light is switched-on by the inhabitants.

These histograms are very important to determine the moment it is probable to switch-off a light. It's possible to see that 90% of the usage intervals for the WC mirror light has a duration of less that 1500s (25 minutes) and almost near 100% of the activities happen in less that 3600s (1 hour). In the hall light usage, the percentile of 90% happens at 1 hour and the value near 100% only appears later on. This indicates that an automatic decision to turn-off the light needs to take into account the user patterns to prevent a wrong action.

The light usage intervals histogram values below 30s were filtered because they have less impact in decision making and in light power consumption.

III. DATA SET

The data captured indicates that the light usages are somehow related and follow a pattern. This means that it may be possible to predict the state of light bulb by looking back to the previous events. It's also necessary to take into account the hour of the day and the weekday to improve the prediction. The weekday will help by separating the user pattern of a normal working day from that of the weekend.

The dataset needs to be prepared before we can use it to train the ANN. The function of the ANN is to classify the state of the light we are trying to predict and control. As input vectors, we need to supply it with cases when the light state is ON and cases when it is OFF.

A. Pre-Processing

The first step is creating a record set to every light source where, for each date, we identify the light state and the time of the state change, i.e. a record set of tokens of {light state (s); time passed from the last occurrence (t)}. The following step is filling in the blanks by expanding the data to get a record for each minute of the day for the time interval selected.

In the implementation we get a light ON state ratio of 1:49 for the hall light, but a better ratio 1:6 for the living-room light. The ratio depends on the amount of time a light is used in the data interval selected for processing.

Finally and before the normalization procedure, the dataset needs to be converted from the token into a single value. The equation 1 presented will do that transformation where τ indicates the max time an event will be considered.

$$v(s,t) = \begin{cases} (\tau - t) \times (2s - 1) & (\tau - t) \geq 0 \\ 0 & (\tau - t) < 0 \end{cases} \qquad (1)$$

The result dataset of the equation will have fading values in the interval $]0..\tau]$ for lights that were deactivated and $[-\tau..0[$ for lights that were activated in the last τ time. Using this equation, it will also prevent that the light left ON does not disturb the patterns detection that may happen later on.

In the implementation we select a $\tau = 60$ minutes, which means that we are looking to patterns that consider an action

occurred max of 60 minutes ago.

B. Normalization

The normalization process is very important because it will speed up the training phase [14]. There are many methods for data normalization: min-max normalization, z-score normalization and normalization by decimal scaling. In this implementation we selected the min-max normalization, because the minimum and maximum are well-known. Min-max normalization performs a linear transformation on the original data. If we consider that min_a and max_a are the minimum and the maximum values for attribute A, the min-max normalization maps a value v of A to v' in the range $[j, k]$ by computing the equation 2.

$$v'(v) = \frac{v - min_a}{max_a - min_a} \times (k - j) + j \qquad j < k \qquad (2)$$

In the dataset the values will be in the interval $[-\tau..\tau]$ and will be transformed into the interval $[-1..1]$, simplifying the min-max calculation and resulting in the equation 3.

$$v'(v) = v.\tau^{-1} \qquad (3)$$

The time interval (minutes of the day) was also normalized using min-max normalization, converting linearly the interval $[0..1439]$ to values within $[0..1]$ and the same normalization was done in the weekday attribute.

IV. ARTIFICIAL NEURAL NETWORKS (ANN)

In order to obtain automatic classification of light state, we applied artificial neural networks (feed-forward full connectionist, with sigmoid activation rule)[15] in the lights usage dataset described in the previous section.

Every input vector will contain the time, the day of the week and the last event time for each light in the house, except the light that we are trying to predict. The value of the light state we are trying to predict will be used to classify the datasets.

A. Structure

The ANN used is composed by three layers (input layer, hidden layer and output layer). The input layer has 26 neurons (one for each of the 24 light sources and one input neuron for the time attribute and other for the weekday attribute). The hidden layer uses the same number of neurons as the input layer. The numbers of the hidden layer neurons can not be too small, because it will decrease the capability for the ANN learn patterns, but if the number of the hidden neuron is too large, there is a risk of overfitting the dataset [16]. The output layer is composed by a unique neuron.

The output y of the single ANN layer is calculated in the equation 4.

$$y = f(Wx + b) \qquad (4)$$

The W represents the matrix layer neurons weights, the x is the input vector, b is the neurons bias weights, and $f()$ is the

activation function.

As activation function, we use the sigmoid function showed on equation 5.

$$f(x) = 1/(1 - e^{-x}) \qquad (5)$$

B. Training

The backpropagation algorithm with momentum was used for ANN classifier training. On the algorithm iteration, the single layer matrix W update is defined in equation 6 where α is the momentum, η is the learning rule and δ is the neuron error.

$$\Delta W_{ij}(t) = (1 - \alpha)\eta x\delta + \alpha\Delta W_{ij}(t-1) \qquad (6)$$

As a way of evaluating the binary classification, we use the Sensitivity (*Se*) and the Specificity (*Sp*) statistical measures.

The Sensitivity or recall rate specifies the portion of true positives (*TP*) in the classification, i.e., the portion of the light ON state that was positively detected. The *Se* is calculated using the equation 7, where *TP* (true positives) is the number of positive cases correctly classified and *FN* (false negatives) is the number of misclassifications for positive cases being incorrectly classified as negative.

$$Se = TP/(TP + FN) \qquad (7)$$

The Specificity is the proportion of true negatives (*TN*) in all negative cases in the dataset, i.e., the fraction of the light OFF state that was correctly detected. The *Sp* is determined by calculating equation 8, where *TN* (true negative) is the number of correct classification of the light OFF state and *FP* (false positive) is the number of misclassifications for negative where it was incorrectly classified as a ON state of the light.

$$Sp = TN/(TN + FP) \qquad (8)$$

In the classification of the light ON state, the *Sp* needs to be as high as possible because a *FP* (false positive) is unacceptable in a home environment since it will activate a light when it´s not the right time to do that. A low value of *Se* is not so critical because we can cross-check it with light usage intervals histogram before executing an action.

V. EVALUATION AND RESULTS

As an evaluation we tested the ANN classifier to determine the state of the hall light in the instrumented home. In this particular light, and for the interval analyzed, we got a total of 178028 records from which 3592 record are ON state and 174436 are OFF state for the hall light (ratio of 1:49).

The ANN took 13 minutes to train in a 1.5 Ghz cpu and converged in 33 epochs for a mean square error (*mse*) threshold of 0.025. The training stopped after a 5 consecutive epoch below the *mse* threshold.

TABLE II
STATISTICAL MEASURES FOR ANN CLASSIFICATION OF HALL LIGHT STATE

	Sensitivity *Se*	Specificity *Sp*	Accuracy *Ac*
Training Set	93.95 %	96.78 %	96.72 %
Validation Set	84.63 %	96.64 %	96.40 %
Test Set	85.88 %	96.46 %	96.25 %

In the table II, the *Se* indicates that the ANN can predict more that 80% of the day real light ON states, which means that these states are somehow related with other previous actions and the pattern was learned by the ANN.

The Accuracy (*Ac*) and the *Sp* show a large value (96%) indicating that the ANN is working well, but because of the large number of the light OFF states records, the absolute number of false positives (*FP*) is still very high. In the fig. 4 we can see that the *FPV* (False Positives Vectors) appear almost everywhere the activity was reported.

One of the functions of the ANN is to try to generalize and that can explain some of the errors in classification because the data is based on the users patterns that do not always choose to turn on the light as they are used to. But the ANN has classification errors and the real problem is to identify them.
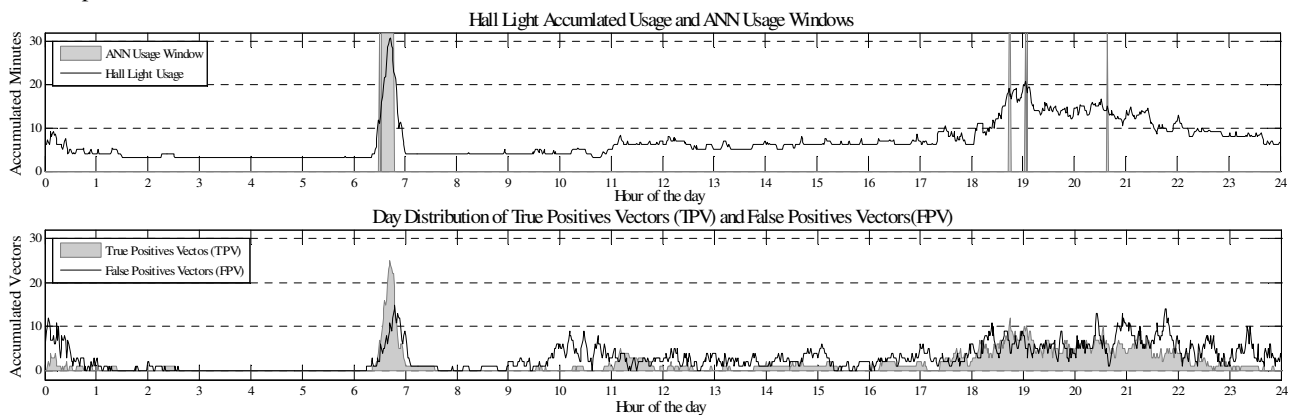

Fig. 4 Day Distribution of True Positives Vectors (*TPV*), False Positives Vectors (*FPV*),
Hall Light Accumulated Usage and ANN usage window ($\kappa = 4$)
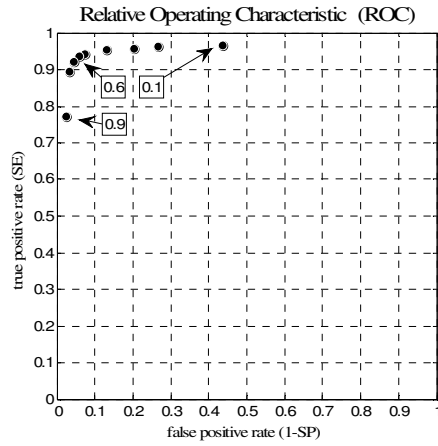
Relative Operating Characteristic (ROC)

Fig. 5 Relative Operation Characteristic (ROC) for ANN threshold

Fig. 5 shows the relative operation characteristic (ROC) for the ANN classifier, i.e., the classification evolution in function of the output threshold. We can see that the threshold of 0.6 will increase the results of the binary classification of both *SE* and *SP*. But in this particular application, a better compromise will be the threshold of 0.8, because the *SP* is greater and *SE* is still around 0.9.

A. Activating the light

In home automation scenario, the wrong actions from the system should be minimized. The usage of the ANN classification should be used only in the periods of the day that present a better performance. The performance indicator (P_m) of the classification can be calculated by the equation 9, where the TPV_m and FPV_m is the True Positive and False Positive Vectors respectively for a specific minute of the day.

$$P_m = TPV_m - FPV_m - \kappa \qquad (9)$$

A positive performance value indicates that there are more TPV_m than FPV_m and the ANN can be used on that minute. The constant κ introduces a bias to the performance indicator, allowing the setup of the level of performance.

In our implementation with $\kappa=4$, the ANN will have the usage windows showed in the figure 4. The ANN usage windows appear at the same time as the higher Hall Light accumulated usage, but in the minute 8:38 p.m. it's not that case. This may happen because it could be a new pattern or a pattern which only happens on some specific days.

In order to improve the processing and to limit the system actions, the call to the ANN classifier can be constrained to the moments where a user event occurs. This will guarantee that an action will always be a result of an event.

B. Deactivating the light

To deactivate the light, the utilization of the ANN should be complemented using the normalized accumulated usage ($a()$) and histogram percentile ($p()$) (see fig. 3). The equation 10 shows how to calculate a value of the function $D()$. In case it is above a threshold, it will point out the moment on which the ANN classifier can be invoked.

This function takes into account the duration of the usage ($p(t-t_0)$), the quantity of the usages for a particular time ($a(t)$) and the progress of the accumulated usage ($a'(t)$).

$$D(t) = p(t - t_0) - a(t) - \varphi.a'(t), \qquad \varphi \geq 0 \qquad (10)$$

The t represents the time of the day, the t_0 is the moment where the light was activated and the φ is a constant that defines the weight of the changes in the accumulated light usage.

The usage of the ANN classifier for deactivating the light will allow the threshold to be lower, resulting in higher power savings, as it may turn off the light earlier. At the same time, it takes into account the users' habits.

C. Spreading behaviors into environment

The process of training the ANN needs to be executed frequently in order to incorporate new patterns and disable older ones. The training process is executed everyday, by the house computer, at nigh hours, as a low priority process. The training process uses a window (122 days in our implementation) with the latest event data. The frequency of the training is sufficient to maintain the system updated.

After the training of ANN, for each light source, the information is sent to the embedded devices in the environment. These devices will monitor the events related with usage of the lights and execute the ANN classification, acting in the environment as already described. This way, the system will have lower latency and, at same time, we get a more robust system, i.e., if the house computer is down for some reason (blocked, rebooting or temporarily power-down) the environment automatic actions will still work. Any missing data caused by the failure of the home computer will not have a great impact in the system automatic action, because of the usage of ANN and basic statistical calculation, i.e., if a computer fails one day in an interval of 122 days less that 1% of data will be missing.

VI. FUTURE WORK

The method presented will be applied to control other devices in the home like: window shutter, heating system, media devices, etc. The stand-by state of devices will be focused to improve energy efficiency all over the home.

As a future work, the selection of a different and more complex ANN structure that makes possible a better *Sp* and *Se* will be studied, but it still needs to be simple enough to run in a small device. We will create a method to calculate the optimal values for the initial setup of the constants τ, κ and φ that will make the autonomous process more efficient.

Although the method shows good results, other A.I. techniques will be studied in order to compare its performance and limitations.

An UI is being implemented, that allows the users to see what automatic actions were taken by the system.

VII. CONCLUSIONS

The work presented in this paper shows that the usage of the ANN to determine user light patterns is feasible, by setting up, *a priori*, a small number of constant parameters. The technique exploits the users' monitored activity information and with that, it deduces the patterns in a complete automatic process. The results show that ANN can decide even when, as a training data, it only uses the lights last events.

The ability to generalization provided by the ANN is one of the major advantages of this technique, but special attention is required when using it. It is also important to point out that, in this scenario, automatic actions of the system should be very conservative and only act when a degree of confidence or performance in ANN is acquired. The combination of ANN algorithm and simple statistical data processing improves the automated process, by minimizing eventual errors from the ANN classification.

The ANN algorithm, versus other A.I. technique, like k-nearest neighbour algorithm (kNN) was used because it can reduce an events database with some megabyte in size to a few Kbytes composed by ANN weights. This compact information is ideal to, after training, distribute them throughout environment embedded devices, with reduced resources.

The ANN training and its consequent everyday distribution is sufficiently frequent to adapt for changes in users' behaviour caused e.g. by year seasons cycle. If the users change constantly and rapidly their behaviours, the system will not be able to detect patterns, resulting in an absence of automated actions.

The method presented offers the ability of a house to become context-aware, as it can learn the users' habits in an automatic process leading to an autonomic home with self-configuration capability.

ACKNOWLEDGMENT

REFERENCES

[1] M. Weiser."The Computer for the Twenty-First Century.", In Scientific American,Sept. 1991.

[2] C. D. Kidd, R. J. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. Maclntyre, E. Mynatt, T. E. Starner and W. Newstetter. "The Aware Home: A Living Laboratory for Ubiquitous Computing Research", Proc. of the Second International Workshop on Cooperative Buildings, 1999.

[3] D. J. Cook, M. Youngblood, III E. 0. Heierman, K. Gopalratnam, S. Rao, A. Litvin, and F. Khawaja, "MavHome: An agent-based smarthome", in Proc. of First IEEE International Conference on Pervasive Computing and Communications (PerCom'03), 2003, pp. 521-524.

[4] Philips Research - Technology. Avaliable: [August 2008], http://www.research.philips.com/technologies/misc/homelab/index.html

[5] S. S. Intille, K. Larson, J. S. Beaudin, J. Nawyn, E. Munguia Tapia, and P. Kaushik, "A living laboratory for the design and evaluation of ubiquitous computing technologies", in Proc. of Conference of Human Factors in Computing Systems, April 02-07, 2005, Portland, OR, USA, pp.1941-1944.

[6] H. Kautz, 0. Etzioni, D. Fox, and D. Weld, "Foundations of assisted cognition systems", University of Washington, Computer Science Department, Technical Report, Tech. Rep., 2003.

[7] E. Munguia Tapia, S. S. Intille, and K. Larson, "Activity recognition in the home setting using simple and ubiquitous sensors", in Proceedings of PERVASIVE 2004, vol. LNCS 3001, A. Ferscha and F. Mattern, Eds. Berlin Heidelberg: Springer-Verlag, 2004, pp. 158-175

[8] F. Rivera-Illingworth, V. Callaghan, and H. Hagras, "Automated Discovery of Human Activities inside Pervasive Living Spaces", Proc. of 1st International Symposium on Pervasive Computing and Applications, 2006, pp. 77- 82.

[9] S. Luihr, G. West, and S. Venkatesh, "Recognition of emergent human behaviour in a smart home: A data mining approach", Pervasive Mob. Comput. 3, 2, pp.95-116, 2007.

[10] Huiru Zheng, Haiying Wang,Norman Black, "Human Activity Detection in Smart Home Environment with Self-Adaptive Neural Networks", in IEEE International Conference Networking, Sensing and Control, 2008., (ICNSC 2008), ISBN: 978-1-4244-1685-1, pp 1505-1510

[11] Polikar R.,"Pattern Recognition, In Wiley Encyclopedia of Biomedical Engineering", Ed. Akay, M., New York, NY: Wiley., 2006.

[12] A. Schmidt, "Implicit Human-Computer Interaction through Context." In International Symposium on Human Computer Interaction with Mobile Devices and Services (MobileHCI), 1999.

[13] Carlos Machado, José Mendes, "Sensors, Actuators and Communicators When Building a Ubiquitous System" in Proc. of IEEE ISIE 2007, in Vigo, Spain, 2007, ISBN:1-4244-0755-9 (p1530-1535).

[14] L.A. Shalabi, Z. Shaaban, B. Kasasbeh,"Data Mining: A Preprocessing Engine.", Journal of Computer Science 2:9 (2006) 735-735.

[15] Gurney, K. (1997),"An Introduction to Neural Networks", London: Routledge. ISBN 1-85728-673-1

[16] Michael J. A. Berry, Gordon S. Linoff, "Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management", 2nd Ed. ISBN: 978-0-471-47064-9

[17] Altman DG, Bland JM (1994). "Diagnostic tests. 1: Sensitivity and specificity". BMJ 308 (6943): 1552. PMID 8019315.

Carlos Machado received his degree in electronics engineering from the Industrial Electronics Dept. of the University of Minho (1999). He received the MSc in Automation and Robotics Area from the Dept. of Industrial Electronics, University of Minho (2004). After that, he taught Computer Architecture and Digital Systems, on the Dept. of Informatics at University of Minho. Since January 2005, he is a PhD student doing the thesis in Autonomic Ubiquitous Systems using Artificial Intelligent Techniques on Industrial Electronics Department. at University of Minho, Portugal. URL: *http://cmachado.pt.vu/*

José A. Mendes has a degree in Electrical Engineering awarded by the University of Coimbra (Portugal) in 1989. In 2000, The University of Hull (UK) awarded a PhD degree in Electronic Engineering. Currently, he is a lecturer of Digital Systems and Microcontrollers at the University do Minho (Portugal), where his research has been focused on Embedded Systems.