2009 Ninth International Conference on Intelligent Systems Design and Applications

Sensor Fusion-oriented Fall Detection for Assistive Technologies Applications

Stefano Cagnoni, Guido Matrella, Monica Mordonini
Dipartimento di Ingegneria dell'Informazione
University of Parma
viale Usberti 181a, I-43124 Parma, Italy
{cagnoni,monica}@ce.unipr.it, guido.matrella@unipr.it

Federico Sassi, Luca Ascari
Henesis srl,
via Passo della Cisa 13, I-43100 Parma, Italy
{federico.sassi}, {luca.ascari}@henesis.eu

***Abstract*—A new trend in modern Assistive Technologies implies making extensive use of ICT to develop efficient and reliable "Ambient Intelligence" applications dedicated to disabled, elderly or frail people.**

In this paper we describe two fall detectors, based on bio-inspired algorithms. Such devices can either operate independently or be part of a modular and easily extensible architecture, able to manage different areas of an intelligent environment. In this case, effective data fusion can be achieved, thanks to the complementary nature of the sensors on which the detectors are based.

One device is based on vision and can be implemented on a standard FPGA programmable logic. It relies on a simplified version of the Particle Swarm Optimization algorithm. The other device under consideration is a wearable accelerometer-based fall detector, which relies on a recent soft-computing paradigm called Hierarchical Temporal Memories (HTMs).

I. Introduction

The number of elderly people who fall and are not injured or sustain minor or moderate injuries is substantially unknown, but is definitely very large. Recent researches estimated that each year, in the U.S., nearly 30% of elderly people incur in falls, and the likelihood of falling increases substantially with age. Falls may directly result in traumas, fractures, permanent disability, or even death. The injuries suffered as consequences of falls can impact strongly on the quality of life of older people, both from a physical and a psychological point of view. Falls can also significantly affect households and national healthcare systems, from both an economical and organizational point of view [1].

The situation of the U.S. is similar to that of all most industrialized countries; in particular, in Europe, the effects of ageing trace worrying scenarios for 2050 [2]. For this reason, many countries support scientific research aimed at finding technological solutions that optimize the costs of healthcare and increase quality of life of elderly, frail or partially autonomous people [3].

One of the most recent trends in "Ambient Intelligence" is to make extensive use of ICT to develop ever more efficient, reliable and economic Assistive Technologies [4], [5], [6]. This approach can be made most effective by integrating several technological aids within a single intelligent environment, in which large amounts of heterogeneous data can be processed. Data fusion can achieve much better and more robust results than single devices integrating simple and cheap devices in a cooperative approach.

Within this framework, automatic recognition of human body movements is a well-known problem which has been tackled in several ways. In [7], for example, knowledge about the kinematics of the movements under investigation has been included in the classifier. This approach yields very high accuracy in detecting such events but, at the same time, limits the type of events that can be addressed.

The mid/long-term goal of our research is to design intelligent systems which, while not interfering with the activities of everyday life, are able to detect events in a timely manner, or even possibly provide reliable predictions by which traumatic events may be anticipated and avoided.

This paper describes two fall detectors, based on bio-inspired algorithms, which can cooperate in a data-fusion oriented way. The two sensors can be integrated into a modular architecture to compensate for each other's limits, favoring the development of a harmonic, modular and easily extensible system able to manage different areas of the environment. The former is a video fall detector which can be implemented in hardware. This cheap embedded video sensor is based on a computationally light algorithm (Particle Swarm Optimization, PSO), that is easily implementable on a standard FPGA programmable logic [8]. The latter is a wearable accelerometer-based fall detector, based on a recent neural network paradigm called Hierarchical Temporal Memories (HTMs). HTM is a biologically-inspired computational paradigm which is specialized in discovering invariant patterns in spatial-temporal data [9].

In the following we describe these two devices and report results of a preliminary experimental evaluation of their potential. Finally, we give some specifications for a possible architecture that can integrate them into a simple hybrid remote sensor network.

II. Visual Sensor based on PSO

The first device we are considering is an embedded system able to analyze and process data locally. This sensor is expected to send a central supervision system only aggregated information and not the whole video stream, with clear

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.203

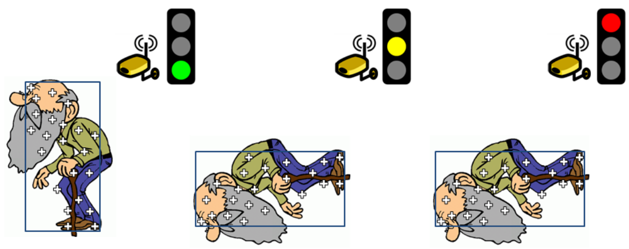


Figure 1. The sensor operating principle.

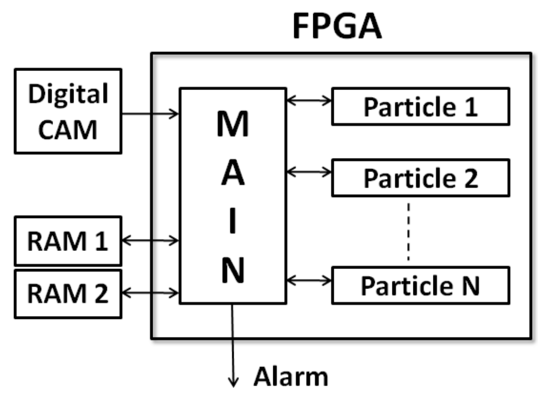


Figure 2. The architecture of the sensor.

advantages in terms of privacy and bandwidth occupation. The sensor output consists only of signals that account for the 'state of alert' on the potential occurrence of a fall. These data can describe 'levels of alert' as do traffic lights: green (normal state), yellow (alert), red (potential danger). When used in conjunction with other sensors (audio, wearable, etc.), such a compact, economical and little-invasive device can provide a description of the environment being monitored, which may be accurately evaluated by an ambient intelligence system as the one described in [10]. The operating principle of the sensor is illustrated in Fig. 1.

The detector is based on a variation of the PSO algorithm [11], which is implemented in hardware on an FPGA programmable logic. Simplified versions of PSO for hardware implementations have been described in other contexts [8], [12]. The device under cosideration can be seen as an embedded, compact and cheap implementation of PSO, customized to detect falls.

Its development takes into account how several computer vision problems can be reformulated in terms of the optimization of a task-dependent function whose value is higher near the objects, if any, which are to be detected in a scene. PSO is a bio-inspired optimization algorithm that searches for the optimum of a function (fitness-function), mimicking the behavior of flocks of birds in search of food (i.e., areas where the values of such a function are high). A set (swarm) of agents (particles) move within the search space (the domain of the function) seeking its extrema. The motion of each particle can be modeled by the following two simple difference equations that describe the position of each particle and its velocity in time:

$$P_n(t) = P_n(t-1) + v_n(t) \quad (1)$$

$$\begin{aligned} v_n(t) = \; & w * v_n(t-1) \\ & + c_1 * rand() * [BP_n - P_n(t-1)] \\ & + c_2 * rand() * [BPG - P_n(t-1)] \end{aligned} \quad (2)$$

where P_n is the position of the n^{th} particle, v_n its velocity, c_1 and c_2 are two positive constants, w is the so-called 'inertia weight', BP_n is the point with highest fitness visited so far by the n^{th} particle, and BPG is the highest-fitness point visited by any member of the swarm so far. The function $rand()$ returns a random value taken from a uniform distribution in the interval [0,1]. The value of the constants w, c_1 and c_2 must be chosen carefully to optimize convergence of the algorithm.

In our application, the particle swarm moves over the image acquired from the video sensor in search of points of interest. The fitness function is computed only for pixels that are actually visited by particles, and returns a value that is high where local visual features are similar to those which are being sought.

The hardware of the sensor includes a digital video camera, connected to a FPGA programmable logic, to which two RAMs are connected. Fig. 2 outlines the architecture of the sensor.

In an initial set-up phase, an image from the digital camera is taken as background (BG) and stored in RAM1. At runtime, the current picture (CF) is sent to the FPGA from the camera and stored in RAM2. The background is usually refreshed periodically, while no significant event is being detected, to compensate for possible variations of the environmental conditions (light intensity and direction, etc.)

The basic task to be accomplished by the swarm is to detect any moving person who enters the scene and spread over it as uniformly as possible, in order to outline its current position (standing, bending, lying on the floor, etc.).

The simplest method of detecting moving objects is based on the computation of local differences between corresponding pixels of the BG and the CF images (eq. 3).

$$\delta(P_n) = BG(P_n) - CF(P_n)$$
$$f(P_n) = \begin{cases} \delta(P_n) & \text{if } \delta(P_n) > threshold \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Basically, our fitness function checks if, at a given point in space, there is a significant difference between the reference background and the current frame, which would mean that someone has entered the scene since the last background refresh.

To detect a possible fall, the algorithm works as follows:

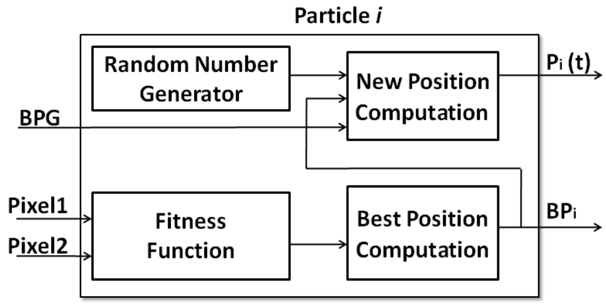


Figure 3. The architecture of a particle.

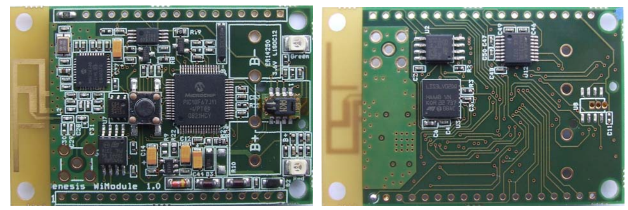


Figure 4. Front and rear sides of the electronic module used in the experiments.

Given K particles and N_{iter} iterations,

1) At time t=0, initialize the positions of the K particles $P_1(0), \ldots, P_K(0)$ randomly;
2) For $i = 1, K$ let $BP_i = P_i(0)$ and $BPG = max_i(BP_i)$;
3) For $t = 1, N_{iter}$
   For $j = 1, K$
   - Compute $P_j(t)$ and the corresponding fitness function $f(P_j)$
   - If $f(P_j) > f(BP_j)$ let $BP_j = P_j$
   - If $f(P_j) > f(BPG)$ let $BPG = P_j$

After N_{iter} iterations most particles of the swarm are expected to have spread over a region, which includes BPG, that features high fitness values. A bounding box can then be drawn that contains all high-fitness particles. When the aspect ratio of the rectangle changes from a stable vertical position (height > width) to a stable horizontal one (width > height), a fall is detected (see Fig. 3).

From a hardware point of view, the use of an FPGA can take advantage of the intrinsic parallelism of PSO as follows:

- The motion equations can be computed independently and in parallel for each particle, returning their next positions and fitness-function values;
- Using two RAMs allows one to access corresponding pixels of the background frame (BG) and of the current frame (CF) at the same time.

The algorithm has been first simulated using Matlab, then, by means of a VHDL description, we implemented the electronic circuit that performs PSO and all the control logic required for its operation. The simulations have been carried out on a back-annotated netlist, which corresponds to the last stage of FPGA implementation; this guarantees that performances are accurate estimates of the ones which can be obtained by an actual hardware implementation.

III. Wearable Wireless Sensor based on HTM

Recently, smaller and cheaper accelerometers that may be used as non-obtrusive continuous monitoring devices attached to one's body have become available, along with low-power wireless data transmission devices, able to stream data directly to a remote server for instant evaluation. A novel neural network paradigm, modeled after the human neocortex and aimed at recognizing spatial-temporal patterns, named Hierarchical Temporal Memory (HTM), has also attracted significant attention from researchers.

Combining these technologies allowed us to introduce a new type of sensory channel, based on a wireless triaxial acceleration sensor module mounted on one's body, which streams data to a server that is able to detect events, such as a fall, in real time.

As regards event detection and classification, our approach is deeply rooted in the assumption that for us, as humans, it is easy to understand motion patterns, even very noisy, by finding "invariants" [9] in the signals that our body generates. The final aim of this research is to build a system able to recognize and correctly classify different movements performed by different people, even in "noisy" situations.

In our method we do not add any a-priori knowledge about the movements we want to classify, besides the fact that they develop over time. This approach has already been successfully validated in a more limited environment where accelerometer data was sent to a IEEE 802.15.4-enabled PDA and then classified off-line using an HTM. The data collection platform consists of a low-cost wireless module, incorporating a triaxial high resolution accelerometer [1] and a IEEE 802.15.4 RF transmitter [2], which continuously sends data to a IEEE 802.15.4-enabled server that receives and evaluate accelerometer data in real-time.

The wireless sensor module was entirely designed and developed by Henesis s.r.l. and is being used for distributed sensing within Henesis WISnP [3].

From the point of view of software, HTM is a computational paradigm, inspired by the biological structure and algorithmic properties of the neocortex, which derives from a more general theory, called *Memory-Prediction Framework* [9]. A HTM is a hierarchical network of nodes where the sensory data enter at the bottom while the outputs of the network are the output of the top nodes, which represent

[1] ST LIS3LV02DL; BW: up to 640Hz; range: $\pm 2.0g$; max res: 1mg

[2] a.k.a. MAC level of a ZigBee network, transmission range outdoor: 100mt; indoor: 10mt (0dBm output power)

[3] WIreless Sensor Network Platform: http://wisnp.henesis.eu

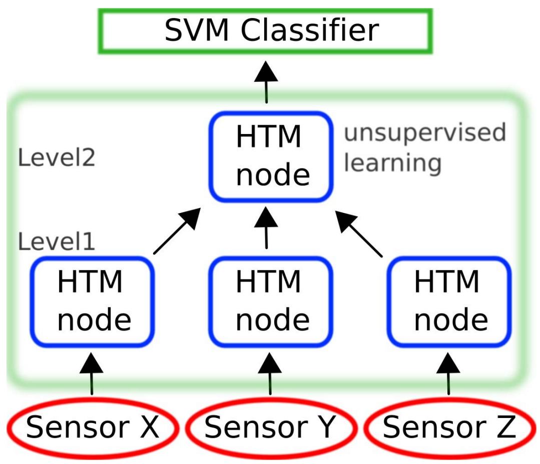
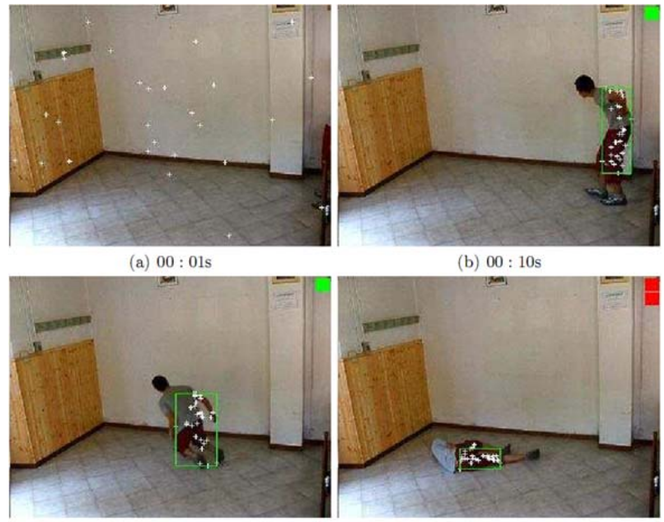


Figure 5. The HTM used in the experiments



(a) 00 : 01s (b) 00 : 10s (c) 00 : 15s (d) 00 : 22s

Figure 6. Some results of the simulation.

the possible *causes* of the input. Every node in the hierarchy runs the same algorithm, looking for spatial-temporal pattern (*invariants*) [13], [14] in its input and grouping them as *causes*. Every node is trained in an unsupervised manner, in the classical meaning, but *time* is considered to be the supervisor: if two events (inputs) occur often consecutively, they are expected to share the same cause (output). HTMs have been successfully used in vision and speech recognition problems.

In our experiment we firmly attached the wireless acceleration sensor to a medical chest-band and placed the sensor corresponding to the lower end of the sternum, to prevent it from hampering movements and to keep it stable over subsequent data acquisitions. This location, very close to the body center of mass, is ideal for acquiring accelerometer data for "whole body" movements.

The sampling rate of the wireless module, which is able to sample acceleration data at rates up to 640Hz, is set at 160Hz. The acquired data are transmitted to a server which implements an HTM and performs fall detection.

Data are processed by a 2-level HTM. In the first level there is one HTM node per axis of the accelerometer data, so every node in this level learn and recognize temporal patterns only of one particular axis. In the second level one single HTM node is fed by all three outputs from the first level nodes. This node runs the very same algorithm of the lower level with the sole difference that its inputs are already "sequences". Therefore this node processes "sequences of sequences". The output of the HTM network is eventually classified by a Support Vector Machine. Fig. 5 shows the HTM network.

This network has been trained on a dataset composed of 4 categories of movements: *S*tanding, Jumping, Walking On and Falling. The training set includes 10 events per category while the test set includes 20 events per category. Data sequences were recorded from three different volunteers. Every event lasts about 3 to 5 seconds.

IV. Experimental Results

This section reports the results of the tests we have performed to evaluate the performance of the detectors, considered as stand-alone devices.

A. Video-based Detector

Tests performed on a limited set of image sequences were more than satisfactory, both in terms of quality (see, as an example, Fig. 6) and in terms of computation time. In fact, in traditional computer vision applications, the whole difference image usually needs to be computed and then analysed by some global algorithm. With PSO, instead, differences are computed between a limited sample of corresponding pixels, thanks to the ability of PSO to rapidly converge onto the most 'relevant' parts of the search space.

Using this approach with a swarm of 11 particles which perform 1500 iterations, and processing an image of 320x240 pixels, 16500 pixels are evaluated per frame, instead of 76800 (=320x240). The average processing time (PSO + bounding box extraction) for a frame is approximately 5ms, allowing for real-time performance or for the introduction of more sophisticated post-processing. It should be pointed out that the resolution at which our tests have been performed can be considered a worst-case scenario for our algorithm, since the gain in term of image sampling ratio obviously becomes more and more favourable to our PSO approach as image size increases.

Evaluations and simulations have been carried out using a Xilinx S3E1200-FPGA device. Table I summarizes the percentage of use of the main FPGA resources.

Table I
ESTIMATED DEVICE UTILIZATION

FPGA resources	% of use
Slices	77%
Slice Flip Flops	23%
4-input LUTs	72%
RAM Block	7%
MULTIPLIERS (18X18)	82%

B. Accelerometer-based Detector

To simulate a real-life situation the classifier has been tested with a continuous flow of different events, without pre-segmenting them. The HTM classifies each time point; then, to obtain a more stable classification, the predicted and the expected output are averaged over time windows whose width is equal to 3 seconds (480 timepoints, which is about the average length of an event) in steps half a second long (80 timepoints). The event which occurs in the sliding window is labeled based on the class that has been detected more frequently by classifying each sample within the current window, both for the predicted and for the expected categories. The good results obtained on the test set are shown in Table II.

To explain the slightly worse performances on events labeled as Falls, it can be observed that the classifier might suffer from the potential co-occurrence of two uncorrelated events within the same time window. Falls are shorter events than the others and may be more sensitive to the above problem. While integrating this sensory channel inside a broader architecture, it may be useful to assign a degree of confidence to the HTM classification of the current input, for example by performing a fuzzy grouping and analysis of predicted and expected classifications inside the sliding windows.

V. DATA-FUSION ARCHITECTURE

Even if work about fall detection is common in the literature, and many patents, as well, have been filed on this topic, these systems are not common in daily geriatric practice. This is partly due to a natural resistance of people to being monitored but, above all, to the many "false alarms" that such devices can generate [15].

Among the main factors that could make such sensors more acceptable are:

- producing as output only the parameters that are strictly necessary for detection, preserving users' privacy;
- the use of passive sensors that can be located in the environment and kept in operation without requiring users' active participation;
- the use of wearable wireless sensors which, while possibly moderately constraining movements, remain active also within areas that can not be covered, for example, by video surveillance systems.

Table II
CONFUSION MATRIX FOR THE TEST SET.

Expected—Predicted	Other events	Fall
Other events	99.54%	0.46%
Fall	2.95%	97.05%

The problem of sensor reliability limits ICT services acceptance by people who are specialized in elderly assistance. Improving service reliability can be achieved by introducing some redundancy or synergy among sensors. This could also lead to offering even more important services, such as predicting (and avoiding) falls or dangerous situations. Such an integration can be achieved by defining a multi-level scalable architecture which offers the possibility to add more services (for example, fall detection with the ability to immediately locate the point where someone has fallen) and to integrate data from different sensors for the detection of a single critical condition using flexible and extensible models based, for example, on multiple classifier sets and/or fuzzy decision trees. Examples of integration can be found in [16] and in [17].

The two fall detectors taken into consideration are the first prototypes of devices that are to be integrated into a larger architecture of ICT services suitable for a non-invasive monitoring of elderly or partially-autonomous people during their daily activities. Both sensors interfere minimally with normal daily activities and the privacy of the patients while, jointly, being able to guarantee a good level of fault-tolerance, thanks to their complementary features. The video fall detector is completely passive and can be used without any user collaboration, while the other one is based on an accelerometer which is wearable. The use of both fall detectors increases the operability of an assistance service in situations in which one of them fails (as, for example, if the subject forgets to wear the former or when the camera of the latter cannot operate because the light is off or too dim).

Therefore, in designing services based on the fusion of data coming from our two sensors, different levels of warning can be defined, such as:

- **L0**: the two sensors are active and not detecting a fall;
- **L1**: only one sensor is active and is not detecting a fall;
- **L2**: both sensors are active and one is detecting a fall;
- **L3**: only one sensor is active and is detecting a fall;
- **L4**: both sensors are active and detecting a fall.

Even if detecting falls after their occurrence is important, it would be even more important if a service could recognize dangerous situations which could likely cause a fall. To achieve this goal, one can consider that both sensors are working on data sequences and that they are both based on adaptive algorithms. Therefore, one could maintain a history of data acquired before the system recognizes a user fall and learn to recognize such a situation.

Moreover, fall detectors can be integrated with other services such as monitoring of blood pressure, body temperature, heart rate, as well as sensors for people localization or for their tracking [18], [19]. Such developments would allow one to offer a global service that is more reliable and able to provide the inputs which allow predictions about falls to be made and, perhaps, to recognize dangerous behaviors of the subjects who are being monitored.

VI. Conclusions and Future Work

We have presented two fall sensors, an embedded video-based one and a wearable accelerometer-based one, which can be managed within a data-fusion-oriented framework, implementing policies aimed at maximizing system reliability and minimizing the presence of false alarms. The video fall detector is based on a digital camera and a FPGA programmable logic, able to locally process the images and to transmit to a server only aggregated information relating to the 'state of alert', with obvious advantages in terms of end users' privacy. The sensor uses a hardware implementation of the PSO algorithm, designed to exploit its intrinsic parallelism. The wearable sensor is based on a new powerful soft-computing paradigm which makes it possible to extend the task it performs to detecting a whole set of situations, and therefore to make the whole architecture more flexible.

Future developments of this work will be mostly aimed at implementing the solutions we anticipated for the data-fusion framework, with particular regard to increasing reliability and to predicting falls before they occur.

References

[1] "Self-reported falls and fall-related injuries among persons aged > 65 years - United States, 2006," 2008. [Online]. Available: http://www.cdc.gov/mmwr/preview/mmwrhtml/mm5709a1.htm

[2] "Ageing characterises the demographic perspectives of the european societies." [Online]. Available: http://epages.eurostat.ec.europa.eu/cache/ITY_OFFPUB/KS-SF-08-072/EN/KS-SF-08-072-EN.PDF

[3] "Europe is facing a demographic challenge. Ambient assisted living offers solutions." [Online]. Available: http://www.aal-europe.eu/Published/FinalVersion.pdf

[4] W. C. Mann, *Smart technology for aging, disability, and independence: the state of the science*. Wiley, 2005.

[5] G. M. Craddock, *Assistive technology: shaping the future : AAATE'03*. IOS Press, 2003.

[6] R. Felder and M. Alwan, *Eldercare technology for clinical practitioners*. Humana Press, 2008.

[7] S. Zhang, J. Ang, H. Marcelo, and oth., "Detection of activities by wireless sensors for daily life surveillance: eating and drinking," *Sensors*, vol. 9, no. 3, pp. 1499–1517, 2009.

[8] S. Mehmood, S. Cagnoni, M. Mordonini, and G. Matrella, "Hardware-oriented adaptation of a particle swarm optimization algorithm for object detection," in *Proc. 11th EUROMICRO Conf. on Digital Systems Design, DSD08*, Sept. 2008, pp. 904–911.

[9] J. Hawkins and S. Blakeslee, *On intelligence*. Times Books, 2004.

[10] G. Matrella, F. Grossi, V. Bianchi, I. De Munari, and P. Ciampolini, "An environmental control HW/SW framework for daily living of elderly and disabled people," in *Proc. Conf. on Telehealth and Assistive Technologies*, Apr 2008.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. conf. on Neural Networks*, vol. IV, 1995, pp. 1942–1948.

[12] S. Cagnoni, M. Mordonini, and J. Sartori, "Particle swarm optimization for object detection and segmentation," in *Proc. on Applications of Evolutionary Computing: EvoWorkshops 2007*. Springer, 2007, pp. 241–250.

[13] G. Dileep, "How the brain might work: a hierarchical and temporal model for learning and recognition," Ph.D. dissertation, Stanford University, 2008.

[14] J. Hawkins and D. George, *Hierarchical Temporal Memory - concepts, theory, and terminology*. [Online]. Available: http://www.numenta.com

[15] N. Noury, P. Rumeau, A. Bourke, G. Laighin, and J. Lundy, "A proposal for the classification and evaluation of fall detectors," *IRBM*, vol. 29, no. 6, pp. 340 – 349, 2008.

[16] C. Doukas and I. Maglogiannis, "Advanced patient or elder fall detection based on movement and sound data," in *Proc. 2nd Int. Conf. on Pervasive Computing Technologies for Healthcare*, 2008, pp. 103–107.

[17] M. Grassi, A. Lombardi, and oth., "A hardware-software framework for high-reliability people fall detection," in *Proc. IEEE Conf. on Sensors 2008*, Oct. 2008, pp. 1328–1331.

[18] V. Bianchi, F. Grossi, I. De Munari, and P. Ciampolini, "Integrating fall detection into a home control system," in *Proc. 10th AAATE Conference*, Sept 2009, pp. 309–313.

[19] R. Cucchiara, A. Prati, and R. Vezzani, "A multi-camera vision system for fall detection and alarm generation," *Expert Systems journal*, vol. 24, no. 5, pp. 334–345, nov 2007.