Low-complexity energy proportional posture/gesture recognition based on WBSN

Alexis Aulery, Jean-Philippe Diguet, Christian Roland
University of Bretagne Sud - CNRS
Lab-STICC, Lorient, France

Olivier Sentieys
INRIA
University of Rennes, France

***Abstract*—This paper addresses the issue of low-power posture and gesture recognition in indoor or outdoor environments without any additional equipment. For applications based on predefined postures such as environment control and physical rehabilitation, we show that low cost and fully distributed solutions, that minimize radio communications, can be efficiently implemented. Considering that radio links provide distance information, we also demonstrate that the matrix of estimated inter-node distances offers complementary information that allows for the reduction of communication load. Our results are based on a simulator that can handle various measured input data, different algorithms and various noise models. Simulation results are useful and used for the development of real-life prototype.**

I. Introduction

Numerous applications in domains such as healthcare and sports rely on Wireless Body Sensor Networks (WBSN). When inertial sensors from inertial measurement unit (IMU) are considered, posture and gesture recognitions can be implemented but the power consumption is a key issue that limits usage and autonomy. The system in [1] is for instance widely used for motion capture based on IMU. Although it offers good performance, it last few hours and the computation of the associated avatar orientation is performed on a remote PC, which means that this solution is not compliant with outdoor and autonomous activities such as sport training (i), environment remote control (ii), physical therapy exercices (iii) or working gestures analysis (iv). Another solution proposed in [2] for motion capture uses IMU and acoustic measures to provide distances information. Data are collected and post-processed, but this does not last few hours either. Autonomy of the system is a still a real challenge, which means an ultra-low power implementation of computation within a wearable system without the need of any remote device.

The main source of energy consumption of WBSN is the radio transceiver. This can be highlighted by a simple comparison between the energy cost per bit for a wireless transceiver and the energy cost of a computing operation. The energy to transmit one bit on a Zigbee Transceiver corresponds to about 100 to 1000 32-bit Multiply and Accumulate (MAC) operations on a Cortex M3 processor [3] [4]. This clearly shows that the best way to increase the lifetime of a node is to reduce the number of data to be sent by means of increasing the complexity of local processing.

The objective is this work is the design of an autonomous WBSN for posture and gestures recognition. Our approach first relies on local and distributed computing to minimise communications and provide a wearable solution. Power optimisation is also based on the appropriate selection of algorithms and sensors that meet application requirements while minimising operation and data transfers. Finally we consider two case studies that match with the four application domain previously cited (i-iv). The first one is the recognition of a posture among a set of predefined postures and the second one is the recognition of a gesture previously learned.

With the aim to reach an energy-autonomous system, we have tested different strategies and algorithms. Ideally algorithms should be tested with real measured data, however it is not always possible to set a new experiment for each case. So we developed our own simulator based on Matlab. This framework allows to use measured data or to generate sensor data with captured motion data files and noise models, and then to apply algorithms and to give feedback with statistics and performance results.

In order to reach the autonomy objective (low power and wearable computing), we use low-power MEMS and take full advantage of redundancy and of existing radio communications that provide information on postures as well. This strategy leads to a solution that combines local computation and a posture classification based on a Principal Component Analysis (PCA) and the Nearest Centroid Classifier (NCC). This is tested for a set of postures considering different combinations of sensor inputs to compare performance. The proposed method for gesture recognition is constructed by extension of the posture case considering a gesture as a sequence of postures. We both tested a slow and a fast gesture. Results show the possibility/necessity to adapt the pace of the system to the motion considered.

The paper is organised as follows. Section II presents the two application case studies. Section III presents the algorithms for calculation and classification and the way they were designed for this study. Section IV presents the simulation framework used to test the algorithms before presenting the results in Section V.

II. Case Study: Posture and Gesture Recognition

A. Posture Recognition

A posture is a particular position of the human body, which can provide a significant amount of important information on

978-1-4673-7201-5/15/$31.00 ©2015 IEEE

nonverbal communication or embodied emotions. As examples, postures can signal both the enduring characteristics of a person and his current emotions and attitudes. In this study we consider posture analysis and recognition with data issued from a WBSN composed of set of nodes distributed on the human body. We consider a set of 12 arm postures, which can for example be used for home automation control.The idea is that postures are based on a combination of a set of angles between joints. For this study, postures have been captured in a BVH file using the Xsens/MVN system [1]. From these captured data, orientation of nodes are computed and used to build the library of reference postures. A second dataset is constructed by adding a variable error to ideal orientations of nodes. This emulates the situation when the user does not exactly replicate the expected orientations of arms and will be used to test the tolerance of algorithms to data variability. This orientation error has been modeled by an Additional White Gaussian Noise (AWGN) on each axis of the Euler angles of orientation. Although it is a very general and simple error model, it is relevant to model experiments in real conditions. Moreover, specific noise models can be easily added for each node orientation, for specific result validations.

B. Gesture Recognition

A gesture can be defined as a movement of part of the body.The aim of gesture recognition is to interpret human gestures via mathematical algorithms and to enable humans to communicate with the machine and interact naturally without any mechanical devices. Gesture recognition is so performed as an extension of the static posture case. A gesture is therefore divided into several postures and the recognition of successive postures from a library validates the gesture.

For this case study, two gestures were tested. The first one is a quick kick coming from a capoeira movement and is depicted in Fig. 2. The second is a slow gesture generated from the sequence of the transition between postures number 3, 11 and 6 of the static case depicted in Fig. 1. It corresponds to a movement that can be used in functional rehabilitation applications.

Figure 1: Posture of arms chosen for recognition.

Figure 2: Frames extracted from the capoeira move.

III. Radio-Aided Posture Recognition

Figure 3 presents the principle of our WBSN, a node is composed of sensors, computing and radio-communication resources. Each node can communicate with other nodes and a central device (typically a smartphone) linked to a local network.

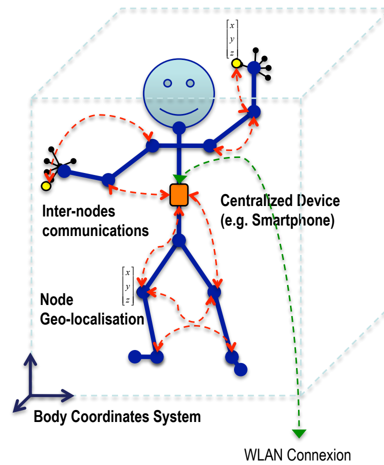


Figure 3: WBSN for posture/gesture recognition.

Considering the power consumption of radio communication order of magnitude larger than processing operations (1 transmitted bit corresponds to the energy of 100-1000 MAC operations), our objective is to implement posture/gesture recognition by means of distributed computing so that radio communication energy can be drastically reduced. We identified the Principal Component Analysis (PCA) as good candidate to efficiently implement such a classification. Firstly it is a linear method that allows to reduce the minimal vector basis to compute posture coordinates, secondly each node can compute its local contribution to the complete projection. Finally a central node gathers all contributions and computes the Nearest Centroid Classifier (NCC) to select the best posture among the remaining candidates.

PCA is a generic method that can use different heterogeneous data. Our objective is to use low-cost and low power IMU sensors (accelerometer, magnetometer, gyroscope, anemometer). These sensors require additional signal processing to deliver relevant data. We have tested different configurations and, more importantly, we have also considered the signal strength matrix – radio signal power from node i to node j, which are freely available data in the context of a WBAN – that can be used to improve the robustness of the IMU-based classification. An accelerometer is very low power component (10µW) that can be used permanently, it can also be used for movement detection. In this project we consider different combinations including accelerometer, magnetometer, radio and gyroscope, these data can be used as raw values or after a local computing. In the next sections we first present the node-orientation computation in both static and dynamic cases, then we give details about the classification algorithm.

A. Orientation Calculation

1) Static case (postures): vector reduction: In the static case, accelerometer and magnetometer provide a minimum set-up to compute orientation of nodes. There are several methods to compute this orientation. Here, since the posture is static, a computation can be performed with a simple algorithm. Moreover, we consider that for each node the data update is done only if the node is really static. It ensures that data transmitted are reduced to new relevant information and it also allows to compute orientation being sure of the input stability.

The principle of the computation is to use environmental information. An accelerometer measures gravity so the down direction and the magnetometer measures earth magnetic field that hold north direction. These two vectors can be used to compute unit vectors of body frame.

$$
\begin{aligned}
\vec{u_z} &= -\vec{acc}/||\vec{acc}|| & (1)\\
\vec{U_x} &= -\vec{mag} - \vec{u_z}(\vec{u_z} \cdot \vec{mag}) & (2)\\
\vec{u_x} &= \vec{U_x}/||\vec{U_x}|| & (3)\\
\vec{u_y} &= \vec{u_z} \times \vec{u_x} & (4)
\end{aligned}
$$

Equations (1) to (4) show a simple way to compute unit vectors of the body frame in static case. $\vec{acc}$ and $\vec{mag}$ are vectors measured by accelerometer and magnetometer and $\vec{u_x}, \vec{u_y}, \vec{u_z}$ are unit vectors of body frame. These vectors form the rotation matrix and so orientation of node.

2) Dynamic case (gestures): Kalman filtering: In dynamic conditions, real time orientation is needed. A tracking has to be done using appropriate algorithms that take advantage of gyroscope measurement. The point is that, in the dynamic case, accelerometer does not only measure gravity but also linear acceleration. Magnetometer is not affected but is not enough to compensate for this error. Gyroscope then helps to track orientation during motion while accelerometer and magnetometer correct orientation during static phase. In our case this data fusion is done by an Extended Kalman Filter (EKF). Several kinds of EKF already exist for orientation estimation. Our EKF is based on [5] version that is gyro-free. Considering the power consumption of a gyroscope, which is two orders of magnitude larger than the accelerometer, it is important to make a sparse use of it. That is why we implement two versions of the filters: with and without gyroscope inputs.

B. Classification

Classification is performed in two steps. The first step is a dimension reduction, similar to a data fusion. The choice of PCA is due to its simplicity in terms of computation. Moreover the linearity of this method allows to distribute computation over all nodes of the WBSN. An important parameter of PCA is the number of eigenvalues considered. Each eigenvalue contains a part of whole information. Then by accepting an information degradation, the number of dimensions kept can be reduced. As power optimization is a key factor, the number of eigenvalues must be minimized to reduce local memory size and the amont of data to transmit with wireless communications.

The second step is the classification by a Nearest Centroid Classifier (NCC). For each reference posture, the projection resulted from PCA is the center of this class. Then, for any posture tested, the distances between the projection and class centers are computed and the nearest one is chosen.

IV. Simulation Framework

The objective of this work is the development of posture/gesture recognition algorithms based on inertial sensor and radio measurements. However, for the development and the validation of the algorithms, a direct implementation in a hardware prototype is a difficult and error prone task and it was therefore necessary to build a specific simulator. Fig. 1 presents the principle of the developed simulator and the different parts of the simulation framework are detailed in the following sections.

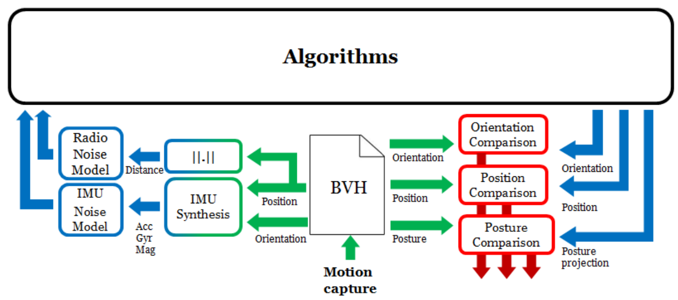


Figure 4: Principle of the developed simulator.

A. Input data generation

In the study, we considered two types of data to be processed: data issued from an Inertial Measurement Unit (IMU), i.e., accelerometer, magnetometer and gyroscope, and Euclidean distances between nodes of the WBSN obtained from radio received signal strength measurements.

The simulator uses standard BVH files that contain motions of an avatar to generate these IMU and radio data. The captures of the avatar movements can be performed using a motion capture system such as Xsens/MVN [1] or Vicon [6] – in the following experiments we used Xsens/MVN. From the BVH file, the position and orientation of each joint of the avatar can be computed for each frame and it is then possible to compute ideal values of IMU sensors as if they were located on any segment of the avatar. The same is true for the real distances between any couple of nodes on the avatar, which can also be directly computed. Then for each IMU axis, an error compliant with different noise models is added to generate realistic but synthetic measurements. Concerning the synthesis of realistic radio signals compliant with scenarios, if Received Signal Strength Indication (RSSI) method is chosen, the received radio signal power P in $[dB]$ at distance d from the transmitter is defined as $P = P_0 + n \log(d_0/d)$ where n is the loss exponent and P_0 the reference power at distance d_0. Then, combined with appropriate channel and noise model [7] [8], realistic distance measurement data can be generated. With this method, any scenario stored in a BVH file can be used to generate data from sensors and radio signal.

978-1-4673-7201-5/15/$31.00 ©2015 IEEE

B. Data computation

During simulation, all the WBSN nodes are updated at each simulation step and the algorithms are executed using generated sensor and radio data for each node. The simulation step is driven by the data originally captured with a given acquisition rate (e.g. 30Hz for the movement in Fig. 2). Depending on the scenario, adequate algorithms can be tested together with variations of some parameters to explore solutions with different trade-offs in communication and computation load.

C. Performance analysis

Different performance metrics are calculated during simulation and are summarised in a log file. In case of posture and gesture recognition, the simulator computes a success metric, which is the number of correct matches for all runs (1000 runs in the following experiments). Due to the fact that scenario data are known, we can easily compare results with original data. Not only postures but also intermediate data as orientation of nodes can be compared. It is then easier to identify which part of the algorithm process is accurate, or if particular nodes calculate wrong values. Other metrics, such as node activity for power consumption or computation load, will be added in future versions of the simulator.

V. Simulation Results

In this section we present results on posture and gesture recognition. For each case experimental conditions and testbeds are first described before giving some results on recognition accuracy for different conditions and algorithms.

A. Posture Recognition

The 12 postures depicted in Fig. 1 were chosen as a testbed for this case study. Nodes of the WBSN are located on the shoulders, elbows, wrists, chest, and hip. As discussed in Section II-A, there are two kinds of errors that affect the inputs. The first one is the error on the posture reproduction and the second one is the sampling noise of the inertial sensors (IMU). The standard deviation of sensor noises is set to 2% of gravity norm for the accelerometer and to 2% of earth magnetic field norm for magnetometer. For distances an AWGN is also added with a standard deviation of 1 centimeter. This noise model on distances corresponds to the case where distances can be evaluated with centimeter precision. The following results are done by the average of 1000 runs for each score bar. The orientation errors on limbs are with a standard deviation of $0°$ to $10°$. Considering that the differences between angles of the 12 different reference postures are about $90°$, then all postures can be classified with a very low asymptotic error.

PCA is tested for different numbers of eigenvalues. A higher number of eigenvalues means more precision but also requires more data to be transmitted. First simulation results showed that only few eigenvalues are enough to give good recognition rate for each posture. Then, the following simulations are performed with only 2, 3, and 4 eigenvalues.

Figure 5 gathers simulation results with 2, 3, and 4 eigenvalues for four combinations of inputs: *orientations only*; *raw distances only*; *orientations and raw distances*; *raw accelerometer data and distances*. Simulations are performed to measure the recognition tolerance for orientation errors that represent the inexact reproduction of posture by the user. When there exists recognition errors with only $0°$ or $2°$ of error on postures, recognition can be considered as inefficient. The first row of Fig. 5 shows that inputs with *orientation only* provide good results with at least 3 eigenvalues. The maximum error is about 12% for posture 7 and other postures are about 1%. But with only 2 eigenvalues the decrease in information is to large to provide good results. Even orientation errors of $2°$ or $4°$ lead to errors of up to 30%. On the other hand, the difference between 3 and 4 eigenvalues is not significant, and adding more eigenvalues does not decrease error rate significantly.

The second row of Fig. 5 shows that *distances only* are not tolerant at all to orientation errors. There is no error with original postures, which means that 1 centimeter precision on distances is enough to allow the discrimination between postures. But, no tolerance to orientation error means that user has to be extremely precise on his posture. In most of the cases, this is a strong constraint for the user and for the application. Moreover increasing the number of eigenvalues does not improve significantly the results. Therefore, *distances only* are not a good candidate for a posture classification.

The third row of Fig. 5 presents interesting results. Even if *distances only* does not help that much, the combination of *distances and orientations* provides better results than *orientations only*. Results with 3 and 4 eigenvalues are similar and slightly better for the third combination. Once again, posture 7 gives the maximum error with about 9%. The interesting point here is the case with only two eigenvalues on the left that shows a real improvement. The maximum error is about 20% with no significant error before $6°$ of orientation error. Depending on the accuracy required by the application, two eigenvalues can be enough.

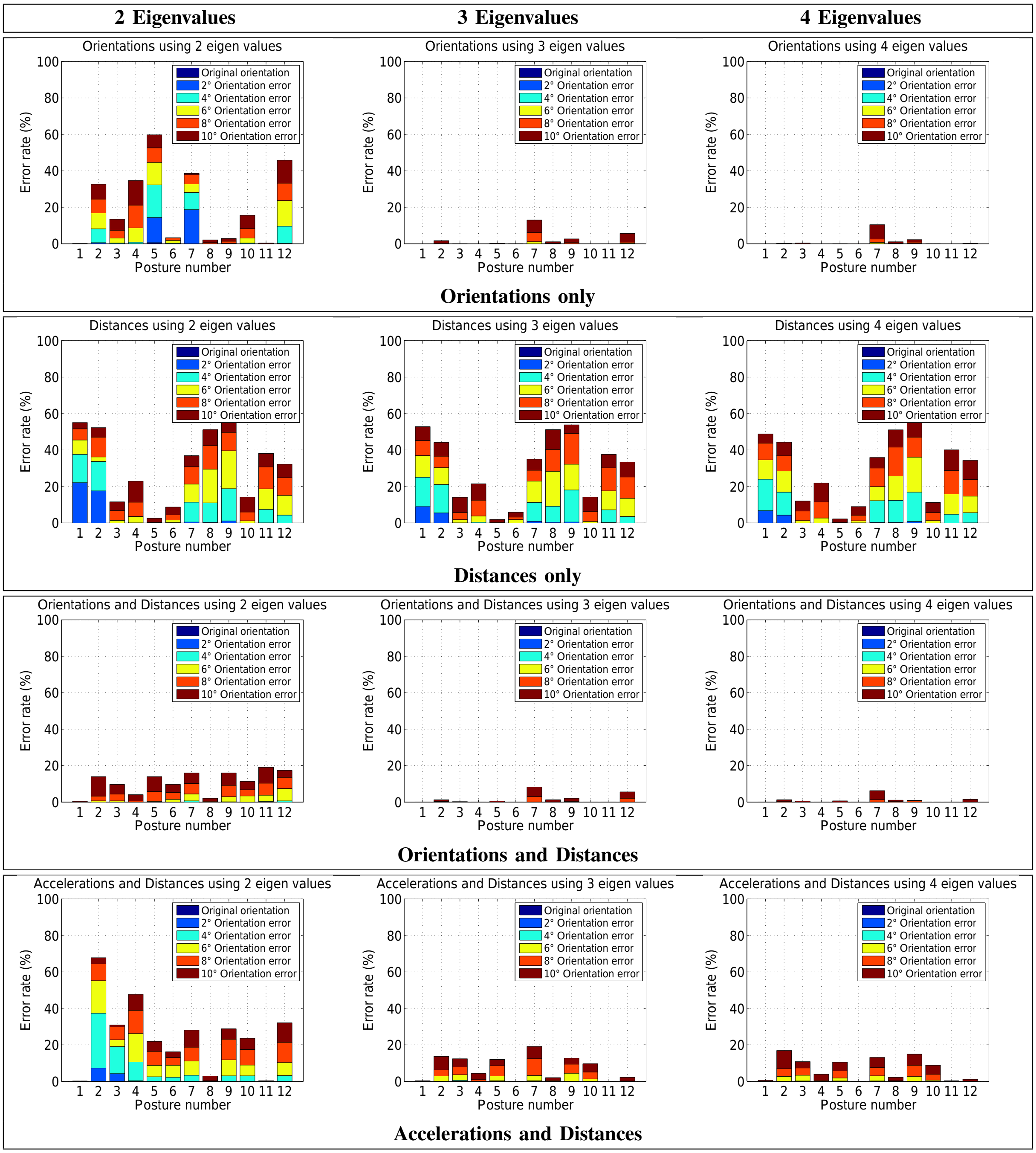
Magnetometer data could be very frequently disrupted by other objects (e.g. phones) or by parts of metal. In this case, only *accelerometer data and distances* can be used and results without magnetometer are given in the fourth row of Fig. 5. From the figure, it can be observed that 3 or 4 eigenvalues give good results with no significant error before $6°$ of orientation error. This shows that even without magnetometer, a classification is possible with an acceptable error rate. Moreover, *acceleration and distances* are independent of north direction, which means that no pre-processing on inputs is necessary to remove this dependency. Raw data extracted from sensors can be used directly on this case.

B. Gesture Recognition

All postures forming a gesture can be considered as a posture library. Applying a PCA on this library would lead to a continuous curve in the PCA space. Then a curve matching algorithm could be used to recognize a gesture. Here we propose and use a less sophisticated method based on the same principle. For this simulation, the two tested gestures are presented in Section II-B: a fast gesture from a capoeira

978-1-4673-7201-5/15/$31.00 ©2015 IEEE

Figure 5: PCA error rate for different combinations of inputs and number of eigenvalues.



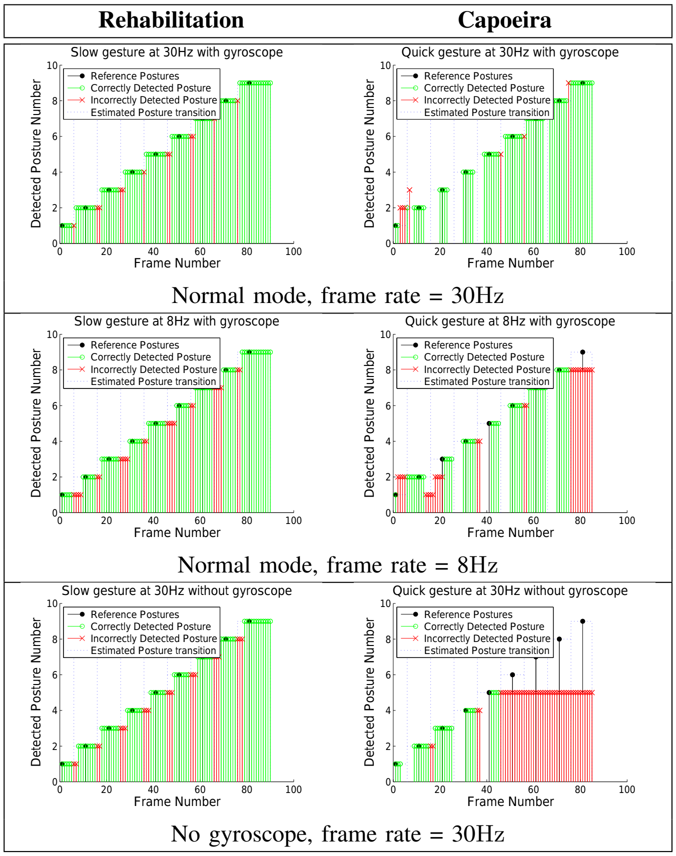
movement and a slow gesture corresponding to a functional rehabilitation movement. The capoeira movement is recorded with the Xsens/MVN system [1] and is composed of 84 frames at 30 frames per second (fps). The second movement is generated at 30 fps with a duration of 3 seconds. For the classification, the reference postures of the gesture are taken every 10 frames that mean 3Hz. The nodes of the WBSN considered for these simulations are those located on ankles, knees, elbows, and wrists.

Inputs of PCA classification are orientations of nodes represented by quaternions calculated by an Extended Kalman Filter (EKF). For each gesture three configurations are tested. The first configuration is the normal mode of the EKF at 30Hz, the second one is the normal mode of the EKF at 8Hz, and the last one is the EKF without gyroscope data at 30Hz. Figure 6 presents the results of the gesture recognition for the two gestures and for the three configurations. For each frame of the gesture, the detected posture, if any, is given. Bounds that delimit time domain of each posture are the middle of consecutive references postures. The objective is to

978-1-4673-7201-5/15/$31.00 ©2015 IEEE

test different sensor data sets and acquisition rates that directly impact power consumption.

Figure 6: Results on gesture recognition of rehabilitation and capoeira movements.



In the first configuration, at 30Hz, both slow and quick gestures are well decomposed and recognised. We can clearly see a regular monotonic progression of detected posture number that corresponds to the continuity of the gesture. In the second configuration, at 8Hz, the slow gesture is still well decomposed. Even if some postures are not within the right bounds, we can observe that the detection profile is still regular and monotonic. On the other side the quick gesture is not so well decomposed, especially in the beginning and at the end. This is due to the fact that the first and the last postures of this gesture are physically similar.

For the two considered gestures, the reduction of sampling rate (and so the computation requirements) does not have the same impact. This shows that, depending on the type of application (slow or fast movements), a coarse or fine detection can be used. Therefore, different algorithm configurations can be considered to address each case and thus allowing for a significant reduction in the energy consumption.

In the last configuration, without gyroscope data, the two gestures show significantly different results. The slow gesture is still well decomposed but the algorithm is unable to decompose and to track the quick gesture. This shows the importance of gyroscope measurement in case of quick motion tracking. This is a really interesting achievement since the power consumption of the gyroscope is three order of magnitude higher than the accelerometer. Then, these results give room to save energy by keeping high computation rate but by cutting off the gyroscope when considering applications dealing with slow gestures.

In these results, bounds that delimit the time domain validity were chosen arbitrary in the middle of references postures. But the real posture transitions do not necessary happen at the chosen middle time. So in the case where a fine detection is required, future work may include the definition of a precise time domain or domain transition.

Moreover, there are many ways to compute a metric for Quality of Service (QoS). For example, we can take into account the duration of each posture in addition to the sequence of postures. The metric has to be defined accordingly to the application chosen. The advantage here is that this metric is only computed by the central node which can be configured at will. These kind of configurations can be explored with our simulator which was designed for that purpose.

VI. Conclusion

The proposed posture recognition algorithm is simple and efficient. Different combinations of inputs allow for an adaptive behavior according to environmental constraints. We showed that an alternative is possible when magnetometer is disturbed using distance information. Moreover, using the fusion of orientation and distances shows better results than orientation only. It proves that distances holds relevant information which are complementary to orientations. The study on gesture recognition shows that different velocities of motion leads to different processing rate. The use, or not, of gyroscope data also permits to save a significant amount of energy. This leads to the development of adaptive algorithms that would be able to manage the operating mode of each node to save power and keep a good application quality of service.

References

[1] "Xsens product," http://www.xsens.com/en/general/mvn.

[2] V. Daniel, A. Rolf, V. Giovanni, B. John, G. Markus, M. Wojciech, and P. Jovan, "Practical motion capture in everyday surroundings," *ACM Trans. on Graphics*, vol. 26, p. 35, 2007.

[3] C. Bachmann, M. Ashouei, V. Pop, M. Vidojkovic, H. Groot, and B. Gyselinckx, "Low-power wireless sensor nodes for ubiquitous long-term biomedical signal monitoring," *Communications Magazine, IEEE*, vol. 50, no. 1, pp. 20–27, January 2012.

[4] B. Marr, B. Degnan, P. Hasler, and D. Anderson, "Scaling energy per operation via an asynchronous pipeline," *Very Large Scale Integration (VLSI) Systems, IEEE Trans. on*, vol. 21, no. 1, pp. 147–151, Jan 2013.

[5] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, B. W. P. D. of Aeronautics, and S. U. Astronautics, "A gyro-free quaternion-based attitude determination system suitable for implementation using low cost sensors," *IEEE Position Location and Navigation Symp.*, 2000.

[6] "Vicon system," http://www.vicon.com.

[7] R. D'Errico and L. Ouvry, "A statistical model for on-body dynamic channels," *Int. Journal of Wireless Information Networks*, vol. 17, no. 3-4, pp. 92–104, 2010.

[8] M. Mackowiak and L. Correia, "A statistical model for the influence of body dynamics on the gain pattern of wearable antennas in off-body radio channels," *Wireless Personal Communications*, vol. 73, no. 3, pp. 381–399, 2013.

978-1-4673-7201-5/15/$31.00 ©2015 IEEE