Development of a Wireless Low-Power Multi-Sensor Network for Motion Tracking Applications

Daniele Comotti*§, Michele Ermidoro§, Michael Galizzi§, Andrea Vitali¶
*Department of Electrical, Computer and Biomedical Engineering, University of Pavia
Via Ferrata 1, 27100, Pavia, Italy, Email: daniele.comotti01@ateneopv.it

§Department of Engineering, University of Bergamo, Viale Marconi 5, 24044, Dalmine (BG), Italy

¶STMicroelectronics, Via Olivetti 2, 20864, Agrate Brianza (MB), Italy

***Abstract*—This work presents a novel wireless and low power Attitude and Heading Reference Systems network based on low-cost MEMS (Micro Electro-Mechanical System) sensors, developed for motion tracking systems. Biomedical and rehabilitation purposes as well as gaming and consumer electronics may be the potential applications of this network. The paper aims to describe the hardware architecture, the embedded sensor fusion algorithm and the motion tracking system.**

***Index Terms*—IMU, MARG sensors, Quaternion Kalman Filter, AHRS, 3D Motion Tracking.**

I. Introduction

Inertial Measurement Units (IMUs) as well as Magnetic, Angular Rate and Gravity (MARG) sensors based on low cost MEMSs are the most promising alternatives to optical systems for applications where an accurate and reliable estimation of the orientation of limbs or nodes is required, such as body motion tracking. These systems typically consist of triaxial gyroscopes, accelerometers and magnetometers combined together in order to provide the nine degrees of freedom that are needed to compute the orientation of the platform. If the processing is performed on board, then an Attitude and Heading Reference System (AHRS) is achieved. By means of miniaturized and wireless AHRSs the motion can be captured without any impact on the body movement. Moreover, low power consumption is another desirable feature for more comfortable and low cost applications.

This work describes the development of a motion tracking system prototype based on wireless and low power AHRSs; even though orientation estimation of IMUs and MARG sensors has been widely investigated in the last years [1] [2], few are the wireless and embedded full systems providing a body motion tracking network. To the best of the author's knowledge, the first wearable wireless 3D body motion tracking system based on consumer-grade MEMS sensors has been demonstrated during last December by Xsens and STMicroelectronics at Electronica 2012 in Munich, Germany [3].
The paper is organized as follows: Section II and Section III present the hardware and the sensor fusion algorithm of the single node respectively. Orientation estimation performance as well as other notable features of the single node are shown in Section IV together with a comparison with current state-of-the-art AHRSs. Finally, Section V describes the network based on five nodes realizing the prototype of a motion tracking system.

II. AHRS Hardware

The single node of the motion tracking system presented in this work is a wireless AHRS prototype based on low cost MEMS sensors and mounted on a printed circuit board (PCB). A system on board provided by STMicroelectronics, the iNemo M1 [4], is the sensing and processing unit; it integrates a 6-axis geomagnetic module, a 3-axis gyroscope and an ARM Cortex M3 32-bit microcontroller of the STM32F family on a $13 \times 13 \times 2\ \mathrm{mm}^3$ board.

The wireless protocol is the Bluetooth V3.0 standard, because of the availability of miniaturized modules providing a simple to use and reliable interface; even though smaller modules are available on the market, during this initial development phase the BT31 [5] provided by Amp'ed RF has been used. It provides a Class 1 Bluetooth V3.0 stack covering up to 100 meters range in a $15 \times 27\ \mathrm{mm}^2$ form factor and comes with a wide set of commands to be used for configuration purpose.

As outlined in Figure 1, the raw data from the sensors are sent by means of the SPI and I2C protocols for the gyroscope and the geomagnetic module respectively, whereas the microcontroller hosted on the iNemo M1, operating at a 32 MHz clock frequency, communicates with the Bluetooth module through the UART bus and performs the sensor fusion algorithm. A set of commands implemented in dedicated C++ libraries on computer side allows the user to communicate with the AHRS and to control the firmware; the connection established between the Bluetooth module onboard and a Bluetooth adapter on the computer provides a serial virtual COM port on which the commands and the data can be exchanged. The output data rates can be set at three different frequencies: 25 Hz, 50 Hz and 100 Hz.

The power is supplied by means of a 3.7 V Li-ion battery with a 100 mAh charge and a $24 \times 10 \times 2\ \mathrm{mm}^3$ form factor, mounted on the PCB. The battery is directly connected to the low-drop out regulator integrated in the iNEMO M1, which in turn provides the power supply to the Bluetooth module.

978-1-4799-0330-6/13/$31.00 ©2013 IEEE

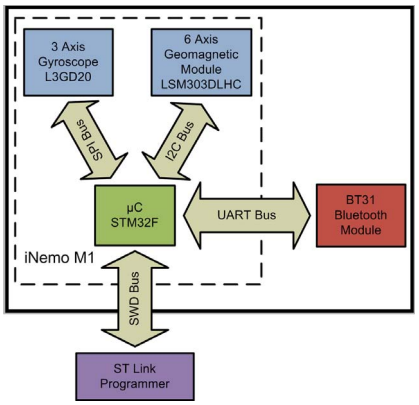

Fig. 1. Overview hardware architecture of the single node, with the iNemo M1 board, the BT31 module and the SWD / JTAG interface.

As shown in Figure 2, the board has not been optimized during this prototyping phase and the PCB dimensions are equal to $35 \times 50 \times 7$ mm^3 whereas the plastic housing is $55 \times 40 \times 15$mm^3 large.

Fig. 2. Picture of the AHRS prototype with the plastic package.

III. Sensor Fusion Algorithm

The orientation of the platform is estimated using the 3D raw data provided by the geomagnetic module and the gyroscope. Common classes of algorithms estimating the orientation are based on three steps: integration, vector observation and Kalman filtering [6]. The integration step predicts the state by means of the gyroscope data, the vector observation step provides the optimized observation using the magnetometer and the accelerometer raw data, and finally, the Kalman filter is applied, providing the optimal estimators which minimize the error covariance. [2].

The presented work aims to embed an onboard orientation estimation algorithm and to provide a low power consumption, hence, a solution which keeps the microcontroller clock at low frequencies, without compromising the accuracy. Most of the algorithms presented in the literature make use of a 7 components state vector [7], [8], which are the quaternion representing the orientation and the biases on the three gyroscope axes. The complexity has been first decreased by using a state vector based on the estimated orientation only. Among the several ways available to represent the 3D orientation, quaternions have been chosen because of their advantages in terms of computability and because of they are not affected by singularity problems [2]. The state vector is then defined as follows:

$$x = [q_w \, q_i \, q_j \, q_k]^T \tag{1}$$

where q_w is the real component and q_i, q_j and q_k are the imaginary components. The following notations will be used:

- $x_{t|t-1}$ is the state predicted at time t by means of the estimated quaternion at the previous time;
- z_t is the state observed at time t;
- $x_{t|t}$ is the state estimated at time t.

In the following, the three steps providing the orientation estimation algorithm are described in detail. Moreover, the magnetometer calibration method performing the estimation of the magnetic scale factors and biases is presented.

A. Integration Step

The integration step is performed using the angular rates provided by the gyroscope. The quaternion rate can be computed according to the following equation [9]:

$$\frac{\delta q_t}{\delta t} = \frac{1}{2} \cdot \left([0 \, \omega_x \, \omega_y \, \omega_z]^T \otimes q_{t-1} \right) \tag{2}$$

where ω_i is the angular rate relevant to the $i-th$ axis, $\otimes$ is the Hamilton product and q_{t-1} is the previous time step quaternion. Then the quaternion at time t is achieved by means of the following:

$$q_t = q_{t-1} + \frac{\delta q_t}{\delta t} \cdot \Delta t \tag{3}$$

where Δt is the sampling period.

The state equation is derived from Equations 2 and 3 by adding the process noise coming from the gyroscope:

$$x_t = f(x_{t-1}) + w_t \tag{4}$$

$$f(x_{t-1}) = \frac{1}{2} \cdot \left([0 \, \omega_x \, \omega_y \, \omega_z]^T \otimes x_{t-1} \right) \cdot \Delta t + x_{t-1} \tag{5}$$

where w_t is the process noise with covariance matrix Q, related to the gyroscope. Hence, the state at time t is given by the prediction equation:

$$x_{t|t-1} = f\left(x_{t-1|t-1}\right). \tag{6}$$

Since $f(x_t)$ is not a linear function of x_t, the transition matrix F_t is computed by means of the following Jacobian:

$$F_t = \frac{\delta f(x_t)}{\delta x}\Big|_{x=x_{t-1|t-1}} =$$

$$\begin{bmatrix} 1 & -\Delta t/2 \cdot \omega_x & -\Delta t/2 \cdot \omega_y & -\Delta t/2 \cdot \omega_z \\ \Delta t/2 \cdot \omega_x & 1 & \Delta t/2 \cdot \omega_z & -\Delta t/2 \cdot \omega_y \\ \Delta t/2 \cdot \omega_y & -\Delta t/2 \cdot \omega_z & 1 & \Delta t/2 \cdot \omega_x \\ \Delta t/2 \cdot \omega_z & \Delta t/2 \cdot \omega_y & -\Delta t/2 \cdot \omega_x & 1 \end{bmatrix}. \tag{7}$$

The prediction covariance matrix is then given as follows:

$$P_{t|t-1} = F \cdot P_{t-1|t-1} \cdot F_t^T + Q \tag{8}$$

where $P_{t-1|t-1}$ is the estimation covariance matrix at $t-1$.

B. Vector Observation Step

The observation z_t is achieved by minimizing the function $\mathbf{e}(x)$, which represents the difference between the data and the references in the body frame:

$$\mathbf{e}(x) = \begin{bmatrix} x^* \otimes \mathbf{a_E} \otimes x - \mathbf{a_B} \\ x^* \otimes \mathbf{m_E} \otimes x - \mathbf{m_B} \end{bmatrix} \tag{9}$$

where x is the quaternion to be observed, $*$ is the conjugate operator, $\mathbf{a_E}$ and $\mathbf{m_E}$ are the accelerometer and magnetometer references in the earth frame respectively, $\mathbf{a_B}$ and $\mathbf{m_B}$ are the accelerometer and magnetometer measurement vectors respectively. The $\mathbf{a_E}$, $\mathbf{a_B}$, $\mathbf{m_E}$ and $\mathbf{m_B}$ vectors are arranged in such a way that the quaternion product can be performed, by using a zero as real component. For example, the $\mathbf{a_B}$ vector is defined as follows:

$$\mathbf{a_B} = [0 \; a_x \; a_y \; a_z]^T \tag{10}$$

where a_x, a_y and a_z are the accelerometer data relevant to the x, y and z axes respectively. The normalized accelerometer reference has been chosen equal to the gravity vector $\mathbf{a_E} = \bar{g} = [0 \; 0 \; 0 \; 1]^T$, whereas the magnetometer reference is computed by carrying the magnetic acquisition in the earth frame and by assuming it directed on the x and z axes:

$$\mathbf{h_E} = x \otimes \mathbf{m_B} \otimes x^* \tag{11}$$

$$\mathbf{m_E} = \left[0 \; \sqrt{h_{E,x}^2 + h_{E,y}^2} \; 0 \; h_{E,z}\right]^T. \tag{12}$$

Since Equation 9 is non linear with respect to x, an optimization algorithm is needed to minimize $\mathbf{e}(x)$. The same Gauss-Newton method based procedure proposed in [8] has been used, even though in this work it has been implemented in the quaternions domain. The $k-th$ iteration is the following:

$$x_{k+1} = x_k - \left(J(x_k)^T \; J(x_k)\right)^{-1} J(x_k)^T \; \mathbf{e}(x_k) \tag{13}$$

where $J(x_k)$ is the Jacobian of $\mathbf{e}(x)$. Concerning the first iteration, the initial guess x_0 is the estimated quaternion at the previous time, $x_{t-1|t-1}$. As depicted in Figure 3, experimental results show that the algorithm converges in 3 - 4 steps, as expected. At the end of the iterations the observed state z_t is achieved.

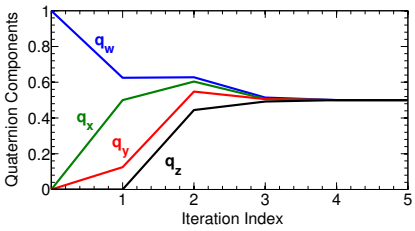

Fig. 3. Convergence of the observation estimation algorithm. The initial guess $[1 \; 0 \; 0 \; 0]^T$ converges to the observed quaternion $[0.5 \; 0.5 \; 0.5 \; 0.5]^T$ in 3 - 4 iterations.

The observation of the state is made according to the following model:

$$z_t = H \, x_t + v(t) \tag{14}$$

where H is the observability matrix and $v(t)$ is the observation noise with covariance matrix R. Since actually the observation z_t is the quaternion representing the state, the H matrix has been chosen equal to the identity matrix $I_{4\times4}$.

C. Kalman Filtering

Once the integration and the vector observation steps have been computed, the Kalman filter is performed:

- The Kalman gain is computed:

$$K_t = P_{t|t-1} \, H^T \left(H \, P_{t|t-1} \, H^T + R\right)^{-1}. \tag{15}$$

- The estimation step is performed:

$$x_{t|t} = x_{t|t-1} + K_t \left(z_t - H \, x_{t|t-1}\right). \tag{16}$$

- The estimation covariance matrix is computed:

$$P_{t|t} = P_{t|t-1} - K_t \, H \, P_{t|t-1}. \tag{17}$$

The covariance matrices of the state and observation models have been experimentally tuned considering that the former is related to the static and dynamic biases of the gyroscope, whereas the latter determines the latency of the accelerometer and magnetometer measurements contribution. In dynamic conditions the gyroscope measurements are more truthful than the geomagnetic ones, viceversa at the steady state the magnetometer and the accelerometer are more reliable. Hence, the following values have been achieved for both the matrices:

$$Q = I_{4\times4} \cdot 0.00005, \tag{18}$$

$$R = I_{4\times4} \cdot 0.6 \tag{19}$$

The initial value of the estimation covariance matrix has been chosen equal to the identity matrix $I_{4\times4}$.

Finally, the initial estimated quaternion, $x_{0|0}$ has been chosen equal to $[1 \; 0 \; 0 \; 0]^T$, since the filter takes only few iterations to converge.

D. Magnetometer Calibration

The magnetometer is the sensor which most suffers from the lack of a calibration procedure. The main sources of errors are the so called soft and hard iron distortions:

- Soft iron materials close to the sensor condense magnetic flux toward themselves, causing distortions depending on the direction of the sensor and a gain over each axis different from 1.
- Hard iron ferrous materials, whose magnetic field remains constant in a fixed location relative to the sensor, produce an offset on each axis of the sensor.

For a calibrated sensor, the measured magnitude will be constant for all orientations. Hence, by performing m measurements covering almost the whole space of the magnetic field, scale factors and biases can be computed by solving the following minimization problem:

$$\arg\min_{\mathbf{x}} \left[\sum_{i=1}^{m} f_i^2\right] \tag{20}$$

$$f_i = \left(M - \left\| K \cdot u_i - \vec{b} \right\| \right) \tag{21}$$

where $\mathbf{x}$ is the calibration parameters vector, M is the intensity of the magnetic field, u_i is the $i-th$ measurement, K is the scale factors matrix and $\vec{b}$ is the offsets vector:

$$K = \begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{xy} & k_{yy} & k_{yz} \\ k_{xz} & k_{yz} & k_{zz} \end{bmatrix} \tag{22}$$

$$\vec{b} = [b_x \; b_y \; b_z]^T \tag{23}$$

$$\mathbf{x} = [k_{xx} \cdots k_{zz} \; b_x \; b_y \; b_z]^T \tag{24}$$

Several optimization algorithms have been investigated and the Gauss-Newton method has been chosen to compute the calibration parameters thanks to its fast convergence in few iterations. Since the calibration procedure can be performed offline, a large number of data can be acquired. The $k-th$ iteration is defined as follows:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left(J^T \; J \right)^{-1} \; J^T \; \mathbf{f} \tag{25}$$

where $\mathbf{f}$ is the m functions array and J is the Jacobian of $\mathbf{f}$:

$$\mathbf{f} = [f_1 \cdots f_m]^T \tag{26}$$

$$J = \begin{bmatrix} \frac{\delta f_1}{\delta k_{xx}} & \cdots & \frac{\delta f_1}{\delta b_z} \\ \vdots & \vdots & \vdots \\ \frac{\delta f_m}{\delta k_{xx}} & \cdots & \frac{\delta f_m}{\delta b_z} \end{bmatrix} \tag{27}$$

The intensity of the magnetic field M is taken as the average value of the acquired data from the magnetometer; however, the algorithm converges with whatever value of M since the intensity affects only the absolute value of the calibration parameters $\mathbf{x}$. Concerning the initial guess of the calibration parameters, these have been chosen equal to the ideal values:

$$\mathbf{x} = [1 \; 0 \; 0 \; 1 \; 0 \; 1 \; 0 \; 0 \; 0]^T \tag{28}$$

Figure 4 shows the uncalibrated and calibrated data set (4000 samples) acquired from the magnetometer in a common environment with the predominant component of the Earth magnetic field: the reference circumferences and sphere of intensity M respectively enclose and superimposes the calibrated acquisition with few exceptions laying outside. Experimental results show that the algorithm converges in 3 - 4 iterations with an RMS error below the 2% of the intensity reference M.

IV. AHRS Performance

The proposed AHRS has been compared with state-of-the-art solutions [10]–[15] in terms of orientation accuracy, power consumption, dimensions, maximum output data rate and wireless capability. Concerning the orientation estimation, preliminary test experiments have been performed on the sensor fusion algorithm at a 50 Hz output rate by means of a stereophotogrammetric motion analyzer (Smart Motion Capture System provided by BTS SRL, Italy) in both static and dynamic conditions. Each component of the roll, pitch and yaw Euler angles related to the estimated quaternion has

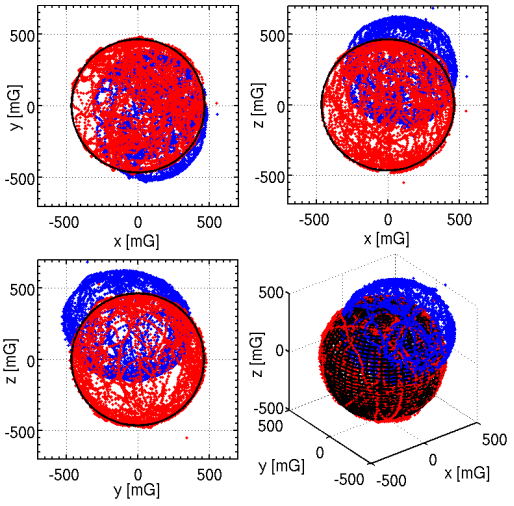

Fig. 4. 3D plot (bottom right) of a calibration experiment and its three perspectives. The black sphere $m_x^2 + m_y^2 + m_z^2 = M^2$ represents the ideal magnetic vector space whereas the blue dotted ellipsoid not centered in (0, 0, 0) describes the uncalibrated raw data acquired from the magnetometer. The result of the calibration is depicted by the red dotted sphere, well fitting the black reference one, centered in the axes origin. It has to be noted that the Earth magnetic field intensity is approximately 470 mG.

been studied separately by performing individual rotations along the x, y and z axes with respect to the initial zero position and has been compared to the optical reference (truth reference) by computing both the RMS and the maximum error. Figure 5 depicts three different measurements achieved from both the AHRS and the BTS. Table I compares the overall performance of the proposed system in terms of static and dynamic accuracy, power consumption, maximum output data rate, size of the package and wireless capability, with respect to the state-of-the-art solutions. The output data rate has been tested with up to five nodes simultaneously connected to the same computer. As far as the orientation accuracy is concerned, the proposed AHRS is comparable to the others in static conditions whereas it is sligthly less accurate during rotation. Moreover, the low power performance, the small form factor and the wireless capability are attractive features in AHRS designs for motion tracking applications.

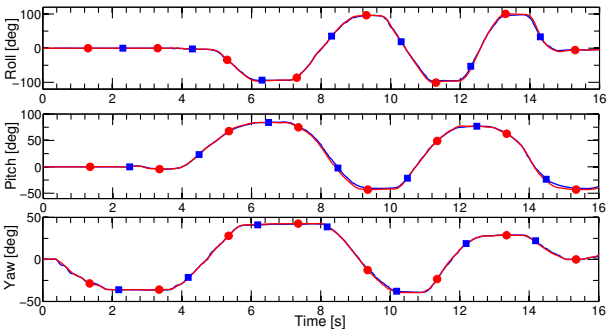

Fig. 5. Roll, pitch and yaw angles of three test cases on the sensor fusion algorithm. The blue line is the angle under test provided by the presented AHRS whereas the red line is the truth reference of the optical system. The accuracy performance well fits the results shown in Table I.

TABLE I
PERFORMANCE COMPARISON BETWEEN THE STATE-OF-THE-ART SOLUTIONS AND THE PROPOSED AHRS.

	Static RMSE [deg]			Dyn. RMSE [deg]			Static Max Error [deg]			Dyn. Max Error [deg]			Power Cons.	Max ODR [Hz]	Size $[mm^3]$	Wireless
	Roll	Pitch	Yaw	Roll	Pitch	Yaw	Roll	Pitch	Yaw	Roll	Pitch	Yaw				
XSens MTi 30	0.2	0.2	n.a.	0.5	0.5	1.0	0.4	0.4	n.a.	0.5	0.5	1.0	≈ 160 mA @ 3.3 V	400	57×42×23	no
XSens MTi 300	0.2	0.2	n.a.	0.3	0.3	1.0	0.25	0.25	n.a.	1.0	1.0	1.0	≈ 245 mA @ 3.3 V	400	57×42×23	no
XSens MTw	n.a.	n.a.	n.a.	2.0	2.0	2.0	0.5	0.5	1.0	n.a.	n.a.	n.a.	n.a.	120	58×35×15	802.15.4
VN100 IMU	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	0.5	0.5	2.0	n.a.	n.a.	n.a.	≈ 70 mA @ 5.0 V	300	36×33×9	no
Shimmer 9DOF + Base Board	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	50	53×32×34	802.15.4
X-io X-IMU	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.	≈ 100 mA @ 3.6 V	512	57×38×21	BT Class1
Proposed AHRS	0.2	0.4	0.4	1.5	1.1	0.9	0.35	0.7	0.6	3.15	2.5	2.3	≈ 50 mA @ 3.8 V	100	52×40×15	BT Class 1

V. NETWORK FOR MOTION TRACKING

In order to achieve a body motion tracking system prototype, a network based on 5 AHRSs has been developed by displacing them respectively on the right arm, right forearm, lower chest, right upper leg (thigh) and right lower leg (crus). The aim is to provide a solution which estimates the movement of each limb independently and without suffering from the bad positioning of the nodes, that is without mounting the hardware and the package in a precise way. Hence, a method for calibrating the network in order to represent the orientation of each node with respect to a common reference system (the reference system of the person) and for any positioning condition, has been developed. As shown in Figure 6, each node can be referred to the following systems:

- The main reference system (0), which is the one achieved from the magnetic and gravity references and represented by the identity quaternion $[1\ 0\ 0\ 0]^T$.
- The reference system of the person wearing the nodes (1), which is assumed to be achieved by means of a yaw offset with respect to the main reference system[1]:

$$q^0_{0,1} = [q_{1w}\ 0\ 0\ q_{1k}]^T. \tag{29}$$

- The reference system of each AHRS (2), which depends on the specific positioning conditions. The quaternion representing the rotation between (0) and (2) is $q^0_{0,2}$
- The reference system of each AHRS during the experiment (3), which is identified by the estimated quaternion $q^0_{0,3}$.

It has to be noted that the reference system of the person is not the same for all the nodes, since the initial position of the person is supposed to be the so called standing T position. Hence, the arm and forearm nodes will be 90 yaw degrees shifted with respect to the other nodes, by means of the following quaternion:

$$q_h = [1/\sqrt{2}\ 0\ 0\ 1/\sqrt{2}]^T. \tag{30}$$

[1]The quaternion $q^c_{a,b}$ represents the rotation from a to b with respect to the reference system c

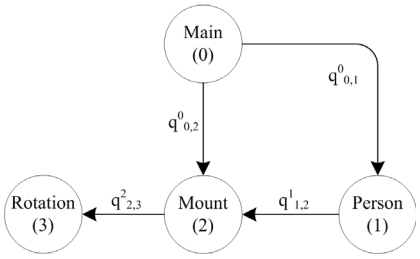

Fig. 6. Diagram of the reference systems of the nodes. The transition between two references is performed by a quaternion.

In order to represent all the rotations in the reference system of the person, the quaternion between (2) and (3) with respect to (1) is needed, which is:

$$q^1_{2,3} = q^1_{1,2} \otimes q^2_{2,3} \otimes q^{1\,*}_{1,2}, \tag{31}$$

where $q^1_{1,2}$ and $q^2_{2,3}$ are computed as follows:

$$q^1_{1,2} = q^{0\,*}_{0,1} \otimes q^0_{0,2}, \tag{32}$$

$$q^2_{2,3} = q^{0\,*}_{0,2} \otimes q^0_{0,3}. \tag{33}$$

It has to be noted that $q^0_{0,1}$ for the right arm and forearm is given by $q^0_{0,1} \otimes q_h$. The unknown quaternion $q^0_{0,1}$ has been achieved by one AHRS precisely positioned and aligned with the reference system of the person; this acquisition is performed once before starting the experiment and then the node can be displaced on the related limb.

Finally, once the quaternion representing the orientation of each limb with respect to the reference system of the person, $q^1_{i\ 2,3}$, is computed, the hierarchical model needs to be defined in order to take into account relative rotations in a sequence of limbs:

$$q_{limb_i} = q^*_{limb_{i-1}} \otimes q^1_{i\ 2,3}, \tag{34}$$

where i is the hierarchical index, ranging from 1 to 3, representing each node accordingly to Figure 7, and $q_{limb_0} = [1\ 0\ 0\ 0]^T$.

The network topology chosen for the motion tracking prototype is based on a star layout where the central hub is a

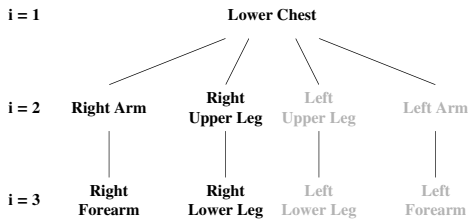

Fig. 7. Hierarchical relation between the limbs of the network. The grey leaves have to be implemented.

main computer connected by means of a Bluetooth adapter to the five AHRSs. A motion tracking application prototype based on the C# programming language has been developed; the software is based on two threads which handle the data communication and the graphical user interface (GUI). The five AHRSs transmit the estimated quaternion at a 100 Hz output rate. When new data are received, the thread which deals with the communication awakes and acquires the new quaternions available. Concerning the GUI, it has been implemented using the XNA framework and consists of a 3D skeleton whose limbs are controlled by the quaternions of the five AHRSs, as shown in Figure 8. The thread which handles the GUI updates every 20 ms (half of the sensor fusion algorithm update rate) each limb of the skeleton with the new quaternion provided by the data communication layer and accordingly to the associaton between the same limb and the related AHRS.

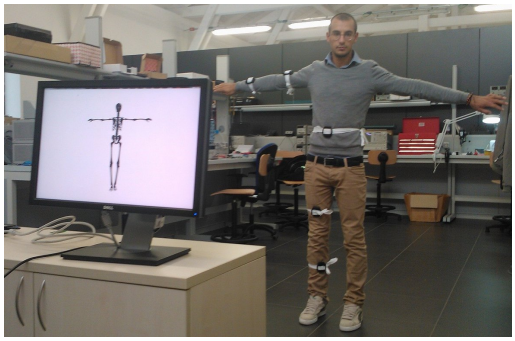

Fig. 8. Picture of the GUI of the motion tracking application prototype with the user standing in T position and wearing the 5 AHRSs.

VI. Conclusions

In this work, the prototype of a motion tracking system based on a wireless and low power AHRS network has been presented. First, the full development of the single node has been described and the performance evaluation with respect to the state-of-the-art solutions has proved its good suitability for AHRS design. Then, a motion tracking system based on five nodes operating at 100 Hz ODR has been implemented (the demonstration video can be found at the YouTube link http://www.youtube.com/watch?v=UT7Rg108syk). Even though some improvements are needed for consumer purposes, the presented network is a promising solution for motion tracking applications.

Ongoing activities are focused on the hardware of the single node; on the basis of the results shown in this work, a new generation of AHRS, whose PCB is depicted in Figure 9, has been developed and fabricated in order to further reduce the area occupancy as well as the power consumption. The characterization of the new hardware, the development of a Bluetooth scatternet for the connection of a higher number of nodes, the full validation of the sensor fusion algorithm and the implementation of a new version of the motion tracking system are foreseen by the end of 2013.

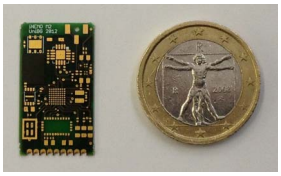

Fig. 9. PCB of the new $14 \times 25\ \mathrm{mm}^2$ miniaturized AHRS.

References

[1] S. Madgwick, A. Harrison, and R. Vaidyanathan, "Estimation of imu and marg orientation using a gradient descent algorithm," in *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, 29 2011-july 1 2011, pp. 1 –7.

[2] A. Sabatini, "Quaternion-based extended kalman filter for determining orientation by inertial and magnetic sensing," *Biomedical Engineering, IEEE Transactions on*, vol. 53, no. 7, pp. 1346 –1356, july 2006.

[3] XsensPressRelease, "Xsens and stmicroelectronics demonstrate wearable wireless 3d body motion tracking," http://www.xsens.com/en/news/ip-licensing-news/xsens-and-stmicroelectronics-demonstrate-wearable-wireless-3d-body -motion-tracking.

[4] STMicroelectroncs, "inemo m1 official web site," http://www.st.com/internet/analog/product/253162.jsp.

[5] Amp'edRF, "Amp'ed rf products," http://www.ampedrf.com/modules.htm.

[6] H. Bruckner, C. Spindeldreier, H. Blume, E. Schoonderwaldt, and E. Altenmuller, "Evaluation of inertial sensor fusion algorithms in grasping tasks using real input data: Comparison of computational costs and root mean square error," *Wearable and Implantable Body Sensor Networks, International Workshop on*, vol. 0, pp. 189–194, 2012.

[7] X. Yun, C. Aparicio, E. Bachmann, and R. McGhee, "Implementation and experimental results of a quaternion-based kalman filter for human body motion tracking," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, april 2005, pp. 317 – 322.

[8] J. Marins, X. Yun, E. Bachmann, R. McGhee, and M. Zyda, "An extended kalman filter for quaternion-based orientation estimation using marg sensors," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 4, 2001, pp. 2003 –2011 vol.4.

[9] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep. 2005-002, Mar. 2005.

[10] "Xsens mti-10 series," http://www.xsens.com/en/general/mti-10-series.

[11] "Xsens mti-100 series," http://www.xsens.com/en/general/mti-100-series.

[12] "Xsens mtw series," http://www.xsens.com/en/mtw.

[13] X-IOTechnologies, "X-io technologies x-imu official-web site," http://www.x-io.co.uk/node/9.

[14] VectorNav, "Vectornav vn100 product brief," http://www.vectornav.com/products/vn100-rug.

[15] Shimmer, "Shimmer 9dof kinematic sensor official web-site," http://www.shimmer-research.com/p/products/sensor-units-and-modules/wireless-9dof-kinematic-sensor.