A warning wearable system used to identify poor body postures

Dan-Marius Dobrea
Electronics, Telecommunications and Information Technology
"Gheorghe Asachi" Technical University
Iaşi, România
mdobrea@etti.tuiasi.ro

Monica-Claudia Dobrea
Electronics, Telecommunications and Information Technology
"Gheorghe Asachi" Technical University
Iaşi, România
mcdobrea@etti.tuiasi.ro

***Abstract*—This paper describes a new wearable warning system (named Intell.TieSens) able to identify a poor body posture – a posture capable to generate in time lumbar and/or cervical pains. The proposed system was designed to be low power, portable, lightweight and having a Bluetooth wireless data communication capability. Through the Bluetooth connection the Intell.TieSens can be paired to a personal computer or a smartphone and notify the user when a poor body posture is identified. All the Intell.TieSens system components are embedded in a tie. The identification of the poor postures is done by defining several system's triggering zones and identifying the entering of the system in these zones. The experimental results show the abilities of the Intell.TieSens to successfully distinguish between the poor and the correct body postures.**

Keywords—back pain, wearable device, capacitive sensor, accelerometer, gyroscope, Kalman filter, Bluetooth Low Energy

I. INTRODUCTION

The back pain affects the population of any age, with a higher incident rate among working and elderly people [1]. The back pain (BP) is also the most common reason for absence from the job [2], [3]. In the same time, from the point of view of its incidence, the low back pain (LBP), lumbar pain, is the most common problem, followed by the neck pain (cervical pain or the higher back pain - HBP) [4], [5]. Even if the pain is localized in the lower back, sometimes it can be felt anywhere along the spine – from the neck down to the hips.

The acute and chronic BP could be generated by: **(a)** muscle tensions or spasms, [6], **(b)** a strain in muscles or ligaments [7], [8], **(c)** an infection of the nerves/spine, **(d)** a spinal misalignment [2], [9], or **(e)** an injury of one or more component parts of different back structures (spinal disks, nerves, muscles, ligaments, tendons, the vertebral periosteum, and the bones – *i.e.*, vertebras) [1], [7], [8], [9]. Commonly, the BP has one of the following causes: **(a)** accidental falls, **(b)** improperly lifting, pushing, pulling or carrying objects, **(c)** making one of these activities with heavy objects, **(d)** making fast and awkward movements etc. However, in this paper, not are these receiving our focus but, instead, we are interested in those BP that are generated by the everyday activities like standing, sitting or driving in a poor body posture. An improper posture may strain some ligaments or muscles, may generate a continuum disc-compression - all of these to the point of an acute and/or a chronic BP. From this, one can state the high importance of keeping a correct body posture.

To build a system able to prevent the BP generated by a poor body position, we have first to consider that such a system to be endowed with the capacity to detect different postures. Today, a wide variety of posture detection systems are already developed, part of them using contact-based sensors such as electromyography (EMG) [3], [6], push buttons [8], pressure sensors [10] or non-intrusive sensors such as video surveillance [11], [12], inductive sensors [13], accelerometers [2], [14], [15], [16], gyroscopes [14], [17] or force sensing resistors [18]. The non-contact systems can be either of wearable type - using, for example, a vest [15], a girdle [17], several elastic straps to fix the sensors [14], [19], [16] or a shoe [10] - or, they can have the sensors embedded in objects from the proximities of the investigated subject (e.g., on a prayer mat [13] or in a chair [18]).

This paper proposes a wearable and cost-effective warning system that monitors and analyzes the spinal position of a subject in the sagittal plane, in real time and in his every day's environments, notifying the user whenever a poor body position was detected. The system measures: **(a)** the tilting angle of the torso (mainly, the tilting angle of the lumbar and thoracic spine parts) and **(b)** the head pitch rotation (the rotation angle in the median/sagittal plane) which is also proportional with the cervical tilting angle of the spine. The tilting angle of the torso was estimated from the information supplied by both sensors: gyroscope and accelerometer. For this, a Kalman filter was used.

II. BACK PAIN DUE TO POOR BODY POSTURES

Maintaining a proper posture or improving an incorrect one are two of the most simple and effective ways to keep the spine healthy or to alleviate the BP. With these simple approaches, we can address the root cause of many types of BP. In a straight posture, the view of the spinal column in the sagittal plane can be likened to an "S" shape. The physicians,

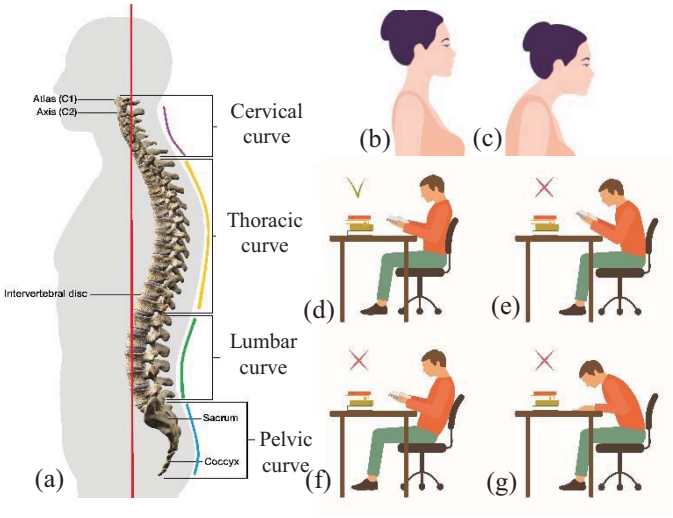


Fig. 1. **(a)** A good posture, the spinal curves are balanced and aligned, **(b)** and **(c)** Forward Head Posture, **(d)**, **(e)**, **(f)** and **(g)** different body postures when reading a book or working with a computer

This study was funded in part by a donation from Texas Instrument Company.

978-1-5386-5558-0/18/$31.00 ©2018 IEEE

through a series of large population studies and biomechanics research, defined the proper posture (see Fig. 1(a)) by the angles between the inflection points of the different spine curves and the deviation about the sagittal and transverse planes [2].

The forward head posture - hunched back and 'text neck' -, presented in Fig. 1(c), versus the correct body posture, presented in Fig. 1(b), has been linked to increasing use of technology - especially the smartphones, tablets, and smartwatches. This body posture generates middle and HBP, shoulder pain, headaches and migraine, all of these having as a cause the disc hernias and spine alignment problems due to prolonged smartphone use [9].

Any position that affects the opening angles of the spinal curves, the balance and the alignment of the vertebras will generate in time back pain due, mainly, to the strain induced in muscle, ligaments or other soft tissues. For example, the lumbar curve has vertebras that should be slightly curled inward the subject body, see Fig. 1(a). A subject loses this natural form when he/she stays on a chair, in a slouching position (i.e., the body leans its back to the back of the chair and slide its bottoms forward, Fig. 1(f)).

From the previously presented arguments as well as from the postures illustrated in Fig. 1, one can infer the necessity to find a way for both monitoring the spine position of a subject (i.e., the different tilt angles of different spine sections) and notifying the subject whenever he/she takes a poor posture, with the final aim of preventing the disease and the pain.

III. THE HARDWARE SYSTEM DESIGN

The ***Intell.TieSens*** embeds within a tie all the following components: a SimpleLink CC2650 wireless microcontroller (MCU), a CR2032 coin cell battery, sensors for both orientation and movement detection (i.e., a gyroscope and an accelerometer – a MPU9250 circuit), a series of capacitive sensors and the driving circuit (FDC 2214), and a Bluetooth transceiver for wireless data communication, see Fig. 2.

A. The capacitive sensors

With the Intell.TieSens system, the head position measurement in the sagittal plane was performed with a system compound of capacitive sensors placed around the neck. To find the sensors optimal number and their best positions that would ensure the minimum measurement error of the head tilt angle in the sagittal plane (the pitch movement - the rotation of the neck in the median/sagittal plane), several positions and numbers of the sensors were tested, as follows: **(a)** four capacitive sensors (C_{lf}, C_{lb}, C_{rb}, and C_{rf}) with the configuration presented in Fig. 3(a), **(b)** three capacitive

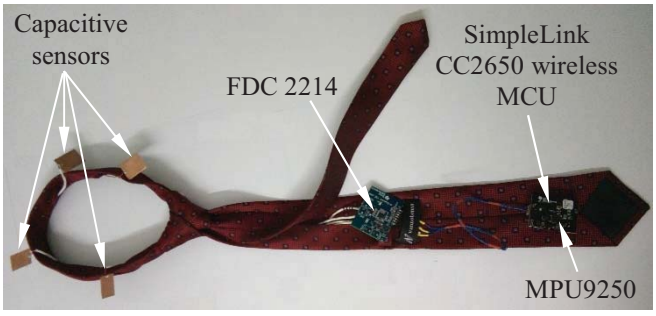


Fig. 2. The ***Intell.TieSens*** system

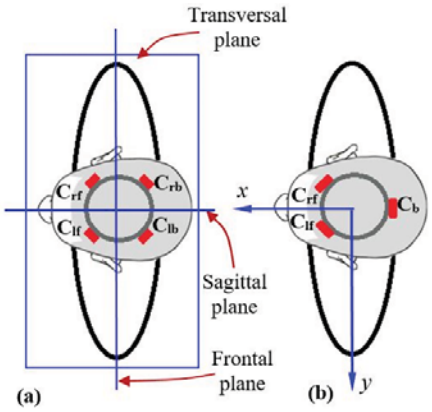


Fig. 3. Sensors positions: (a) four sensor approach and (b) three sensor approach

sensors (C_{lf}, C_{rf}, and C_b) disposed as in Fig. 3(b) and **(c)** one sensor; this last approach has as reference the Fig. 2(b) from which C_{rf} and C_{lf} sensors were eliminated.

B. The system hardware architecture

A block diagram of the wearable ***Intell.TieSens*** system is presented in the Fig. 4. The wearable system's core is given by a SimpleLink CC2650 wireless System-on-Chip (SoC) MCU and three different types of sensors. The gyroscopic sensor and the accelerometer sensor, both embedded within the MPU9250 circuit, are used to compute the tilting angle of the subject's torso – the tilting angle of the lumbar and thoracic spine components. By using these sensors, several poor body positions – like the ones presented in Fig. 1(f) and (g) – can be very easily highlighted. The values of the capacitive sensors were converted to numerical values by using the existing 28-bits capacitance-to-digital converter of the FDC2214 circuit.

SimpleLink CC2650 wireless SoC is composed of one main CPU, a RF core and a sensor controller engine. The managing of the physical layer of the Bluetooth 2.4 GHz RF transceiver (low-level control of the Bluetooth Low Energy (BLE) stack) is done by a Cortex-M0 processor. The user application and the upper layers of the BLE protocol stack are managed by the main CPU - a Cortex-M3 processor.

The external sensors circuits (FDC2214 and MPU9250) are controlled by an ultra-low-power Sensor controller engine (SCE) – a 16-bits microcontroller. To develop the ***Intell.TieSens*** system, two software modules were developed. The first module runs on the Cortex- M3 processor and it

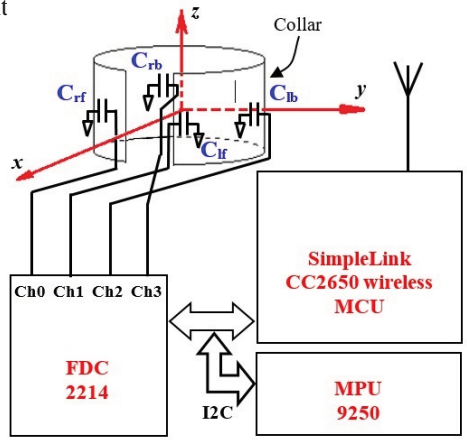
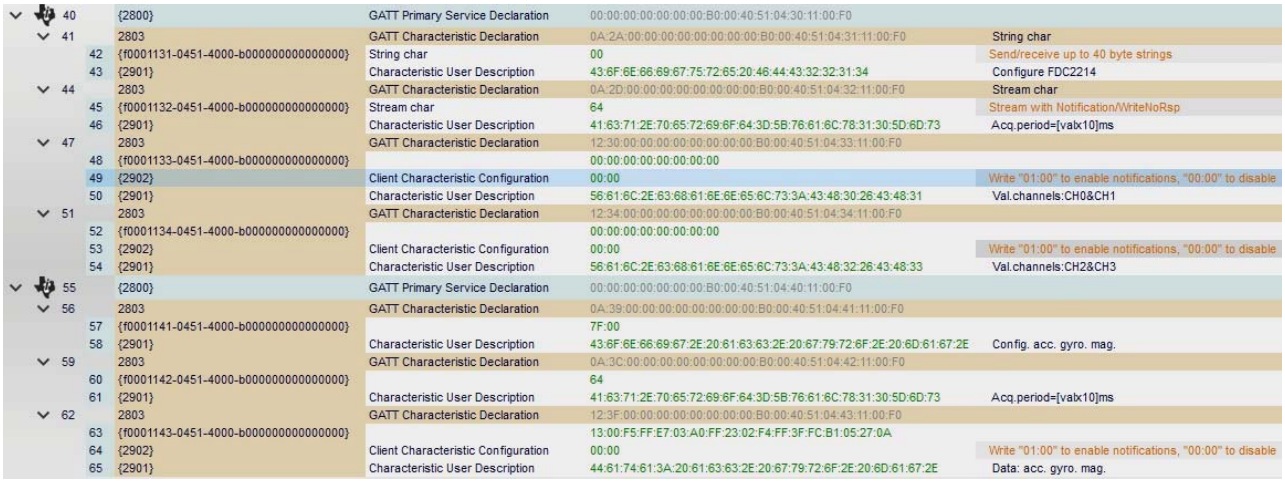


Fig. 4. The block diagram of the wearable ***Intell.TieSens*** system



40		{2800}	GATT Primary Service Declaration	00:00:00:00:00:00:00:00:B0:00:40:51:04:30:11:00:F0	
41		2803	GATT Characteristic Declaration	0A:2A:00:00:00:00:00:00:00:00:B0:00:40:51:04:31:11:00:F0	String char
	42	{f0001131-0451-4000-b000000000000000}	String char	00	Send/receive up to 40 byte strings
	43	{2901}	Characteristic User Description	43:6F:6E:66:69:67:75:72:65:20:46:44:43:32:32:31:34	Configure FDC2214
44		2803	GATT Characteristic Declaration	0A:2D:00:00:00:00:00:00:00:00:B0:00:40:51:04:32:11:00:F0	Stream char
	45	{f0001132-0451-4000-b000000000000000}	Stream char	64	Stream with Notification/WriteNoRsp
	46	{2901}	Characteristic User Description	41:63:71:2E:70:65:72:69:6F:64:3D:5B:76:61:6C:78:31:30:5D:6D:73	Acq.period=[valx10]ms
47		2803	GATT Characteristic Declaration	12:30:00:00:00:00:00:00:00:00:B0:00:40:51:04:33:11:00:F0	
	48	{f0001133-0451-4000-b000000000000000}		00:00:00:00:00:00:00:00	
	49	{2902}	Client Characteristic Configuration	00:00	Write "01:00" to enable notifications, "00:00" to disable
	50	{2901}	Characteristic User Description	56:61:6C:2E:63:68:61:6E:6E:65:6C:73:3A:43:48:30:26:43:48:31	Val.channels:CH0&CH1
51		2803	GATT Characteristic Declaration	12:34:00:00:00:00:00:00:00:00:B0:00:40:51:04:34:11:00:F0	
	52	{f0001134-0451-4000-b000000000000000}		00:00:00:00:00:00:00:00	
	53	{2902}	Client Characteristic Configuration	00:00	Write "01:00" to enable notifications, "00:00" to disable
	54	{2901}	Characteristic User Description	56:61:6C:2E:63:68:61:6E:6E:65:6C:73:3A:43:48:32:26:43:48:33	Val.channels:CH2&CH3
55		{2800}	GATT Primary Service Declaration	00:00:00:00:00:00:00:00:B0:00:40:51:04:40:11:00:F0	
56		2803	GATT Characteristic Declaration	0A:39:00:00:00:00:00:00:00:00:B0:00:40:51:04:41:11:00:F0	
	57	{f0001141-0451-4000-b000000000000000}		7F:00	
	58	{2901}	Characteristic User Description	43:6F:6E:66:69:67:2E:20:61:63:63:2E:20:67:79:72:6F:2E:20:6D:61:67:2E	Config. acc. gyro. mag.
59		2803	GATT Characteristic Declaration	0A:3C:00:00:00:00:00:00:00:00:B0:00:40:51:04:42:11:00:F0	
	60	{f0001142-0451-4000-b000000000000000}		64	
	61	{2901}	Characteristic User Description	41:63:71:2E:70:65:72:69:6F:64:3D:5B:76:61:6C:78:31:30:5D:6D:73	Acq.period=[valx10]ms
62		2803	GATT Characteristic Declaration	12:3F:00:00:00:00:00:00:00:00:B0:00:40:51:04:43:11:00:F0	
	63	{f0001143-0451-4000-b000000000000000}		13:00:F5:FF:E7:03:A0:FF:23:02:F4:FF:3F:FC:B1:05:27:0A	
	64	{2902}	Client Characteristic Configuration	00:00	Write "01:00" to enable notifications, "00:00" to disable
	65	{2901}	Characteristic User Description	44:61:74:61:3A:20:61:63:63:2E:20:67:79:72:6F:2E:20:6D:61:67:2E	Data: acc. gyro. mag.

Fig. 5. The implemented GATT services (for capacitive sensors and movement unit) and the associated characteristics, obtained with the BLE Device monitor air sniffer developed by Texas Instruments company

sustained by the TI-RTOS (TI real-time operating system). This module is responsible for the control and for the data exchange functions that are using the BLE protocol. The second module runs on the SCE.

IV. The Software Components

The software components - those that the development of the ***Intell.TieSens*** system (see Fig. 2) have involved -, use as a base the software modules developed and presented in [20] and [21]. The differences between this paper and those previously presented [20], [21] consist both in the concepts and in the implementation of the new warning wearable system that is used to identify poor body postures. Besides, some new code was developed to work with MPU9250 – a hardware component that was not used in [20], [21]. This code includes four distinct sections: (**a**) a first component, responsible to exchange data through the Bluetooth LE standard using a new Generic Attributes (GATT) service, (**b**) a second component, that runs on SCE and manages the MPU9250, (**c**) a third component, that is sustained by TI-RTOS and behaves like a link layer between the first two components, and (**d**) a fourth component, that was developed as a Windows-based application and that configures the ***Intell.TieSens*** system, gets data (accelerometric and gyroscopic data), estimates the torso tilting angle based on a Kalman Filter, recognizes the bad human posture and warns the user about it.

A. The software modules running on CC2650 wireless SoC

To encapsulate all the required behaviors of the ***Intell.TieSens*** system a new service was implemented with three new characteristics, see Fig. 5. This service is associated with the MPU9250 motion tracking device (that includes 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer) and has a 128-bit universally unique identifier (UUID) address, namely, F0001140-0451-4000-B000-000000000000. Using the first characteristic of this service, a user can select which sensor and which data/axis of the sensor should be activated or not. By writing 0x00, one can disable all channels for the MPU9250 circuit and set it in the shutdown mode. The second characteristic (with UUID = F0001142-0451-4000-B000-000000000000) allows the setting of the MPU9250 measurement period (with a resolution of 10 ms) in the [100 ms, 2.55 seconds] interval. The third characteristic stores the data for all channels, and for each sensor type. The data order is: gyroscope, accelerometer and magnetometer data, starting with x-axis an ending with z-axis. Each axis consists of a signed value represented on 16 bits. Using this characteristic, the sensors data can be accessed through the reading and notifying mechanisms of the BLE protocol. In conclusion, this new service, the new characteristics, together with the new descriptors and declarations can be interrogated over the air; Fig. 5 gives a global image of all these. Moreover, this service is also used to configure the MPU9250 and to acquire data from it.

B. The PC based software module

Also, we developed a Windows-based application able to run on a personal computer or laptop. First, this application was used to configure and record the readings from the ***Intell.TieSens*** system – the accelerometric, gyroscopic (both newly implemented) and the capacitive values. Second, this Windows 10 application acts as a client to the wearable device (***Intell.TieSens***), being responsible for triggering a warning to the user whenever a poor body posture is detected.

The Windows application was developed starting from the ***blessTags package*** version 10.0.0.0 [22]. The first author of this paper developed the ***blessTags package***, a program commercialized within the virtual Microsoft Store [22]. The ***blessTags application*** sets, communicates, uses and displays information obtained from five different types of SensorTags: CC2541DK, CC2650STK (devices manufactured by Texas Instruments Company), Thunderboard React and Sense (both SensorTags produced by Silicon Labs Company) and WICED Sense2 (product of Cypress company). In addition to these, several other functions are implemented and sustained by the ***blessTags package*** [23]: different events, triggers, activities, gadgets etc. In this research, the ***blessTags package*** was used as a support layer. Namely, from this package all the necessary functions, needed to implement the Bluetooth communication protocol, were used – see Fig. 6 with the configuration panel. In the beginning stages of our application development, the package's abilities for data visualization (either in the text or graphic format) were exploited to correctly implement the data acquisition functionalities, Fig.7.

In the present implementation of the system, the

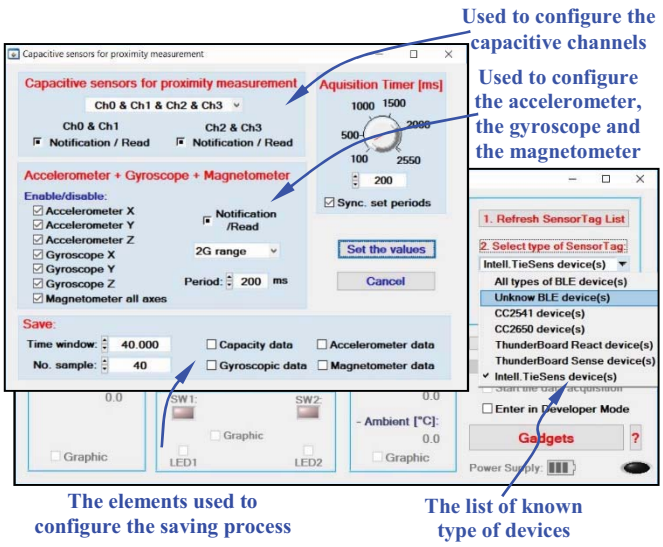


Fig. 6. **(a)** The ***blessTags'*** main user interface and **(b)** the panel used to configure ***Intell.TieSens*** wearable device

Intell.TieSens (see Fig. 2) is working as a Server: it supplies the values from the accelerometric sensor, gyroscopic sensor and from the capacitive sensors (FDC2214 circuit). On the other side, the Windows application works as a Client component: (**a**) it takes these values, (**b**) computes the subject's head pitch angle, using for this a neural network, and (**c**) computes the upper body tilting angle, by applying a Kalman Filter algorithm. In the end, a threshold technique was applied to distinguish between the good and the bad human body posture. Thus, if the tilting angle value of the subject torso is within the [-15°, +15°] interval and the pitch angle of the head is within the [-10°, +10°] interval, the user has a good body posture. Otherwise, after 30 seconds of continuously detected poor body posture, a warning signal will be generated, and the ***Intell.TieSens*** system user will be notified.

C. Kalman Filter for torso tilting angle

The accelerometer sensor embedded in the tie (*i.e.*, inside the MPU9250 circuit, see Fig. 2) is used for the computation of the tilting angle of the subject torso. With the accelerometer sensor placed on the subject chest, accordingly to Fig. 3(b) – *i.e.*, with the Z-axis vertically perpendicular on the transversal

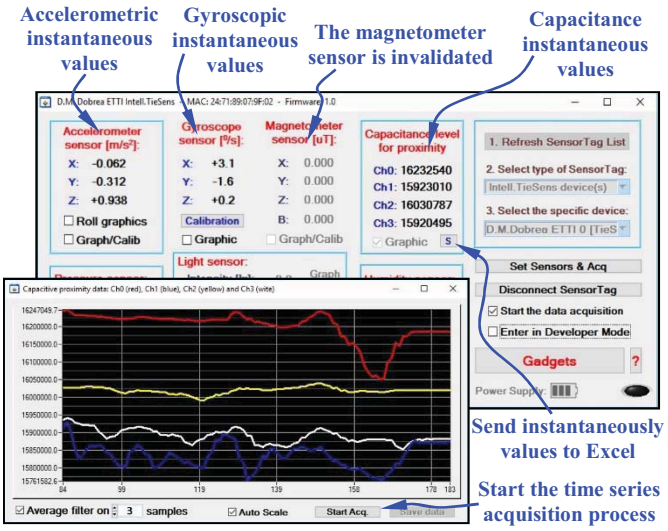


Fig. 7. The text and graphical data representation. Saving functions.

plane –, one can easily compute the upper body lean forward or backward angle by using the formula:

$$\alpha_{acc.} = tan^{-1}\left(\frac{A_z}{\sqrt{A_x^2+A_y^2}}\right) \quad (1)$$

Based on the gyroscopic sensor (that measures the angular velocity, $\omega(t)$), through a simple integration, we can compute the body tilting angle:

$$\alpha_{gyro.} = \int_0^t \omega(\tau)d\tau \quad (2)$$

Both of the above sensors incorporate different types of errors. The sources for these could be: **(a)** the noise from the basic physical processes of the semiconductors (we refer here to the electro-mechanical nature of these devices); **(b)** the noise introduced by the digitization process, **(c)** other types of noises like the human physiological tremor and **(d)** the offset drift of the gyroscopic sensor etc.

The Kalman filter (KF) is a very powerful tool [24], used to estimate a variable. Having the information supplied by the accelerometric and gyroscopic sensors and using the KF that extracts the maximum amount of data with minimal error and lower quantity of noise, we succeeded to estimate the tilting angle of the subject torso. Precisely, the Kalman filter's output was given by $x_k = [\alpha \quad \dot{\alpha_b}]^T$, the estimated values – without errors and noise – of the real tilting angle of the subject torso (α variable) and of the gyroscope drift ($\dot{\alpha_b}$ variable).

In a **first stage** the Kalman filter estimates the current state based on a previous state, a state transition model (A matrix), a control input model matrix (B matrix) and a control input:

$$\bar{x}_k = A \cdot x_{k-1} + B \cdot u_k + w_k \quad (3)$$

For our particular case, the A and B matrix are defined as:

$$A = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \quad (4)$$

with Δt representing the sampling period. At the time k the measurement of the true state, $\bar{z}_k$, is made accordingly with:

$$\bar{z}_k = H \cdot z_k + v_k \quad (5)$$

Here, H, the observational model, maps the true state space measurements provided by the sensors into the observed space. In our case $z_k = [\alpha_{acc.,k} \quad \dot{\alpha}_{gyro.,k}]^T$. In (3) and (5) the w_k (the predicted process noise) and v_k (the measurement noise) are independent white noises with a Gaussian distribution ($w_k \sim N(0, Q)$ and $v_k \sim N(0, R)$ – Q is the process noise covariance matrix and R is a covariance matrix directly related to the sensors quality too). In the prediction of a new value, the error covariance matrix should be estimated, too:

$$\bar{P}_k = A \cdot P_{k-1} \cdot A^T + Q \quad (6)$$

In the **second stage**, the estimate (that will be used to compute the torso tilt angle) is updated using the measurement:

$$x_k = \bar{x}_k + K_k \cdot (\bar{z}_k - H \cdot \bar{x}_k) \quad (7)$$

where the Kalman gain is calculated as:

$$K_k = \frac{\bar{P}_k \cdot H^T}{H \cdot \bar{P}_k \cdot H^T + R} \quad (8)$$

In the end, the error covariance matrix is updated:

$$P_k = (I - K_k \cdot H) \cdot \bar{P}_k \quad (9)$$

The Kalman filter is a recursive estimator, repeating these two stages – the first stage, considered the *prediction step*, and the second stage, considered the *update step*.

V. ESTIMATING THE HEAD TILT ANGLE

The relation that links the capacitance values (C_{lf}, C_{lb}, C_{rb}, and C_{rf}) with the head pitch angle:

$$\alpha_{pitch} = f_{pitch}(C_{lf}, C_{lb}, C_{rb}, C_{rf}) \quad (10)$$

is a very complex nonlinear function and it depends by many unknown factors [21], like: the tissues water content, the concentrations of ions suspended in this water, the internal movement of muscles and tendons etc. This relation was deeply analyzed in [21]. Two artificial neural networks (ANN) were used to model the relationships presented in (10).

In addition to the results presented in [21], in this paper we extended the research: **(a)** by studying and testing a new capacitive sensors configuration that assumed different positions as well as a different number of the used sensors, **(b)** by using a new training database (DB) for the ANN (the entire DB was recorded once again in a more reliable way – with according dramatic performance improvement) and **(c)** by computing a new performance parameter that helped us to better interpret the obtained results.

A. Database recording

Three subjects, at the age of 22, 42 and 44, participated in this study. The subjects were seated on a chair, with the body in a vertical position and with the back supported by the deck of the chair. On the front wall different landmarks were placed, from 5 to 5 degrees, starting with -60° and ending with +60°. The landmarks were placed in the vertical position, at the intersection line between the subject sagittal plane and the front wall, in order to be able to measure the exact head orientation – the head pitch angle. The participants were asked to look, one by one, at a particular landmark indicated via voice command. To ensure a reproducible measurement of each head pitch angle, a laser pointer (LP) was mounted on a helmet, at the subject eyes level, on the transversal plane, pointing right in front. The participants had to hold the LP on the specified landmark until the next command was issued.

The entire setup was made in order to obtain the training set for the neuronal networks: **the input values** – the capacitive sensors values, and the corresponding **outputs values** – the head positions. For each landmark (with some exceptions) and for each user, six different acquisitions were done at different time intervals. The exceptions regard the extreme values of the pitch angle that not all users had the physical capacity to express. In the end, 444 training set points were acquired and used to train the neuronal models.

B. The neuronal models

Two types of ANNs were used: (**a**) a MLP (multilayer perceptron) and **(b)** a RBF (radial basis function). These ANNs were chosen due to the different properties of the elementary functions used to approximate the relation (10). In MLP we used the tangent hyperbolic functions (THF) - elementary functions that respond to the full input space. Unlike these, the RBF network used localized radial basis function as elementary functions.

The MLP had only one hidden layer of neurons with a single linear output neuron. For the RBF network, the centers and width of the localized radial basis function were set by an unsupervised competitive learning rule with an intrinsic Euclidian metric. During the unsupervised learning phase (which lasted 100 epochs), the conscience full competitive rule was used. The training set used 80% of the database while the remaining 20% of data formed the cross-validation (CV) set. All performances presented in the tables below were obtained on the CV data set.

C. Performace measurements metrics

The performances of both neuronal structures used in this study were evaluated with four metrics: the mean squared error (MSE), the normalized mean squared error (NMSE), the correlation coefficient (r), and the percent error ($\%_{error}$). The MSE and NMSE were used to determine how well the ANN outputs fit the desired outputs. MSE and NMSE does not reflect, however, whether the two sets of data move in the same direction. To eliminate this shortcoming, the correlation coefficient was also computed. The percent error (with dy_i – denormalized network output for exemplar i, and dd_i – denormalized desired output):

$$\%_{error} = \frac{100}{444}\sum_{i=0}^{443}\left|\frac{dy_i - dd_i}{dd_i}\right| \quad (11)$$

can easily be misleading. For example, in our case, the output data range is [-60°, +60°]. If we obtain an error of +5° for a real value of 10°, then, the percent error ($\%_{error}$) will be of 50 (hence, a huge error); for a value of 50°, the same error (of +5°) will generate a percent error of only 10.

VI. RESULTS AND DISCUSSION

A first "visible" result of the ***Intell.TieSens*** development was the implementation of a dedicated service able to manage the MPU9250 Motion Processing Unit™, see Fig. 5. The Fig. 5 was obtained based on a Texas Instruments proprietary software package: BLE Device monitor.

Out of the two analyzed classifiers, by using the entire data set (the data from all three users) as well as the four-sensors configuration of the ***Intell.TieSens*** system, the best performances were achieved with the RBF network. In this case, the performances were: 0.0426 for MSE, 0.1491 for NMSE and a value of 0.975 for the correlation coefficient.

Table I presents the results obtained by the ***Intell.TieSens system*** in its four-sensors configuration. Comparing these results with the ones presented in [21], one can notice the improvement, in this study, in the MSE error with a factor of 13.94 for the MLP network and with a factor of 23 for the RBF ANN. The main difference between this study and [21] was the introduction of the laser pointer that ensured the accuracy and the reproducibility of the measurements. Also, the results presented in Tables I to III represent the mean values computed on all users performances and for distinct sensor configurations. In this mode, we better highlighted the mean performance shifting (i.e., decrease) obtained for the head pitch movement and for the capacitive sensor configurations with four sensors, three and, respectively, with only one sensor.

Comparing the results in Tables I, II and III, one can conclude that the use of a higher number of sensors will generate a lower approximation error. Also, the RBF ANN always gets superior performances compared to the MLP network. For example, the ***Intell.TieSens*** system in its 3-

sensors configuration, conducted with an improvement factor of 6.42 (in the case of MSE error). In all tables, immediately

TABLE I. PITCH APPROXIMATION PERFORMANCES FOR THE CASE OF FOUR SENSORS

	MSE	NMSE	r	$\%_{error}$
MLP (12)	0.000624	0.00234	0.9988	5.44
RBF (45)	0.000326	0.00122	0.9993	3.23

TABLE II. PITCH APPROXIMATION PERFORMANCES FOR THE CASE OF THREE SENSORS

	MSE	NMSE	r	$\%_{error}$
MLP (8)	0.019536	0.054264	0.9757	23.88
RBF (40)	0.003040	0.008445	0.9961	15.67

TABLE III. PITCH APPROXIMATION PERFORMANCES FOR THE CASE OF ONE SENSORS

	MSE	NMSE	r	$\%_{error}$
MLP (7)	0.085039	0.236201	0.9155	55.90
RBF (10)	0.071631	0.198961	0.9139	65.95

after the network name, the number of the neurons used for the hidden layer is specified within parentheses. This number of neurons was the optimal one required in order to minimize the approximation error and it was obtained after many training sessions.

From the user point of view, the system is working very well with four as well as with three sensors – the pitch angle errors obtained are not perceived by the user. However, the situation becomes annoying for the configuration with one single sensor, when the ***Intell.TieSens*** system generates many false alarms.

The way of computing the tilt angle of the body is a very accurate one. The system design can be very easily extended to incorporate more sensors useful in estimating the torso tilt angle or the hardware part of the ***Intell.TieSens*** system can be incorporated very easily into other wearable (e.g., a shirt).

VII. CONCLUSIONS

There are several systems used to detect different types of postures, but a large part of them is dedicated to other goals, such: fall detection [19], detection of early Parkinson disease [15] or medical assessment [6], [14]. Certainly, this system can also be used for such purposes, but its goal is to detect in a non-invasive manner the poor body position and to notify the user.

One of the main advantages of the ***Intell.TieSens*** system, as well as its novelty, are given by the innovative approach in computing the tilting angle of the head. The ability to connect through Bluetooth LE is another big gain, the system being able to work with a PC but, also, with a Smartphone.

REFERENCES

[1] A.K. Burton, R.D. Clarke, T. D. McClune, and K.M. Tillotson, "The natural history of back pain in adolescents," Spine, Vol. 21, 1996, pp. 2323-2330.

[2] A.B. Crane, S.K. Doppalapudi, J. O'Leary, P. Ozarek, and C.T. Wagner, "Wearable posture detection system," Annual IEEE Northeast Bioengineering Conference, Apr. 25-27, 2014.

[3] W.W. Lu, K.D.K. Luk, K.M.C. Cheung, and J.C.Y. Leong, "Using EMG to evaluate muscle functions in patients with low back pain (LBP) syndromes," International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 5, 1998, pp. 2666-2669.

[4] P. M. Wolsko, D. M. Eisenberg, R. B. Davis, R. Kessler and R. S. Phillips, "Patterns and perceptions of care for treatment of back and neck pain: results of a national survey," Spine, 2003, pp. 292-298.

[5] P. Cote, J. D. Cassidy, and L. Carroll, "The Saskatchewan health and back pain survey: the prevalence of neck pain and related disability," Spine, 1998, pp. 1689-98.

[6] F. Zhou, H.H. Li, G.J. Song, and L. Wang, "Comparison between healthy subjects and low back pain patients based on surface electromyography features," International Conference on Signal Processing, Nov. 6-10, 2016, pp. 1384-1387.

[7] D.N. Ghista, S. Subbaraj, J. Mazumdar, and S.M. Rezaian, "The biomechanics of the back pain," IEEE Eng. in Med. and Biology, Vol. 17, No. 3, May-Jun. 1998, pp. 36-41.

[8] S. Chopra, M. Kumar, and S. Sood, "Wearable posture detection and alert system," 5th International Conference on System Modeling & Advancement in Research Trends, 25-27 November 2016, pp. 130-134.

[9] J.M. Cuéllar, T.H. Lanman, "Text neck: an epidemic of the modern era of cell phones?," Spine, Vol. 17, No. 6, June 2017, pp. 901-902.

[10] M. Matsumoto, and K. Takano, "A posture detection system using consumer wearable sensors", 10th International Conference on Complex, Intelligent, and Software Intensive Systems, Jul. 6-8, 2016, pp. 526-531.

[11] K. Wongpatikaseree, A.O. Lim, Y. Tan, and H. Kanai, "Range-based algorithm for posture classification and fall-down detection in smart homecare system," IEEE Global Conference on Consumer Electronics, Oct. 2-5, 2012, pp. 243-247.

[12] C.F. Juang, and C.M. Chang, "Human body posture classification by a neural fuzzy network and home care system application," IEEE Trans. Syst. Man Cybern. A Syst. Hum., Vol. 37, No. 6, Nov. 2007, pp. 984-994.

[13] S. Labarni, S. Khan, and Anis Nurashikin Nordin, "Design and implementation of an inductive-based human postures recognition system," IEEE Conference on Biomedical Engineering and Sciences, Dec. 8-10, 2014, pp. 634-638.

[14] J.K. Lee, G.T. Desmoulin, A.H. Khan, E.J. Park, "A portable inertial sensing-based spinal motion measurement system for low back pain assessment," Annual International Conference of the IEEE EMBS, Aug. 30 – Sept. 3, 2011, pp. 4737-4740.

[15] W.S Wu, W.Y. Lin, and M.Y. Lee, "Forward-flexed posture detection for the early Parkinson's disease symptom," IEEE International Conference on Systems Man and Cybernetics, San Diego, USA, Oct. 5-8, 2014, pp. 1181-1185.

[16] B.S. Jonwal, and A. Wakankar, "Wearable system to measure lateral and flexion bending," International Conference on Automatic Control and Dynamic Optimization Techniques, Sep. 9-10, 2016, pp. 1085-1088.

[17] J.E. Estrada, and L.A. Vea, "Real-time human sitting posture detection using mobile devices," IEEE Region 10 Symposium, May 9-11, 2016, pp. 140-144.

[18] T. Fu, and A. MacLeod, "IntelliChair: an approach for activity detection and prediction via posture analysis," International Conference on Intelligent Environments, Jun 30- Jul. 4, 2014, pp. 211-213.

[19] W.J. Yi, and J. Saniie, "Design flow of a wearable system for body posture assessment and fall detection with Android smartphone," IEEE International Technology Management Conference, Jun. 12-15, 2014.

[20] D.M. Dobrea, and M.C. Dobrea, "Concepts and developments of an wearable system-an IoT approach," International Symposium on Signals, Circuits and Systems, July 13-14, 2017.

[21] D.M. Dobrea, and M.C. Dobrea, "A neuronal model of the 3D head position based on a wearable system," 10th International Conference and Exposition on Electrical and Power Engineering, October 18-19, 2018, Iasi, Romania, in press.

[22] D.M. Dobrea, "The blessTags application," Windows Store, https://www.microsoft.com/store/apps/9p054xsjjr1n.

[23] D.M. Dobrea, "The BLE SensorTag (***blessTags***) application.", http://www.blesstags.eu/2017/05/normal-0-false-false-false-en-us-x-none.html

[24] I. Cikajlo, Z. Matjai, and T. Bajd, "Efficient FES triggering applying Kalman filter during sensory supported treadmill walking," J. Med. Eng. Technol., Vol. 32, No. 2, 2008, pp. 133–144