Computers in Biology and Medicine 42 (2012) 364–375

Contents lists available at SciVerse ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/cbm

Real-time recognition of patient intentions from sequences of pressure maps using artificial neural networks

Manuel Chica [a,c,*], Pascual Campoy [b], María Ana Pérez [a], Tomás Rodríguez [a], Rubén Rodríguez [a], Óscar Valdemoros [d]

[a] Inspiralia Tecnologías Avanzadas, Estrada 10, 28034 Madrid, Spain
[b] Automatics and Robotics Center—Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, 28006 Madrid, Spain
[c] European Centre for Soft Computing, Gonzalo Gutiérrez Quirós, 33600 Mieres (Asturias), Spain
[d] Spaldin Sleep Systems, Calle Vieja 62-64, 26080 Logroño, Spain

ARTICLE INFO

Article history:
Received 18 May 2011
Accepted 2 December 2011

Keywords:
Artificial neuronal networks
Image processing
Movement intention recognition
Human–machine interfaces
Medical beds

ABSTRACT

Objective: In this paper we address the problem of recognising the movement intentions of patients restricted to a medical bed. The developed recognition system will be used to implement a natural human–machine interface to move a medical bed by means of the slight movements of patients with reduced mobility.
Methods and material: Our proposal uses pressure map sequences as input and presents a novel system based on artificial neural networks to recognise the movement intentions. The system analyses each pressure map in real-time and classifies the raw information into output classes which represent these intentions. The complexity of the recognition problem is high because of the multiple body characteristics and distinct ways of communicating intentions. To address this problem, a complete processing chain was developed consisting of image processing algorithms, a knowledge extraction process, and a multilayer perceptron (MLP) classification model.
Results: Different configurations of the MLP have been investigated and quantitatively compared. The accuracy of our approach is high, obtaining an accuracy of 87%. The model was compared with five well-known classification paradigms. The performance of a reduced model, obtained by through feature selection algorithms, was found to be better and less time-consuming than the original model. The whole proposal has been validated with real patients in pre-clinical tests using the final medical bed prototype.
Conclusions: The proposed approach produced very promising results, outperforming existing classification approaches. The excellent behaviour of the recognition system will enable its use in controlling the movements of the bed, in several degrees of freedom, by the patient with his/her own body.

1. Introduction

The recognition of movement intentions to control medical beds can significantly improve the patient wellbeing in hospitals. Patients, who are physically handicapped and are restricted to stay in bed for long periods of time, need frequent changes to avoid the side effects of a static position (e.g. scares). Currently, patients are moved in hospitals by nurses and health care personnel, also by relatives. Moving a patient requires significant physical effort and causes back, neck, and shoulder injuries in nurses and support staff. For instance, some sources estimate that 85% of nurses will suffer back injury at some point in their careers [1].

Consequently, there is a social demand for the development of a medical bed system that will allow patients with reduced mobility to undertake the most common postural changes, both lateral and sitting, with a natural communication between patient and bed. Previous experiences aimed at improving the living conditions of long-stay bed patients failed. For example, existing cabled or wireless remote controllers are not accessible for people with reduced mobility, and voiced commanded interfaces are too imprecise, require adaptation, and may also become confused by ambient noise in the hospital room [2].

The medical bed system we propose here is based on the use of a pressure matrix laid on the bed. Pressure sensors distributed on the matrix will collect the pressure exerted by the body of the

* Corresponding author at: Inspiralia Tecnologías Avanzadas, Estrada 10, 28034 Madrid, Spain. Tel.: +34 914170457; fax: +34 915563415.
E-mail addresses: manuel.chica@softcomputing.es (M. Chica), pascual.campoy@upm.es (P. Campoy), maria.perez@itav.es (M.A. Pérez), tomas.rodriguez@itav.es (T. Rodríguez), ruben.rodriguez@itav.es (R. Rodríguez), ovaldemoros@spaldin.com (Ó. Valdemoros).
URL: http://eposbed.pera.com (M. Chica).

0010-4825/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compbiomed.2011.12.003

patient. This information must be acquired and processed in real-time with the aim at detecting the user's intention to move. Feedback control is smooth and continuous, resulting in a natural interface between the bed and the patient. The user only needs to perform slight intentions in the usual way, and the bed will follow his/her movements.

However, in order to achieve the final medical bed, a recognition system to detect and classify the movement intentions has to be developed. This is not a simple task because not all the patients realise their movement intentions in the same way. In addition, the system is forced to work with many patients with different somatotype characteristics, and distinct ways of communication intentions, even for the same patient. Several references and research can be found in the literature addressing the analysis and classification of static pressure maps on beds and chairs [3–7]. Nevertheless, there is not any research to recognise, continuously and in real-time, intentions from the pressure maps. And certainly, neither of them is devoted to develop a classification system to serve as a natural bed control interface.

The goal of this paper is to develop a novel recognition system able to transform sequences of pressure maps from sensors information into intentions of movement, valid and robust for the majority of patients. We propose a recognition system based on artificial neural networks (ANNs) and with a complementary processing chain composed of segmentation, image analysis algorithms, and an extraction process of spatial and dynamic features. A multilayer perceptron (MLP) has been chosen as the ANN classification model. The reasons for this choice were: (a) the excellent prediction performance of ANNs in a wide range of applications, from control to business contexts [8–10]; (b) reliable references in many different medical applications such as early detection of health condition in patients [11], analysis of employment history as a health risk factor [12], classification of cardiac valve disorders [13] or medical image analysis [14]; (c) their use in similar applications [7,6]; and (d) the low computational requirements of the MLP that will allow us to embed the system into the low-cost electronics of a medical bed. Different configurations of MLPs were designed and tested to find the most suitable parameters values.

In addition, we applied feature selection methods to reduce the complexity of the initial model and the requirements of the system for its future integration with the medical bed. Furthermore, and to prove the strength of our approach, we ran a complete set of pre-clinical tests with many different patients. Thanks to this experimentation, we have the ability to compare our results against well-known classification techniques. These classifiers also belong to different paradigms: classification rules, decision trees, instance-based classifier, probabilistic functions and ensemble of classifiers.

The paper is structured as follows. In Section 2 we present a general description of the medical system as well as a study of related work. Next, in Section 3, an overview of the complete model to solve the recognition problem is given. In Section 4, the image analysis, the spatial and dynamic feature extraction, and the feature selection algorithms are described. The ANN-based classification model is discussed in Section 5. The method and materials used for the database collection and experimentation, the results, and their analyses are depicted in Section 6. Finally, some concluding remarks and suggestions for further improvements are discussed in Section 7.

2. Preliminaries

In this section we first introduce the innovative medical system for which the recognition system has to be designed. Finally, some existing state-of-the-art approaches to solve similar problems are reviewed.

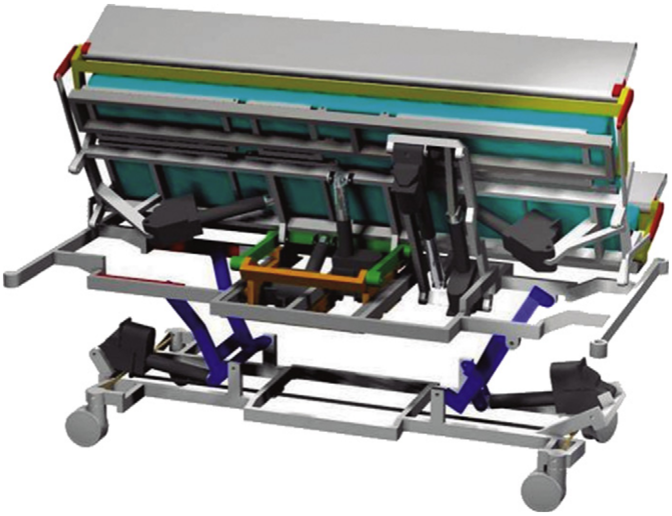

Fig. 1. The CAD model of the mechanical bed that was designed and used for this work.

2.1. System and problem description

The proposed innovative medical bed is a complex system composed of different elements such as a sophisticated bed frame with Fowler[1] and lateral positioning, nine motorised actuators to achieve the complex positioning, a pressure matrix embedded into the mattress, and a low cost microprocessor and electronics for embedded software [see the computer-aided design (CAD) model of the bed in Fig. 1].

Pressure maps are obtained from the pressure matrix consisting of 1664 capacitive pressure sensors (64 columns and 26 rows) fixed to the bed, having a sampling frequency of 10 pressure maps per second and a pixel depth of 2 bytes. The actuators of the bed are controlled by the bed electronics according to the decision made by the software. A diagram of a general view of the system with the three main components, i.e. pressure sensors, intelligent software, and actuator system of the mechanical bed, is shown in Fig. 2.

The most critical element in the new medical bed is designing the intelligent software, which is the scope of this paper. As addressed in Section 1, the interpretation of patient intentions is complex because of many reasons. Patients with reduced mobility are not able to perform abrupt movements, only slight and natural intentions. It is also clear that recognising intentions in a sequence of pressure maps for a great number of different patients with different somatotype characteristics is not an easy task. Furthermore, the same patient may present distinct ways of communicating his/her intentions at different times. Also, the difficulty of the problem is worsened by the fact that bed configurations may change with the movements of the patient thus changing also their influence on the resulting pressure map.[2] In Fig. 3 we can observe four pressure maps of the same movement intention (*right* movement intention) which are very different to each other.

[1] Fowler position is a medical standard position. The patient is placed in a semi-upright sitting position (45–60°) and may have knees either bent or not. There are several types of Fowler positions, as low, semi, and high [15].

[2] This problem will not be tackled in this work since its solution involves the use of the actuators of the medical bed and the control system which will be designed in future work.

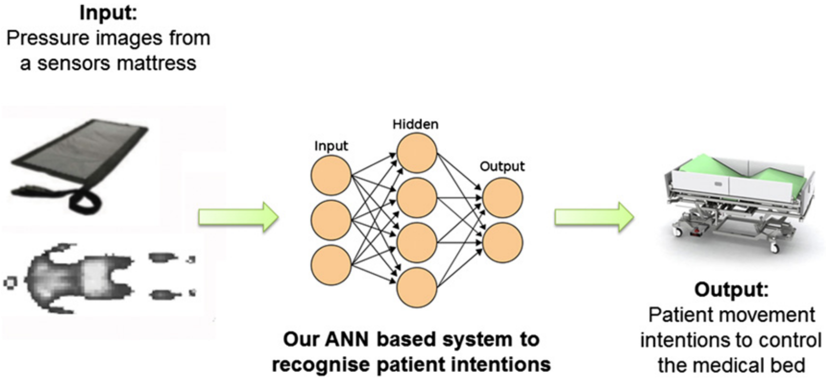
Input:
Pressure images from a sensors mattress
Input
Hidden
Output
Our ANN based system to recognise patient intentions
Output:
Patient movement intentions to control the medical bed

Fig. 2. Overall view of the proposed medical bed with their main components. From the pressure maps (input information), the recognition software is able to identify the movement intention of the patient and move the bed consequently (output of the system). The recognition software proposed in this work will be used embedded into the control electronics.

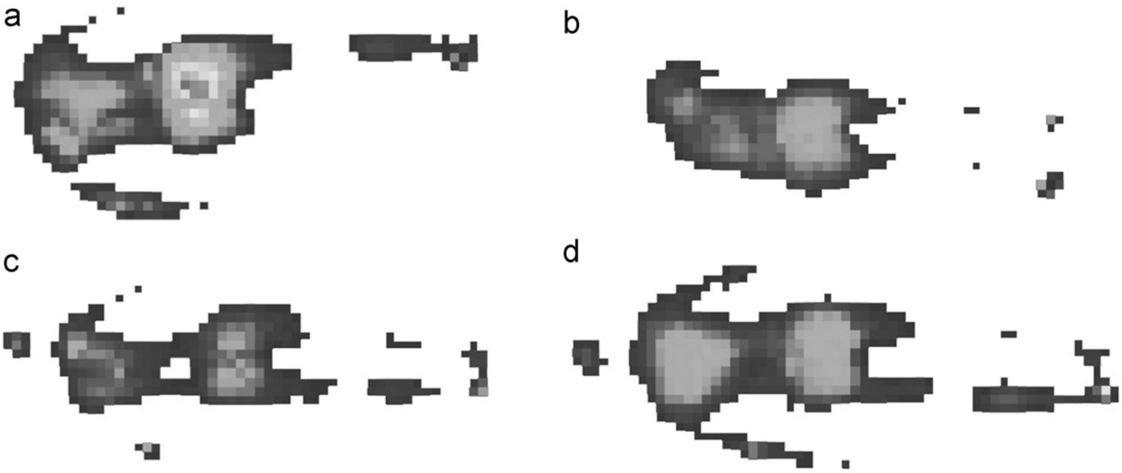

Fig. 3. Four pressure maps from four different patients performing the same intention of movement (*right* intention). Note the high variability in the patients' patterns.

The recognition software devoted to identify patient intentions has also to be embedded into a low-cost microcontroller. It will be able to interpret sequences of pressure maps (10 fps.) obtained from the sensors matrix for a wide variety of patient complexions, with sizes ranging from 1.40 m to 2.00 m, and weights from 45 kg to 120 kg. Basically, the software must be able to recognise five movement intentions in real-time:

- *rest*: the patient wants to stay in the current position.
- *sit up*: the patient needs to be incorporated from a supine position to a Fowler or semi Fowler position.
- *lean back*: starting from a sitting position, the patient needs to come back to a resting position, leaning back on the bed.
- *right*: the patient wants to be laid on his/her right side.
- *left*: the patient wants to be laid on his/her left side.

2.2. Related work

Although we have highlighted the existence of related work in the literature (Section 1), there is not any similar work where an intention of movement has to be recognised real-time from pressure maps. In this section we will discuss the developments done in the closest works. These works can be divided in two main groups depending on the application: (a) beds and (b) chairs or seats.

In the first case, many articles involving the use of pressure sensors attached to a bed have been found. For instance, in [5] authors developed a automated system able to estimate the pose of a patient when he/she is in supine, right lateral, left lateral, or sitting position on a pressure mattress. They proposed a system composed by a dimensionality reduction step using principal component analysis (PCA) and an ANN model to classify the input features. However, they only addressed static pressure maps at a determined instant. Also, the behaviour of the system using different patients is not robust and there is not any comparison against other methods.

A system to estimate the posture of the patient and some abrupt movements is proposed in [16]. It is based on a template construction and a matching process between the saved templates and the new incoming pressure instance. No quantitative methods for experimentation are provided. Other authors have worked with pressure information to estimate respiratory rate [17] or infant behaviour [18]. In both cases, there is not any comparison methodology, and there is a lack of estimation accuracy.

However, we can find more complete and interesting research regarding the use of pressure maps on chairs and seats. Tan et al. [3] applied PCA to solve the problem of classifying sitting postures. The system is able to perform a static posture estimation with an accuracy of 96% for familiar users and 79% for unfamiliar users. Another recognition system for seated postures is proposed in [6]. In this case, problem-dependent features that measure geometric and physical variabilities are extracted from pressure maps. Then, four different classifiers are compared to each other: simple logistic, naive Bayes (NB), ANNs, and support vector machines (SVM). The simple logistic algorithm achieved the best performance (82%) followed by the ANN with an

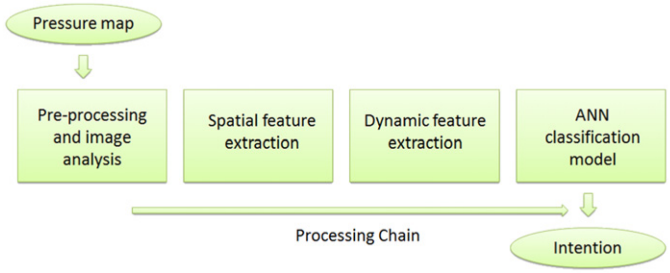
Pressure map
Pre-processing and image analysis
Spatial feature extraction
Dynamic feature extraction
ANN classification model
Processing Chain
Intention

Fig. 4. Spatial features and dynamic features (evolution of the spatial features) are calculated from a sequence of pressure maps. The ANN classification model is in charge of recognise intentions (classes) from the dynamic features.

accuracy of 79%. Unfortunately, the recognition of dynamic intentions is postponed for future work.

A complete sensing seat system for human authentication is developed in [7]. Trucks, cars, or security in offices are referred as possible applications. The compared classification methods were: the minimum distance classifier, SVM, PCA, probabilistic ANN, MLP, and Kohonen self-organising maps (KSOM). Authors claimed that efficiency, in terms of memory space and computing time, has to be taken into account. Thus, they decided to employ the MLP and KSOM classifiers instead of SVM and PCA. The MLP achieved the best accuracy: 93% for recognition and 96% for rejection. However, the system needed the cooperation of the subject in order to work properly.

Finally, some posture analysis on chairs for recognising a child's interest level is done in [4] where ANNs and hidden Markov models are used. The obtained accuracy is 76% when tested with real data.

3. The proposed system for recognising patient intentions

In this section, we describe our proposed solution for recognising the patient intentions. As stated, this model will be used for the medical bed described in Section 2.1. An overview of the processing chain is presented in Fig. 4. We start with a segmentation step aimed at identifying the location of body parts in a single image. Next, we apply a feature extraction process to identify distinguishing characteristics in the segmented body parts (spatial features). Each pressure map in the sequence is analysed with the aim of extracting statistical and physical features of the most important body parts.

The evolution of the features between a pair of pressure maps separated by a determined time window is computed (dynamic features). The latter values are the input of the ANN-based classification model. The classification model will need to identify transitions or changes between these features at different time instants. The output of the model will be one of the five classes explained in Section 2.1. An example describing how the system performs is presented in Fig. 5.

With respect to the time window used to calculate the dynamic features, it is valuable to remember that the high frame rate (10 fps) provided by the physical pressure matrix is over killing in this application. This is because of the evolution between two consecutive pressure maps are almost imperceptible. For that reason, we maintain the frame rate of 0.1 s but a definition of a time window of 1 s is needed.[3] Consequently, our system computes the evolution of two pressure maps, separated by one time window, every 0.1 s. This approach provides a good compromise between time resolution and responsiveness.

In the next sections we will analyse each component of the system in detail: the preprocessing and segmentation algorithms, the feature extraction process, and feature selection algorithms to reduce the complexity and size of the system (Section 4), and the ANN classification model (Section 5).

4. Image processing and feature extraction

First, the image processing algorithms used to analyse the pressure maps are described. In the next sections, the complete feature extraction process is explained in detail, i.e. spatial features from each pressure map and dynamic features to measure the evolution of the latter. In Section 4.4 we also discuss about the feature selection algorithms used to reduce the complexity of the system.

4.1. Pre-processing and image segmentation

The first pre-processing step is achieved by thresholding the incoming pressure map and is done using a known minimum value. Those pressure values that are lower than the fixed value come from the electrical noise of the sensors and from external pressure like the fabric of the mattress. This threshold provides good results with all the analysed sequences. Next, we apply a 2×2 median filter to reduce noise and make the pressure maps smoother. This simple noise removal approach proved sufficient in an environment where noise is typically caused by defective sensors only.

The most important process in this part of the system is the segmentation algorithm to locate the different body parts. We have taken advantage of prior medical knowledge and experts to identify the most important body parts that influence the classification decision. The algorithm is able to create segments within the pressure map in seven body parts: (a) head, (b) back, (c) glutei, (d) left elbow, (e) right elbow, (f) left heel, and (g) right heel.

The applied segmentation algorithm is mainly based on pressure profiles analysis and region growing algorithms. Two pressure profiles are calculated for each incoming pressure map. The first is obtained by adding the pressure of the columns of the pressure map. The second is generated from the rows addition. See Fig. 6 where two pressure profiles have been calculated from the pressure map of Fig. 7. The vertical lines are the boundaries of the bounding boxes and were calculated by the algorithm.

The algorithm starts analysing the horizontal pressure profile in order to divide the profile into four main areas: head, back, glutei and heels (see the first image profile of Fig. 6). This is done by the following steps:

- First, the algorithm looks for patient's glutei. It localises the highest pressure region and the valleys that enclose the glutei.
- When the glutei region is identified, the algorithm searches for the back. Starting from the first column of the glutei in the horizontal profile, the algorithm goes backward till a pressure fall is found. This pressure fall corresponds to the neck, if the head's pressure exists, and also to the first column of the back in the profile.
- The head can be in the pressure map or not. Using a region growing algorithm, we extract the pressure area that is enclosed by the beginning of the whole pressure map and the first column of the back region, calculated in the previous step.
- Finally, the aim is to find the horizontal profile columns where heels are. This is done by obtaining the pressure area of the horizontal pressure profile that is between the end of the pressure map and the end of the glutei area.

[3] The value of 1 s was set after a preliminary experimentation on the most suitable value.

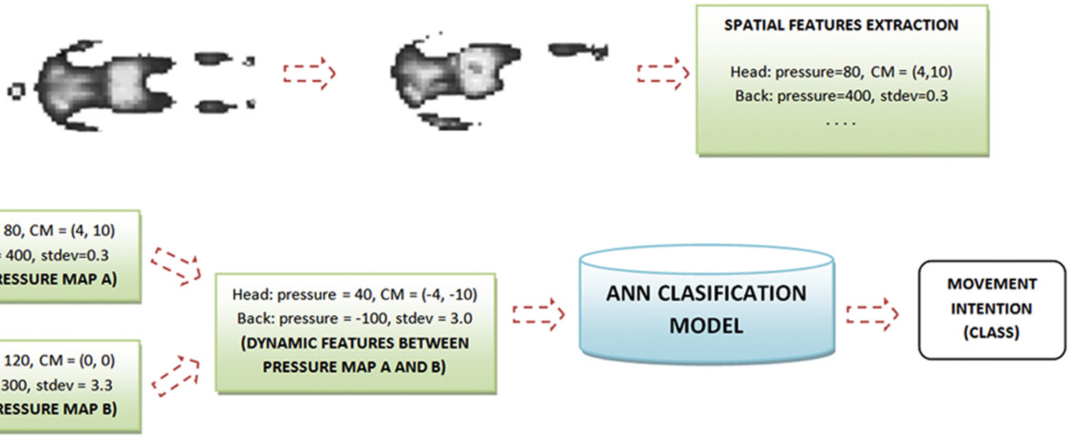
SPATIAL FEATURES EXTRACTION
Head: pressure=80, CM = (4,10)
Back: pressure=400, stdev=0.3
. . . .
Head: pressure = 80, CM = (4, 10)
Back: pressure = 400, stdev=0.3
(FEATURES OF PRESSURE MAP A)
Head: pressure = 120, CM = (0, 0)
Back: pressure = 300, stdev = 3.3
(FEATURES OF PRESSURE MAP B)
Head: pressure = 40, CM = (-4, -10)
Back: pressure = -100, stdev = 3.0
(DYNAMIC FEATURES BETWEEN PRESSURE MAP A AND B)
ANN CLASIFICATION MODEL
MOVEMENT INTENTION (CLASS)

Fig. 5. A diagram showing an example of the processing chain of the proposed recognition system. From the initial input, features are calculated to obtain the desired output by the ANN-based model, i.e. the class associated to one of the movement intentions.

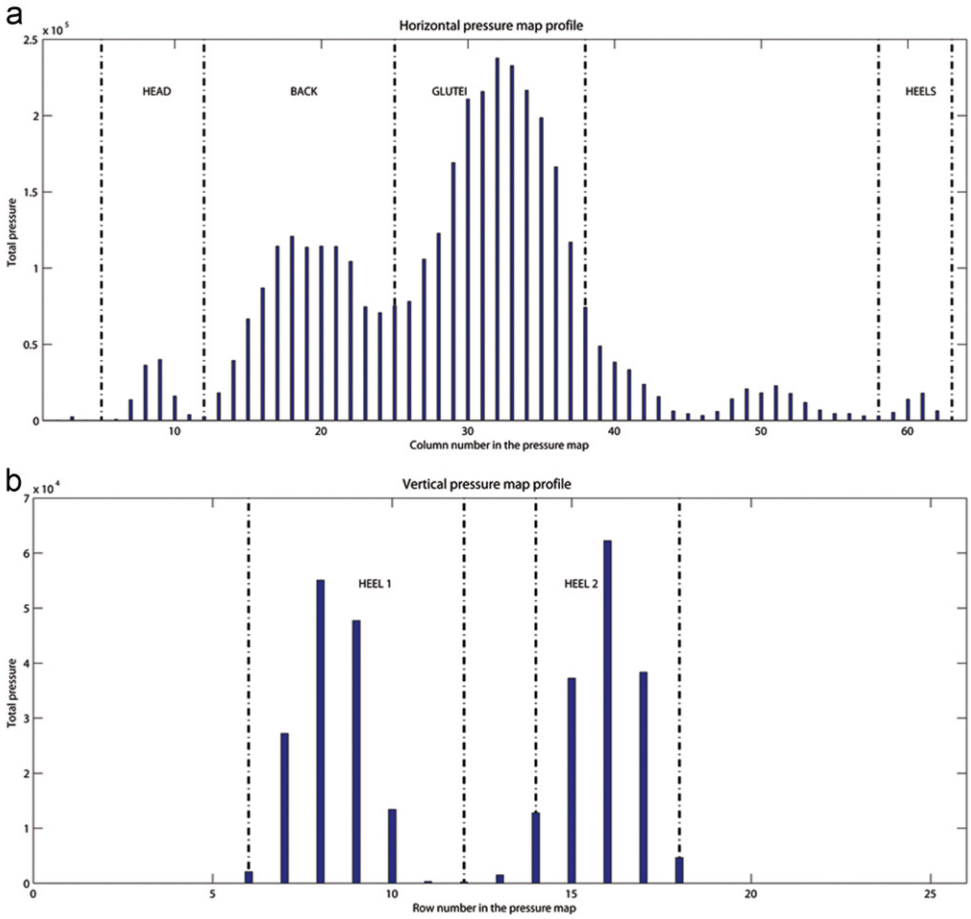

Fig. 6. Vertical and horizontal profiles of the pressure map of Fig. 7. The boundaries of the different body parts (vertical lines) have been calculated by the segmentation algorithm.

At this point, the algorithm has the column boundaries of the four areas. Now, the segmentation algorithm needs to divide the heels' area into two different heels and extract elbows from back and glutei areas by obtaining the rows of the vertical pressure profile. The algorithm works as follows:

- The complete heels segmentation must take into account that sometimes both heels are merged in only one pressure area. Also, the algorithm needs to discover if one or two heels are missing. The local minima are found by the algorithm to perfectly segmentate the heels. See the second image of Fig. 6 where the vertical pressure profile of the heels area is computed as well as the found boundaries.
- Later, elbows are separated from the trunk (back and glutei) of the body by a region growing algorithm which will segmentate the main pressure area from the pressure belonging to elbows and hands.

When the segmentation algorithm has finished, the corners' coordinates of each bounding box are localised. An example of

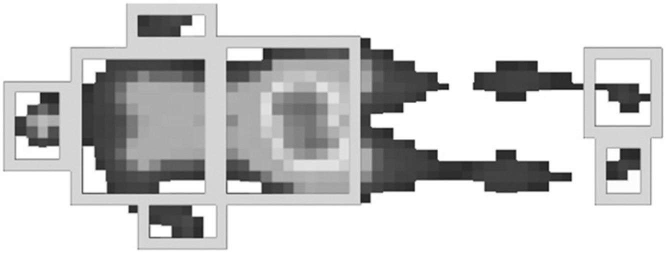

Fig. 7. A segmented pressure map after the application of the image analysis algorithm. The bounding boxes are superimposed into the pressure map. The profiles of Fig. 6 were obtained from this pressure map.

segmented pressure map can be seen in Fig. 7. This information will be the input of the spatial feature extraction algorithm. As we will see in the experimentation section, the robustness of the whole system and therefore, the segmentation algorithm, is very high. Also, the use of a segmentation algorithm mainly based on pressure profiles will guarantee us a low computational time which is a pre-requisite of the system.

4.2. Spatial features

The initial features of the pressure maps have been decided after a medical study comprising thousands of pressure map records. These features must be extracted from different parts of the patient body and are one the most critical elements of the processing chain. If these features are not related with the patient intentions, the classifier model will not be able to identify the slight movements.

After an extensive medical study we have initially considered 25 features. Note that we will refer to a body part as the section or subimage of the pressure map enclosed by one of the bounding boxes calculated by the segmentation algorithm. The 25 spatial features are:

- **Total pressure**: Where p_k is the pressure value of each pixel k in the bounding box B:

$$TP = \sum_{k \in B} p_k \tag{1}$$

  There are seven features of this type, one for each body part: TP_{head}, TP_{back}, TP_{elbow1}, TP_{elbow2}, TP_{glutei}, TP_{heel1} and TP_{heel2}.
- **Centre of mass**: The centre of mass is defined as the pair of coordinates (CM_x,CM_y):

$$CM_x = \frac{\sum_{k \in B} p_k x_k}{\sum_{k \in B} p_k} \quad CM_y = \frac{\sum_{k \in B} p_k y_k}{\sum_{k \in B} p_k} \tag{2}$$

  being (x_k,y_k) the coordinates of the pressure pixel. There are seven CM features, one for each body part: CM_{head}, CM_{back}, CM_{elbow1}, CM_{elbow2}, CM_{glutei}, CM_{heel1} and CM_{heel2}.
- **Pressure distribution**: This represents how the pressure area is distributed with respect to the centre of mass. A similar concept was used in [19] to detect edges and corners in images.
- We first compute the variance, d_x and d_y, and the covariance d_{xy} in both spatial axis:

$$d_x = \frac{\sum_{k \in B} (x_k - CM_x)^2 p_k}{\sum_{k \in B} p_k} \tag{3}$$

$$d_y = \frac{\sum_{k \in B} (y_k - CM_y)^2 p_k}{\sum_{k \in B} p_k} \tag{4}$$

$$d_{xy} = \frac{\sum_{k \in B} (x_k - CM_x)(y_k - CM_y) p_k}{\sum_{k \in B} p_k} \tag{5}$$

  Next the covariance matrix *cov* is formed from the latter values:

$$\mathbf{cov} = \begin{pmatrix} d_x & d_{yx} \\ d_{xy} & d_y \end{pmatrix}$$

  Finally, we calculate the eigenvalues λ_1 and λ_2 of the *cov* matrix and summarise this information by computing the ratio r between both eigenvalues:

$$r = \frac{\lambda_1}{\lambda_2} \tag{6}$$

  From the computation of the eigenvalues of the covariance matrix we can obtain the predominant direction of the pressure map region. For instance, if the pressure distribution value $r \to \infty$, the predominant direction of the pressure is the one associated to the eigenvalue λ_1. If $r \to 0$, then the pressure distribution direction is associated to λ_2. Otherwise, if $r \to 1$, the pressure distribution is anisotropic, and there is no principal direction.
  This value is computed for the back and glutei, r_{back} and r_{glutei}, since it was observed that can be quite important when the patient intends to communicate his/her intentions.
- **Pressure variation**: This feature will measure the standard deviation of the pressure in the bounding box:

$$s = \frac{\sum_{k \in B} (p_k - \overline{p})}{N} \tag{7}$$

  where N is the number of pixels in the bounding box B, p_k the pressure value at each pixel k, and $\overline{p}$ is the mean value of the pressure. Again, this feature is computed only for the glutei and back body parts: s_{back} and s_{glutei}.

Table 1
Pool of the 25 spatial features to be used in our proposal. For each incoming pressure map, the feature extraction algorithm is applied to calculate them from the bounding boxes of the pressure map.

Head	Back	Elbows	Glutei	Heels
TP_{head}	TP_{back}	TP_{elbow1} TP_{elbow2}	TP_{glutei}	TP_{heel1} TP_{heel2}
$CM_{head}(x)$	$CM_{back}(x)$	$CM_{elbow1}(x)$ $CM_{elbow2}(x)$	$CM_{glutei}(x)$	$CM_{heel1}(x)$ $CM_{heel2}(x)$
$CM_{head}(y)$	$CM_{back}(y)$	$CM_{elbow1}(y)$ $CM_{elbow2}(y)$	$CM_{glutei}(y)$	$CM_{heel1}(y)$ $CM_{heel2}(y)$
	r_{back} s_{back}		r_{glutei} s_{glutei}	

We summarise the complete set of spatial features extracted for each pressure map in Table 1.

4.3. Dynamic features

The input of the classifier is obtained by calculating the evolution of the spatial features between the initial and final frames of a time window (1 s apart). The relative increment between two features F_1 and F_2, ∇_F, is calculated as follows:

$$\nabla_F = \frac{F_2 - F_1}{\frac{F_2 + F_1}{2}} \tag{8}$$

This formula was used in order to make the dynamic features non-dimensional. In some cases, some spatial features can be missing in some pressure maps because of the seven body parts do not always appear. If one spatial feature is missing, the computation of the corresponding dynamic feature is not done and the evolution of the feature is null, setting ∇_F to 0. The system continues processing the remaining dynamic features. The system is robust to these missing features, as it will be shown in Table 5, where many patients do not have all the features and the missclassified instances are very low.

Finally, a normalisation algorithm (z-score standardisation) is applied to the computed dynamic features to avoid the effects of having different normal distributions.

4.4. Feature selection

Unfortunately, although the set of 25 features explained in Section 4.2 was chosen after a complete medical study, it is usual to have irrelevant features and useless information that degrade the performance of the models both in speed, due to the high dimensionality, and accuracy, due to irrelevant information [20]. Feature selection has the aim of choosing the smallest possible subset of features ideally necessary to describe a problem. In other words, it can be defined as a search process of P features from an initial set S of N variables, with $P \leq N$. It aims at eliminating irrelevant and/or redundant features and to obtain a simpler classification system. In some problems, feature selection results not only in faster performance, but also in more accurate classification than using the whole set [21].

For a feature selection problem, the specific goal is trying to reduce the initial pool of 25 features with just the most important, without affecting the overall performance of the classification model. To achieve this objective, we have applied a feature selection algorithm based on ranking the most important features. We experimented with two evaluation measures to achieve this ranking:

- Relief [22]: The general idea of this method is to choose the features that can be most distinguished between classes. These are known as the relevant features. At each step of an iterative process, an instance x is chosen at random from the dataset and the weight for each feature is updated according to the distance of x to its *nearmiss* and *nearhit*.
- Gain ratio: It evaluates the worth of a feature by measuring the information gain ratio with respect to the class. Thus, it is a feature selection algorithm based on information theory and information gain (a variation of the MIFS algorithm [23]).

The results of applying the previous feature selection algorithms and the reduced set of features is described in Section 6.3.3 of the experimentation.

5. The ANN classification model

The general structure of the ANN is given in the first section. In the second section, the different MLP configurations are shown.

5.1. ANN paradigm

ANNs are a collection of simple, interconnected nodes, which operate in parallel and store knowledge through connection weights between adjacent nodes. There are many types of ANNs [24]; including probabilistic ANNs, radial basis function networks, learning vector quantisation, etc.

Here, we have adopted a MLP approach which offers significant advantages in terms of computational requirements, memory usage, and time responsiveness. As already presented, they have been used extensively in many different applications including the most related with our problem (see Section 2.2). This type of ANN has a layered, feed-forward network topology with its neurons connected in such a manner that there are no feedback loops. The MLP represents a deterministic mapping from the inputs to the outputs, and it is known that, with enough hidden neurons, the mapping exhibits the property of universal function approximator [25].

Table 2
Common parameter values used for all the MLP configurations.

Parameter	Value
Learning rate η	0.3
Maximum number of epochs	5000
Transfer function	*tan-sigmoid*

Inside the neuron, a transfer function σ transforms the input value into the output of the neuron. The number of neurons in the input layer is given by the number of dynamic features (25 features), whereas the number of neurons in the output layer is equal to the number of movement intentions to be recognised (five intentions). In addition to the input variables, there is a constant input of 1.0, called the bias, which is fed to each of the hidden layers. The bias is multiplied by a weight and added to the sum going into the neuron. The final output of the classifier system, in response to the applied input vector, is obtained by selecting the intention in the output vector having the value closest to 1.

There are several important issues involved in the design of a MLP. For example, the generalisation of the ANN, which is improved by using the 20% of the training data to validate the model while it is being trained. This validation set is used to detect the point when its performance begins to decrease, which signals the fact that generalisation has achieved a peak. Therefore, training on the input data continues as long the training does not increase the model's error on the validation data. When the validation error increases, the training is stopped. If the validation error never increases, the other stopping criteria is reaching the maximum number of epochs.

However, there are also issues which include critical factors such as the selection of the number of hidden layers, the number of neurons in each layer, or the method used to find a globally optimal solution that avoids local minima. We have compared different values for these elements, designing 10 MLP configurations that are described in the next section.

5.2. MLP configurations

The common parameters of all the MLP configurations are shown in Table 2.

As commented in the previous section, 10 different MLP configurations have been implemented with different values for the number of hidden layers, the number of neurons in the hidden layers, and the algorithm used for the MLP training. Consequently, we have combined the different configuration values in 10 different algorithms. The resulting algorithms and their specific parameters are detailed in Table 3. A comparison among them is given in Section 6.3.1.

6. Experimentation

Firstly, the process to generate the database and validate the model is shown. Then, the experiment conditions and the classification models to be compared with our proposal are explained. Finally, we present the results of the experimentation and analyse them.

6.1. Data collection and database generation

Sequences of pressure maps were recorded using real patients laid on the medical bed prototype (see the real experimentation environment in Fig. 8). Our objective was to create the database to

Table 3
The different parameter values of the 10 MLP configurations compared in this work.

Algorithm	Number of hidden layers	Number of neurons in the hidden layers	Training algorithm
MLP#1	1	10	Levenberg-Marquardt
MLP#2	1	15	Levenberg-Marquardt
MLP#3	2	{12, 18}	Levenberg-Marquardt
MLP#4	1	20	Levenberg-Marquardt
MLP#5	1	25	Levenberg-Marquardt
MLP#6	1	10	Bayesian regulation
MLP#7	1	15	Bayesian regulation
MLP#8	2	{12, 18}	Bayesian regulation
MLP#9	1	20	Bayesian regulation
MLP#10	1	25	Bayesian regulation

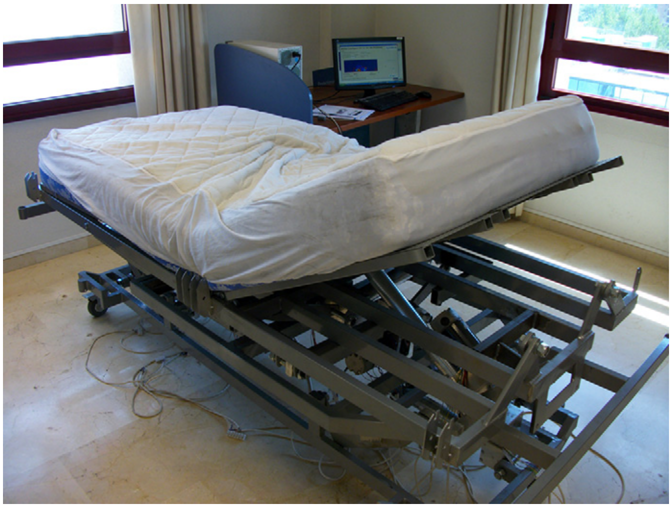

Fig. 8. The mechanical bed used in the experimentation. Twenty different patients helped in the data collection, obtaining 1154 final instances to train and test the model.

train and test the behaviour of the proposed recognition system independently from other external influences like the actuators response or the changes in the bed position. Therefore, every movement intention was captured using a fixed horizontal bed position.

Patients were asked to perform their different movement intentions to be processed afterwards by the training algorithm. The movement intentions were repeated several times by the same person at different times to achieve a good representation of the inherent characteristics of each movement. After the recording stage, these pressure sequences were tagged with the corresponding intentions. Then the sequences were processed following the algorithms described in Section 4. We obtained and stored the extracted dynamic features in a database.

The database contains information from a very diverse pool of patients with different somatotype. They have been selected from a representative group of the population covering different ages, reduced mobility, weights, from 55 kg to 100 kg, and heights, from 1.60 m to 1.90 m. Women and men were present in equal terms in the sample. On the whole, validation data were collected from 20 patients. The collection data were converted into 1154 instances or movement decisions for the four basic movements or classes, i.e. *sit up*, *lean back*, *right*, and *left*, and the *rest* position. For each patient, we built instances for movement classes and for the *rest* class at different moments. An intention of movement usually takes 15 pressure maps or 1.5 s. Training and test instances were formed by calculating the 25 features differences defined in Section 4.2.

6.2. Experimental protocol

In order to test the quality of our MLP design, it was compared with five well-known classifiers using the same database and performance indicators. The classification models belong to different paradigms: classification rules, decision trees, instance-based classifier, probabilistic functions and ensemble of classifiers:

- OneR: We found some references in the literature that state that simple rules may achieve surprisingly high accuracy on many datasets [26]. Therefore, we wanted to compare the performance of our soft computing approach with a simple and effective rule system. We used the OneR algorithm [27] which creates one rule per class and classify instances on the basis of a single attribute, called “1-rules”. The rule extraction algorithm treats all numerically valued attributes as continuous and uses a straightforward method to divide the range of values into several disjoint intervals.
- C4.5: This is an algorithm introduced by Quinlan [28] for inducing decision trees from data. It determines a decision tree that, on the basis of answers to questions about the non-category features, predicts correctly the value of the class. C4.5 can manage continuous and unknown features. We run the algorithm with a pruning mechanism with three folds and a confidence factor of 0.25.
- *k* nearest neighbours (kNN): this is a instance-based classifier [29] that classify new instances into a class comparing them with the nearest instances of the training data (nearest neighbours). The most common is the 1NN which uses the closest training instance to compare.
- Naive Bayes (NB): The NB classifier technique [30] is based on conditional probabilities and uses the Bayes’ theorem. This is a formula that calculates a probability by counting the frequency of values and combinations of values in the historical data. It is particularly well suited when the dimensionality of the input is high. Despite its simplicity, NB can often outperform more sophisticated classification methods.
- AdaBoost: AdaBoost is an algorithm for constructing a “strong” classifier as linear combination of “weak” classifiers. It has been widely used with C4.5 as its classifier [31] showing excellent results in many problems.

Cross-validation is used to evaluate and compare the designed models of this study. It is a well-known statistical method which is based on dividing data into two segments: one used to learn or train a model and the other used to validate the model. In typical cross-validation, the training and validation sets must cross-over in successive rounds such that each data point has a chance of being validated against. The most common form of cross-validation is *k*-fold cross-validation with $k=10$. This is the method used in this paper.

In 10-fold cross-validation the data is first partitioned into 10 equally (or nearly equally) sized segments or folds. Subsequently 10 iterations of training and test are performed such that within each iteration a different fold of the data is withheld for validation while the remaining nine folds are used for learning. In our case, at each iteration the models were tested with instances from completely different patients from those used in the training of the model. For all the shown experiments, the performance of the models is presented as the 10-fold cross-validation accuracy (correctly classified intentions divided by the total number of classified intentions).

Table 4
Results obtained by the 10 different MLP configurations described in Section 5.2 for the 10 data partitions (accuracy in %). Mean ($\bar{x}$) and standard deviation (σ) are shown in the last two rows of the table.

Data partitions and global measures	MLP configurations									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
P1	71.62	74.32	91.89	66.22	64.86	71.62	79.73	71.62	72.97	74.32
P2	41.03	55.13	67.95	53.85	70.51	52.56	53.85	52.56	58.97	60.26
P3	83.33	73.08	92.31	87.18	80.77	89.74	94.87	96.15	93.59	94.87
P4	83.33	93.59	74.36	87.18	82.05	93.59	93.59	87.18	93.59	93.59
P5	83.93	76.79	78.57	75.89	83.04	77.68	80.36	84.82	86.61	84.82
P6	89.29	88.39	86.61	86.61	91.96	95.54	95.54	94.64	94.64	97.32
P7	86.61	80.36	89.29	86.61	81.25	91.07	87.5	88.39	87.5	82.14
P8	83.04	93.75	90.18	83.04	94.64	93.75	90.18	92.86	95.54	96.43
P9	89.29	93.75	91.96	92.86	89.29	89.29	90.18	97.32	93.75	91.96
P10	87.8	82.93	81.30	83.74	84.55	86.18	86.99	85.37	87.8	86.99
$\bar{x}$	79.93	81.21	84.44	80.32	82.29	84.10	85.28	85.09	**86.5**	86.27
σ	14.58	12.19	8.52	11.87	9.11	13.4	12.32	13.66	**11.76**	11.66

The experiments included in this paper were conducted on Intel Pentium ^{TM} Core 2 Duo at 2 GHz and 3 Gb of RAM memory, and Windows XP as operating system.

6.3. Results and discussion

We present the experimentation results and the analysis of them. First, we include a comparison between MLP configurations. Then, the MLP is compared against five classification paradigms and finally, the result of applying feature selection algorithms is discussed.

6.3.1. Analysis of the different MLP configurations

The accuracy in the detection of the movement intentions obtained in the different MLP configurations for the 10 data partitions is shown in Table 4. Mean and standard deviation are also specified in the last two rows.

The analysis of the experiments for the different MLP configurations showed that:

- Best results are obtained by the MLP#9 configuration, which has 20 neurons in one hidden layer and was trained by the Bayesian regulation algorithm.
- The Bayesian regulation algorithm always outperforms Levenberg–Marquardt configurations.
- Twenty neurons in one hidden layer seem to be sufficient to solve the problem. Increasing the number of neurons in the hidden layers does not seem to offer any significant advantage; i.e. a MLP with 25 hidden nodes achieved lower accuracy because of a poorer generalisation.

According to the results we have chosen the MLP#9 as our reference ANN configuration since it has best performance, using one hidden layer with 20 neurons. Its confusion matrix obtained after classifying the test data of the 10 partitions (all instances in the database) is shown in Table 5.

6.3.2. Comparison with other classification models

In Table 6, the results of the comparison between our MLP algorithm with the other five classification techniques are shown. The analysis of the results showed the following facts:

- The most important conclusion is that our ANN-based system achieves the best performance, being around 3% more accurate than the second classifier in performance, AdaBoost. The difference with respect to the other classifiers is much higher, outperforming C4.5 with an accuracy difference of 13%.
- The MLP offered the best results in all partitions except for the P2 and P7 partition. The second best technique in the list is the AdaBoost algorithm which is the best in the latter two partitions. Although the performance of AdaBoost is worse than our model, it seems to be much better than the other four comparison techniques.
- The baseline classifier, OneR, and NB do not obtain good performance in the experiments (55.42% and 68.69%), demonstrating the complexity of the classification problem.

Table 5
Confusion matrix of the MLP#9 classified instances. Accuracy is shown in brackets in %. The values are obtained from the sum of the classified instances of the 10 test datasets. We can also observe the robustness of the segmentation algorithm and the feature extraction process in the high classification accuracy.

Real classes						
rest	*sit up*	*lean back*	*right*	*left*		
294 (**95.77**)	6 (1.95)	4 (1.3)	1 (0.33)	2 (0.65)	***rest***	**Predicted**
7 (5.34)	**113** (**86.26**)	0 (0)	9 (6.87)	2 (1.53)	***sit up***	
13 (15.48)	2 (2.38)	**64** (**76.19**)	3 (3.57)	2 (2.38)	***lean back***	
10 (4.42)	37 (16.37)	0 (0)	**173** (**76.55**)	6 (2.65)	***right***	
10 (4.12)	6 (2.47)	2 (0.82)	2 (0.82)	**223** (**91.77**)	***left***	

6.3.3. Analysis of the classification impact of the feature selection

In this section we present the results of applying selection algorithms to the pool of 25 features used for the identification of patient movements. In Table 7 we show the ranking of the different features, classified in order of importance, according to the obtained Relief and Gain Ratio measurements.

The main analysis is: (a) the centres of mass of the small body parts, i.e. head, heels, and elbows, have a low feature ranking, (b) the pressure of the elbows also has a low position in the feature selection rankings, and (c) the X component of the centre of mass of the glutei is low ranked in the Gain Ratio test, while

Table 6
Performance results of the OneR, NB, C4.5, 1NN, AdaBoost, and MLP algorithms respectively (in %) for the 10 dataset partitions. Mean ($\bar{x}$) and standard deviation (σ) are shown in the last two rows of the table.

Data partitions	OneR	NB	C4.5	1NN	AdaBoost with C4.5	MLP
P1	55.41	47.3	63.51	51.35	72.97	**72.97**
P2	50	70.51	62.85	65.26	**75.64**	58.97
P3	57.14	85.71	92.86	77.14	88.57	**93.59**
P4	56.52	82.61	63.04	78.26	76.09	**93.59**
P5	63.39	78.57	77.68	73.21	83.93	**86.61**
P6	48.48	84.85	90.91	87.89	93.94	**94.64**
P7	40.18	65.18	72.32	59.82	**89.29**	87.5
P8	73.17	56.1	63.41	90.24	92.68	**95.54**
P9	62.71	55.93	79.66	69.49	84.92	**93.75**
P10	47.15	60.16	65.85	78.86	80.49	**87.80**
$\bar{x}$	55.42	68.69	73.21	73.15	83.8	**86.5**
σ	9.51	13.78	11.63	12.01	7.38	**11.76**

Table 7
Ranking output of the two feature selection algorithms used in this study. The dynamic features are ordered by the importance given by the algorithms. The horizontal line divides the final 13 selected features from the whole pool of 25 features.

Relief		Gain ratio	
0.053	$CM_{back}(x)$	0.400	TP_{glutei}
0.049	TP_{back}	0.384	TP_{head}
0.048	TP_{glutei}	0.339	$CM_{back}(y)$
0.046	$CM_{glutei}(y)$	0.336	TP_{heel2}
0.042	$CM_{back}(y)$	0.332	TP_{back}
0.042	r_{back}	0.324	r_{back}
0.039	s_{glutei}	0.316	TP_{heel1}
0.039	TP_{head}	0.297	r_{glutei}
0.035	s_{back}	0.293	$CM_{glutei}(y)$
0.033	$CM_{glutei}(x)$	0.291	s_{glutei}
0.033	TP_{heel1}	0.262	TP_{elbow1}
0.029	r_{glutei}	0.259	s_{back}
0.022	TP_{heel2}	0.220	$CM_{back}(x)$
0.019	$CM_{elbow1}(x)$	0.213	TP_{elbow2}
0.013	$CM_{heel2}(y)$	0.208	$CM_{head}(y)$
0.013	TP_{elbow2}	0.206	$CM_{glutei}(x)$
0.013	$CM_{heel1}(y)$	0.186	$CM_{elbow1}(x)$
0.012	$CM_{heel2}(x)$	0.182	$CM_{elbow1}(y)$
0.011	$CM_{heel1}(x)$	0.174	$CM_{elbow2}(x)$
0.008	$CM_{elbow1}(y)$	0.167	$CM_{heel1}(y)$
0.006	TP_{elbow1}	0.138	$CM_{elbow2}(y)$
0.004	$CM_{head}(y)$	0.136	$CM_{head}(x)$
0.003	$CM_{elbow2}(y)$	0.129	$CM_{heel1}(x)$
0.002	$CM_{head}(x)$	0.120	$CM_{heel2}(x)$
0.001	$CM_{elbow2}(x)$	0.103	$CM_{heel2}(y)$

Table 8
A list with the 13 selected features after applying the feature selection algorithms.

Head	Back	Glutei	Heels
TP_{head}	TP_{back}	TP_{glutei}	TP_{heel1}
			TP_{heel2}
	$CM_{back}(x)$	$CM_{glutei}(x)$	
	$CM_{back}(y)$	$CM_{glutei}(y)$	
	r_{back}	r_{glutei}	
	s_{back}	s_{glutei}	

Table 9
Comparison of the accuracy obtained by classifying the 10 independent test sets with the whole set of features (25) and the reduced set (13). The given values are the mean ($\bar{x}$) and standard deviation (σ) obtained from the runs on the 10 independent test data sets.

	25 features		13 features	
Classification models	$\bar{x}$ (%)	(σ)	$\bar{x}$ (%)	(σ)
OneR	55.41	9.51	55.41	9.51
NB	68.69	13.78	74.76	12.73
C4.5	73.21	11.63	78.97	11.28
1NN	73.15	12.08	79.63	11.17
AdaBoost with C4.5	83.85	7.38	84.74	9.89
MLP	**86.5**	**11.76**	**87.61**	**9.55**

rank is still high in the Relief measure. In contrast, the *Y* component remains high in both tests.

Consequently, and based on the experiments, we have created a new model composed of the most important 13 features selected from the Gain Ratio and Relief tests. The new model implies a 48% reduction with respect to the original 25 features. This new model is referred as MLP#9-13 because it maintains the same configuration than the best MLP configuration, MLP#9. See the selected features in Table 8.

We have investigated the performance of our MLP model trained with a reduced number of features (MLP#9–13) in comparison with the original model (MLP#9). The last row of Table 9 shows the accuracy, mean, and standard deviation of the classification results of MLP#9 (left) and MLP#9–13 (right) with the 10 independent partitions. In addition, we have redesigned the five comparison classifier models only with the selected features to be compared with the models that used the complete set of 25 features (see Table 9 and Fig. 9).

Looking at the results, we can observe how the MLP#9–13 model offers better performance than the original MLP#9 model. The improvement is in the order of 1.11%. This experimentation shows that the reduction made by the feature selection algorithms in the feature space do not result in a loss of accuracy; it even increases the overall performance. Also, the reduced features do not alter the ranking of the different classifiers methods. The MLP method still remains well ahead of the next best classifier. However, performance gains are more relevant in these classifiers, being the NB classifier the one getting higher improvement.

In Table 10, we can also see a time comparison between using the MLP with 25 features (MLP#9) and 13 features (MLP#9–13). Obviously, there is an efficiency increase in the computation time (20% faster with 13 input features). Thanks to the rapidness of the MLP, it can be observed that the time needed to classify one instance is not critical for the system in order to make a decision below 0.1 s.

The sum of the different confusion matrices of the 10 test folds is shown in Table 11. We can compare it with the confusion matrices obtained with the full feature set in Table 5. We notice that, in the MLP#9–13 model, the number of misclassified instances were reduced and that the diagonal of the confusion matrix shows better results.

7. Concluding remarks and future work

In this paper, a novel and complete classification system based on ANNs for recognising patient intentions on a bed was proposed. The classification system will be used to develop a natural human–machine interface to allow patients to control a motorised bed with slight body movements.

The recognition system is able to perfectly segment the pressure map of the patient. Then, a feature extraction process is performed to obtain the most representative features of the pressure maps. A MLP was successfully designed to transform these features in classified intentions of movement. Different MLP

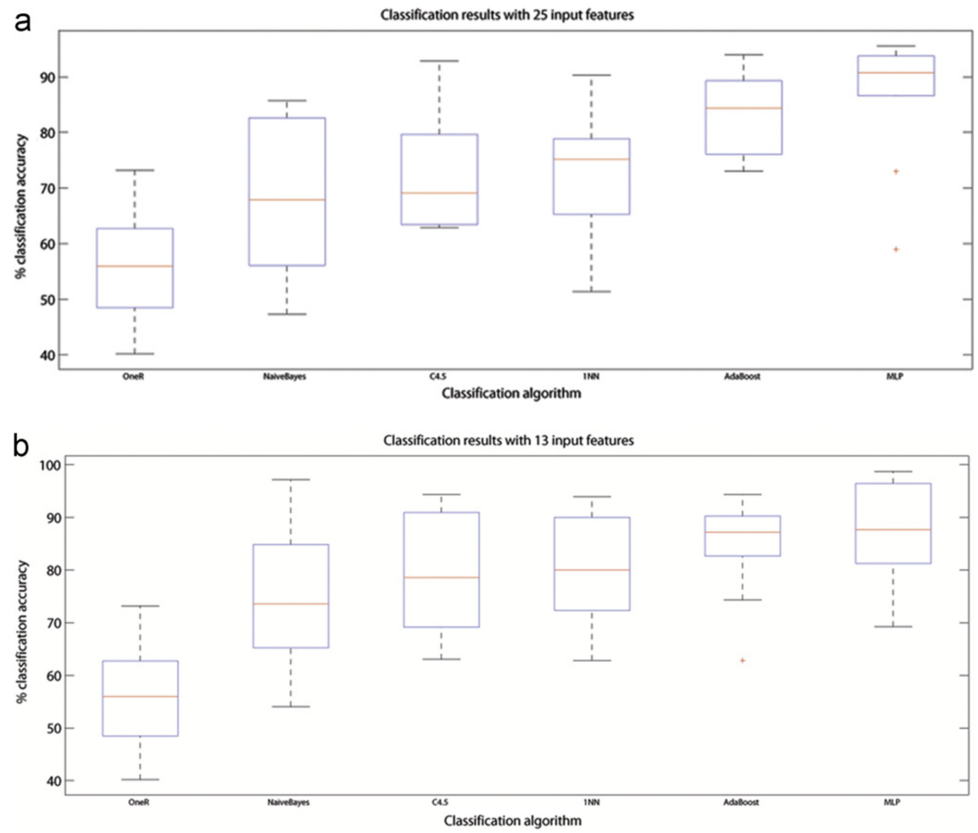

Fig. 9. Boxplots showing the performance of the five tested algorithms in the 10 data partitions. On the left, the models created using the initial 25 features. On the right, the classification models just with the reduced set of 13 features.

Table 10
Computation time of the MLP (with 25 and 13 features) in milliseconds. The time was calculated using the computer described in Section 6.2.

Algorithm name	Time for one instance	Time for the whole set of instances (**1154**)
MLP#9	0.02	24.16
MLP#9–13	0.02	20.36

Table 11
Confusion matrix of the MLP#9-13 classified instances using the reduced set of 13 features. The values are obtained from the sum of the classified instances of the 10 test datasets. Accuracy is in brackets in %.

Real classes						
rest	***sit up***	***lean back***	***right***	***left***		
297 (**96.74**)	4 (1.3)	3 (0.98)	0 (0)	3 (0.98)	***rest***	**Predicted**
8 (6.11)	**118** (**90.08**)	0 (0)	5 (3.82)	0 (0)	***sit up***	
8 (9.52)	7 (8.33)	**68** (**80.95**)	0 (0)	1 (1.19)	***lean back***	
2 (0.88)	38 (16.81)	0 (0)	**181** (**80.09**)	5 (2.21)	***right***	
18 (7.41)	7 (2.88)	10 (4.12)	1 (0.41)	**207** (**85.19**)	***left***	

configurations have been compared to each other, and the best configuration was a MLP with one hidden layer with 20 neurons, trained by the Bayesian regulation algorithm.

Our model was tested with real patients, and the results showed excellent performance on the test data, obtaining an accuracy higher than 87%, which may be good enough for the development of the future intelligent medical beds. We have compared the MLP classifier against five well-known classification paradigms. The MLP classifier always outperforms the other classifiers, being a 3% better than the well-known AdaBoost (ensemble of classifiers) and 13% better than C4.5 (decision trees).

We have observed how the MLP approach achieves good classification rates for the five intentions. The *rest* class presents the best results (more than the 95% of the test instances); followed by the *left* class (92%). Accuracy becomes reduced in the *sit up* and *right* classes. This is justified due to the fact that some patients started their *right* intention with a rising movement that could be incorrectly classified as a *sit up* intention.

In addition, feature selection algorithms to reduce the complexity and time cost of the model were applied. The results showed that the set of input features was reduced by 48%, and the models improved their performance. The computation time was also reduced by 20%. The feature selection algorithms considered that the centres of mass of the elbows, heels, and head are not very relevant to predict movement intentions since only small displacements are done in these areas, removing the corresponding features.

Future work will be devoted to online close the control loop with the motorised actuators of the bed, according to the MLP classification outputs. Furthermore, we expect that the system will improve its accuracy when running online since, after a short time, patients will train themselves on how to achieve a better performance. We also intend to perform further improvements in: (a) the classifier model, using evolutionary algorithms to identify a near optimal design of the configuration and architecture of the ANN (as done for example in [32]), and (b) in the sensors layout and distribution.

Conflict of interest statement

None.

Acknowledgements

The work presented in this paper has been carried out in the scope of the EPOSBED project. EPOSBED has been co-funded by the European Commission under the R4SME 7th Framework Program.

References

[1] G. Byrns, G. Reeder, G. Jin, K. Pachis, Risk factors for work-related low back pain in registered nurses, and potential obstacles in using mechanical lifting devices, J. Occup. Environ. Hyg. 1 (2004) 11–21.

[2] T.L. Douglas, Voice-actuated, speaker-dependent control system for hospital bed, J. Acoust. Soc. Am. 97 (1995) 2017.

[3] H.Z. Tan, L.A. Slivosky, A. Pent, A sensing chair using pressure distribution sensors, IEEE Trans. Mechatron. 6 (2001) 261–268.

[4] S. Mota, R.W. Picard, Automated posture analysis for detecting learner's interest level. in: Proceedings of the First Computer Vision and Pattern Recognition Workshop, vol. 5., IEEE Computer Society, Los Alamitos, CA, USA, 2003, pp. 49–55.

[5] K. Seo, C. Oh, J. Lee, Intelligent bed robot system: pose estimation using sensor distribution mattress, in: Proceedings of IEEE International Conference on Robotics and Biomimetics, Shenyang, China, pp. 828–832.

[6] B. Mutlu, A. Krause, J. Forlizzi, C. Guestrin, J.K. Hodgins, Robust, low-cost, non-intrusive sensing and recognition of seated postures, in: C. Shen, R.J.K. Jacob, R. Balakrishnan (Eds.), Proceedings of the 20th Annual Symposium on User Interface Software and Technology, ACM, Newport, RI, USA, 2007, pp. 149–158.

[7] M. Ferro, G. Pioggia, A. Tognetti, N. Carbonaro, D.D. Rossi, A sensing seat for human authentication, IEEE Trans. Inf. Foren. Sec. 4 (2009) 451–459.

[8] M.A. Denaï, F. Palis, A. Zeghbib, Modeling and control of non-linear systems using soft computing techniques, Appl. Soft Comput. 7 (2007) 728–738.

[9] M. Egmont-Petersen, D. de Ridder, H. Handels, Image processing with neural networks—a review, Patt. Recogn. 35 (2002) 2279–2301.

[10] K.A. Smith, J.N. Gupta, Neural networks in business: techniques and applications for the operations researcher, Comput. Oper. Res. 27 (2000) 1023–1044.

[11] S.R. Chowdhury, H. Saha, Development of a FPGA based fuzzy neural network system for early diagnosis of critical health condition of a patient, Comput. Biol. Med. 40 (2010) 190–200.

[12] G. Dombi, J. Rosbolt, R. Severson, Neural network analysis of employment history as a risk factor for prostate cancer, Comput. Biol. Med. 40 (2010) 751–757.

[13] S. Babaei, A. Geranmayeh, Heart sound reproduction based on neural network classification of cardiac valve disorders using wavelet transforms of pcg signals, Comput. Biol. Med. 39 (2009) 8–15.

[14] P. Delogu, M.E. Fantacci, P. Kasae, A. Retico, Characterization of mammographic masses using a gradient-based segmentation algorithm and a neural classifier, Comput. Biol. Med. 37 (2007) 1479–1491.

[15] G.R. Fowler, Diffuse septic peritonitis, with special reference to a new method of treatment, namely, the elevated head and trunk posture, to facilitate drainage into the pelvis, with a report of nine consecutive cases of recovery, Med. Class. 4 (1940) 551–580.

[16] T. Harada, T. Sato, T. Mori, Estimation of bed-ridden human's gross and slight movement based on pressure sensors distribution bed, in: Proceedings of the 2002 IEEE International Conference on Robotics and Automation, Washington DC, USA, pp. 3795–3799.

[17] M.H. Jones, R. Groubran, F. Knoefel, Reliable respiratory rate estimation from a bed pressure array, in: Proceedings of the IEEE Engineering in Medicine and Biology Society (EMBS'06), 28th Annual International Conference, New York, USA, pp. 6410–6413.

[18] T. Harada, A. Saito, T. Sato, T. Mori, Infant behavior recognition system based on pressure distribution image, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00), San Francisco, USA, pp. 4082–4088.

[19] C. Harris, M. J. Stephens, A combined corner and edge detector, in: Proceedings of the Fourth Alvey Vision Conference, Manchester, UK, pp. 147–151.

[20] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[21] H. Liu, H. Motoda, Feature Selection for Knowledge Discovery and Data Mining, Kluwer Academic, Boston, USA, 1998.

[22] K. Kira, L.A. Rendell, A practical approach to feature selection, in: Proceedings of the Ninth International Workshop on Machine Learning, Morgan Kaufmann, San Francisco, CA, USA, 1992, pp. 249–256.

[23] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Networks 5 (1994) 537–550.

[24] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, USA, 2006.

[25] W. Huang, R. Lippmann, Neural net and traditional classifiers, in: D. Anderson (Ed.), Neural Information Processing Systems, American Institute of Physics, New York, 1988, pp. 387–396.

[26] L. Rendell, R. Seshu, Learning hard concepts through constructive induction: framework and rationale, Comput. Intell. 6 (1990) 247–270.

[27] R.C. Holte, Very simple classification rules perform well on most commonly used datasets, Mach. Learn. 11 (1993) 63–91.

[28] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Mateo, CA, USA, 1993.

[29] E. Frank, I.H. Witten, Generating accurate rule sets without global optimization, in: Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, USA, 1998, pp. 144–151.

[30] G.H. John, P. Langley, Estimating continuous distributions in Bayesian classifiers, in: Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, Montreal, Quebec, 1995, pp. 338–345.

[31] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: Proceedings of the 13th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA, USA, 1996, pp. 148–156.

[32] H. Abbass, An evolutionary artificial neural networks approach for breast cancer diagnosis, Artif. Intell. Med. 25 (2002) 265–281.