

# Human Pose Estimation from Pressure Sensor Data

Leslie Casas<sup>1,2</sup>, Chris Mürwald<sup>2</sup>, Felix Achilles<sup>1,2</sup>, Diana Mateus<sup>1</sup>, Dietrich Huber<sup>2</sup>, Nassir Navab<sup>1,2</sup>, Stefanie Demirci<sup>1,2</sup>

<sup>1</sup>Computer Aided Medical Procedures, Technische Universität München, Germany

<sup>2</sup>Sanvisio, Austria

<sup>3</sup>Computer Aided Medical Procedures, Johns Hopkins University, USA

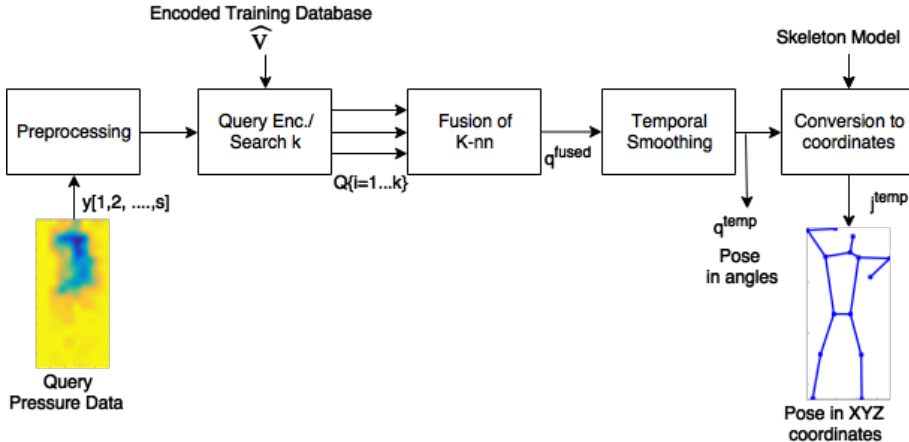
leslie.casas@tum.de

**Abstract.** In-bed motion monitoring has become of great interest for a variety of clinical applications. In this paper, we introduce a hash-based learning method to retrieve human poses from pressure sensors data in real time considering temporal correlation between poses. The basis of our approach is a multimodal database describing different in-bed activities. Database entries have been created using an array of pressure sensors and an additional motion capture system. Our results show good performance even in poses where the subject has minimal contact with the sensors.

## 1 Introduction

In today’s highly digitized health care workflows, in-bed patient motion monitoring has become a crucial requirement for various aspects such as patient positioning for precise treatment, disease and disorder diagnosis [1], detection of bed-exit and fall events [2], and bedsore prevention [3].

Automatic monitoring systems have generally used cameras that monitor patients in two dimensions, typically reducing a scene to a flat image. There has been only a few attempts to monitor patient movements on a bed mattress. An early idea presented by M. P. Toms [4], describes the use of fluid filled cells between the patient and a support in order to detect motion via pressure fluctuations. Alaziz et al. [5] suggest to use low-end load cells placed under each bed leg, and classify 27 pre-defined movements by analysing the computed forces. A similar approach has been validated by Hoque et al. [2] replacing load cells with active RFID sensors equipped with accelerometers. Both approaches are able to show promising results, but lack high-speed algorithmic solutions in order to enable a real-time processing. Employing a powerful deep learning approach, Heydarzadeh et al. [3] yield close-to-real-time in-bed patient posture classification based on pressure data collected from a commercially available force sensing array mat that is placed between bed mattress and linen. Despite short processing time of the classification step, the preprocessing of pressure data involves tedious filtering which slows down the entire speed.

**Fig. 1.** Human pose estimation from a pressure distribution image

In this paper we introduce a hash-based learning method to retrieve human posture poses in real time. Similar to the aforementioned approach, we retrieve pressure data from sensors that are incorporated into the patient bed mattress. Our approach is inspired by Harada et al. [6] who have proposed to solve the general computer vision pose estimation problem via a synthetic pressure distribution image generated by a surface human mesh model. This way, the pose can easily be computed by minimizing a cost function considering pressure distribution, gravity and momentum.

## 2 Materiales and Methods

In this section, we explain the pipeline of the proposed work as depicted in Fig. 1. After presenting the acquisition and processing of the database, our encoding model learning process and the real-time estimation of the human pose are introduced.

### 2.1 Multimodal Database

In this work, we acquired a multimodal dataset  $\mathbf{D}$  using two systems. The first is an array of  $s$  pressure sensors arranged horizontally and equidistantly covering an area of 2m x 1m. In addition, we have employed a commercial 5-calibrated camera motion capture system, which tracks the position of a 14-joint skeleton (head, neck, shoulders, elbows, wrists, hips, knees and ankles). The synchronization of both acquisitions has been performed offline using the time stamps of each recording and eventually setting the whole system acquisition frequency to 30Hz.

The acquisition was performed on 6 subjects, 2 female and 4 male, performing 5 distinct activities (see Table 1) for 1 minute each, according to an acquisition

Activity	Name	Description
A1	Random horizontal positions	Subject simulates sleeping positions.
A2	Fixed horizontal positions	Subject changes horizontally from left, center and right position.
A3	Rotation of joints	Keeping the trunk horizontal, subject rotates upper and lower limb joints, and head.
A4	Body stretching	Subject performs stretching positions not keeping the horizontal position.
A5	Simulated Seizures	Subject performs rapid twitching movements simulating the clonic phase of an epileptic seizure

**Table 1.** Description of the activities performed in the acquisition process.

protocol inspired by work of Achilles et al. [1]. Each subject performed the activity recording lying on a mattress, which was located on top of the array of pressure sensors, and wearing motion capture sensors.

The pressure-based modality (pressure sensor array) is represented as a pressure distribution image normalized to  $[0,1]$  range per sensor. Hence, each sensor represents a stand-alone-feature in the feature space. The pose-base modality (motion capture system) is presented as joint angles describing the 14-joint skeleton in 13 segments, each segment being represented by 3 directional angles (one per axis). Therefore, each multimodal database element, which represents the acquired data in frame  $i$ ,  $\mathbf{D}_i = \{\mathbf{p}_i, \mathbf{a}_i\}$  consists of a tuple of the normalized pressure distribution images arranged as a vector  $\mathbf{p}_i = \{p_i^1, \dots, p_i^s\}$  and a vector of directional angles  $\mathbf{a}_i = \{a_i^1, \dots, a_i^{13 \times 3}\}$ .

## 2.2 Hash Model Learning

Inspired by *Iterative Quantization* [7], we employ a similar strategy to encode the pressure distribution training data.

Let  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \in \mathbb{R}^{n \times d}$  be a set of  $n$  d-dimensional, zero-centered data points. Our technique learns a binary code matrix  $\mathbf{B} = \text{sgn}(\mathbf{X}\mathbf{W}) \in \{-1, 1\}^{n \times c}$  where  $c$  is the code length and  $\mathbf{W} \in \mathbb{R}^{d \times c}$  is the matrix of hyperplane coefficients.

First, an orthogonal transformation is applied to the data to reduce the dimension and obtain an initial encoding function  $I$  which maximizes the variance of each bit and provides de-correlated pairwise code bits:

$$I(\mathbf{W}) = \sum_k \text{var}(h_k(\mathbf{x})) \quad (1)$$

where  $h_k(x) = \text{sgn}(x\mathbf{w}_k)$  and  $\mathbf{w}_k$  are the columns of matrix  $\mathbf{W}$  ( $\mathbf{W}^t\mathbf{W} = 1$ ). When the orthogonal transformation is Principal Component Analysis (PCA),  $\mathbf{W}$  is obtained by taking the top  $c$  eigenvectors for the covariance matrix  $\mathbf{X}^t\mathbf{X}$ . Let  $\mathbf{V} = \mathbf{X}\mathbf{W}$  be the matrix of projected data. There exists a orthogonal transformation matrix  $\mathbf{R} \in \mathbb{R}^{c \times c}$  applied to  $\mathbf{V}$  such that it minimizes the quantization loss function

$$Q(\mathbf{B}, \mathbf{R}) = \|\mathbf{B} - \mathbf{V}\mathbf{R}\|_F^2, \quad (2)$$

which measures the distance between the projected data  $\mathbf{V}$  and the vertices of the hypercube  $\{-1, 1\}^c$ . The final encoding of  $\mathbf{X}$  is  $\hat{\mathbf{V}} = \mathbf{XWR}$ .

### 2.3 Real-time Pose Estimation

At this stage, we employ the previously created database to realize our real-time pose estimation algorithm as online query. The huge advantage of this implementation is its speed. In the remainder of this section, we describe our algorithm according to the pipeline displayed in Fig.1.

**Step 1: Preprocessing.** As first step, incoming query pressure data  $\mathbf{y} = \{y_1, y_2, \dots, y_n\}$  is normalized to  $[0, 1]$ .

**Step 2: Query Encoding/Search.** In this step, we apply model  $\mathbf{M}=\mathbf{WR}$  learned using the initial hashing algorithm (see section 2.2) to the input pressure data vector  $\mathbf{y}$  yielding an encoded representation  $\hat{\mathbf{y}} = \mathbf{yM}$ . Then, the search consists in finding the  $k$  closest embeddings with indexes  $\{i_1, \dots, i_k\}$  within the pressure distribution training dataset according to Hamming distance. Finally, an array of  $k$  poses  $\mathbf{Q} = \{\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_k}\}$  corresponding to the  $k$  closest encoding found in the search is retrieved.

**Step 3: Fusion of  $k$  poses.** The aim of this step is to eliminate corrupted or outlier poses within set  $Q$  in order to allow for a robust estimation. To this end, we employ median absolute deviation (MAD) to eliminate outlier poses in  $Q$  and yield an average pose set  $\mathbf{q}^{fused}$ .

**Step 4: Temporal Smoothing.** In order to ensure correlation between poses in subsequent frames, a computing window of  $2m + 1$  frames  $[t - m, t + m]$  is used to smooth the transition and avoid corrupted static poses. We also employ MAD as in Step 3 to obtain the final pose  $\mathbf{q}^{temp}$ .

**Step 5: Coordinate Conversion.** For visualization purposes, we compute a synthetic pose in joint 3D positions  $\mathbf{j}^{temp}$  from the resultant pose  $\mathbf{q}^{temp}$  using a template skeleton.

## 3 Results

We have performed cross validation on our database in order to evaluate our presented approach on the subjects as well as activities. Table 2 shows the results of leave-one-subject-out cross validation. The median absolute error (MAE) in column 2 and 3, takes as reference a template skeleton to all the subjects (see Step 5 in section 2.3). Table 3 shows the leave-one-subject-out cross validation results.

## 4 Discussion

In this paper, we have presented a simple yet effective method to yield real-time patient posture estimation from pressure sensor data. Each pose estimation took 16ms in average using Matlab 2016b on a Core i7 with 2.60GHz and 12GB RAM.

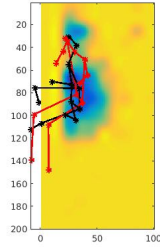
Subject	MAE(deg)	MAE(cm)
S1	24.223	13.379
S2	25.375	13.868
S3	22.340	10.582
S4	17.545	9.781
S5	20.283	9.762
S6	23.177	14.024

**Table 2.** Pose estimation mean absolute errors(MAE) using leave-one-subject-out cross validation

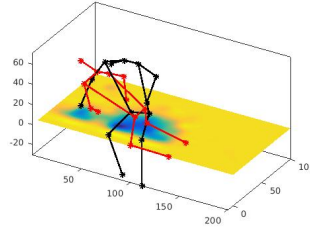
Activity	MAE(deg)	MAE(cm)
A1	29.266	14.029
A2	27.954	13.871
A3	14.526	9.829
A4	33.556	21.958
A5	19.750	10.928

**Table 3.** Pose estimation mean absolute errors(MAE) using leave-one-activity-out cross validation

**Fig. 2.** Bad Pose Estimation: ground truth pose (black) vs. estimated pose (red)



(a) View 1



(b) View 2

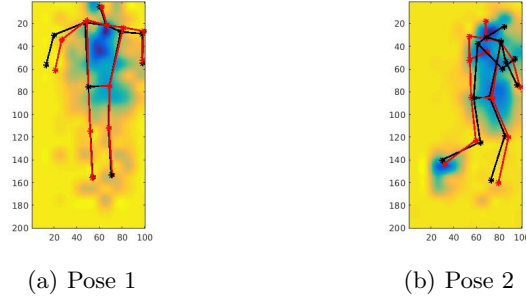
Our experiments have revealed only a slight position error that has no severe influence on the overall posture detection. As expected, best results have been obtained when the subject is positioned flat (see Fig. 3). On the other hand, poses where the subject has less contact with the surface (such as activity 4) yield the highest error (see Table 3). However, even in those cases, the estimated pose correlates with the real patient posture (see Fig. 2).

Our results further indicate that a potential inclusion of more robust learning techniques such as convolutional/recurrent neural networks (CNN/RNN) for temporal constraint, may improve the accuracy and robustness of our human pose estimation approach. In order to move forward our approach to be applied in clinical environment, we will follow this path for future work.

## References

1. Achilles F, Ichim AE, Coskun H, Tombari F, Noachtar S, Navab N. Patient MoCap: Human Pose Estimation Under Blanket Occlusion for Hospital Monitoring Applications. In: MICCAI (1). vol. 9900 of Lecture Notes in Computer Science; 2016. p. 491–499.
2. Hoque E, Dickerson RF, Stankovic JA. Monitoring Body Positions and Movements During Sleep Using WISPs. In: Wireless Health 2010. WH '10. New York, NY, USA: ACM; 2010. p. 44–53.

**Fig. 3.** Good Pose Estimation: ground truth pose (black) vs. estimated pose (red)



3. Heydarzadeh M, Nourani M, Ostadabbas S. In-bed posture classification using deep autoencoders. In: Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the. IEEE; 2016. p. 3839–3842.
4. Toms MP. Patient movement detection. Google Patents; 2000. US Patent 6,036,660. Available from: <https://www.google.com/patents/US6036660>.
5. Alaziz M, Jia Z, Liu J, Howard R, Chen Y, Zhang Y. Motion Scale: A Body Motion Monitoring System Using Bed-Mounted Wireless Load Cells. In: 2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE); 2016. p. 183–192.
6. Harada T, Sato T, Mori T. Pressure distribution image based human motion tracking system using skeleton and surface integration model. In: Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164). vol. 4; 2001. p. 3201–3207 vol.4.
7. Gong Y, Lazebnik S, Gordo A, Perronnin F. Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2013 Dec;35(12):2916–2929.