

Patient 3D Body Pose Estimation from Pressure Imaging

Leslie Casas · Nassir Navab · Stefanie
Demirci

Received: date / Accepted: date

Abstract *Purpose* In-bed motion monitoring has become of great interest for a variety of clinical applications. Image-based approaches could be seen as a natural non-intrusive approach for this purpose, however video devices require special challenging settings for a clinical environment. We propose to estimate the patient's posture from pressure sensors' data mapped to images.

Methods We introduce a deep learning method to retrieve human poses from pressure sensors data. In addition, we present a second approach that is based on a hashing content-retrieval approach.

Results Our results show good performance with both presented methods even in poses where the subject has minimal contact with the sensors. Moreover, we show that deep learning approaches could be used in this medical application despite of the limited amount of available training data. Our ConvNet approach provides an overall posture even when the patient has less contact with the mattress surface. In addition, we show that both methods could be used in real-time patient monitoring.

Conclusions We have provided two methods to successfully perform real-time in-bed patient pose estimation, which is robust to different size of patient and activities. Furthermore, it can provide an overall posture even when the patient has less contact with the mattress surface.

Keywords pressure sensors · patient monitoring · human pose estimation · hashing · ConvNets · deep learning

L. Casas (✉) · N. Navab · S. Demirci
Computer Aided Medical Procedures, Technische Universität München, Boltzmannstr 3,
85748 Garching, Germany
E-mail: leslie.casas@tum.de

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 [22], disease and disorder diagnosis [1], detection of bed-exit and fall events [14], and bedsore prevention [13]. The main idea is to rapidly and accurately detect changes in patient position and pose while lying in bed or on operating tables. This can be compared to a similar challenge in computer vision that aims at estimating human body poses from images in various scenarios [5, 9, 19, 20, 26]. The main difference, however, is that humans in computer vision scenarios are generally captured in an up-right position, whereas patients in clinical scenarios are mostly lying or sitting. Nevertheless, many approaches related to computer vision challenges, have been used as a non-intrusive way towards patient motion monitoring. However, sensors like depth-sensors or video cameras require calibration or specific lighting conditions and thus cannot be easily installed in a clinical environment. To overcome these limitations, the trend goes towards using pressure-sensing mattresses that produce image imprints of the human body. Their main advantage is that they do not require the installation of additional equipment inside the clinics. In the remainder of this section, we give a detailed overview of all related approaches and introduce our main contribution in regard to this state-of-the-art.

2 Related work

Automatic monitoring systems have generally relied on cameras that monitor patients in two dimensions, typically reducing a scene to a flat image. In [3], video images are employed to detect the 3D body pose as well [7, 15]. Depth images have been used in [4] to detect the area of bed occupied for fall detection using dense SIFT features. A combination of depth and video images have been also employed to monitor patients by estimating human pose through a ConvNet approach [1].

There has been only a few attempts to monitor patient movements on a bed pressure mattress. An early idea presented by M. P. Toms [24], describes the use of fluid filled cells between the patient and a support in order to detect motion via pressure fluctuations. Alaziz et al. [2] 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. [14] 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 real-time processing.

A solution to this problem are machine learning approaches that have also been introduced for bed monitoring. Kortelainen et al [17] use pressure and video images for sleeping posture classification using a multiclass SVM. Employing a powerful deep learning approach, Heydarzadeh et al. [13] 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.

On the contrary, we aim at estimating the in-bed patient 3D pose in a wider application scope. Harada et al. [12] propose to solve the general computer vision pose estimation problem via a synthetic pressure distribution image generated by a surface human mesh model.

Grimm et al.[11] estimate the body pose and orientation from synthetic depth and pressure images by first detecting the body orientation heuristically and later retrieving the pose by nearest neighbors.

ConvNet methods. Human pose estimation has been significantly advanced with the use of deep neural networks. Relative simple deep models have resulted in successfully regressing the body pose as set of values [6]. More recently, heatmap regression [5, 19, 26] and more complex models [9, 20] have improved the localization accuracy.

Hashing-based methods. Hashing has been used in the computer vision community for image retrieval in large scale databases where the task becomes complex due to the high-dimensionality of the data [18]. This technique aims at encoding the data into a different representation in order to enable a fast query. In clinical applications, several works have employed hashing to large medical database search [27] or in histopathological image analysis [28].

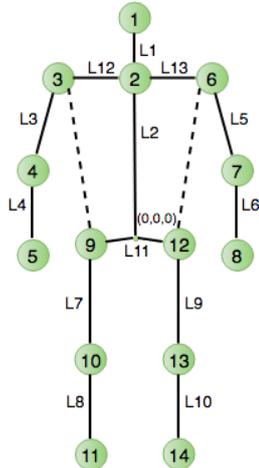
For pose estimation, hashing has been widely used for 3D object pose estimation [16, 21]. However for human pose estimation, the literature is limited. [23] employs locality-sensitive-hashing to retrieve articulated poses from a large database of exemplars.

2.1 Contribution

In this paper, we present two simple yet effective methods to retrieve 3D human poses from pressure mattress images. Compared to earlier approaches, we create our pressure image from only a very small subset of pressure sensors normally used within such setups, and show that efficient 3D pose retrieval is still possible.

The first is a hashing-based method which is a content-retrieval approach [8]. It searches a query pressure image in a training hash table and retrieves the nearest neighbors 3D body poses, which are fused to obtain a final 3D pose. The second method is a Convolutional Neural Network (ConvNet) approach which learns to estimate a 3D human pose from a pressure image by regression. In this work, we rely on a simple regressor from [6]. The reason is that we have access to limited amount of training data in our problem and thus it is not possible to adopt a heavy deep model.

Fig. 1 Graphical model of the human body. The green circles represent the $J=14$ joints and the black lines (S1-S13) are the body segments used to estimate orientation.



In our experiments we show that both methods perform well with some potential towards activity generalization for the ConvNet-based approach. We show that despite of the low resolution images represented by an array of pressure sensors, it is possible to estimate in-bed patient poses for monitoring purposes in real-time. Moreover, we show that even though large databases of pressure images are not available, the ConvNet-based method is able to learn very efficiently.

3 Materials and Methods

In this section, we explain the methods we use for 3D in-bed patient pose estimation. Let $\mathbf{x} \in \mathbb{R}^{R \times C}$ be a pressure image obtained from an $R \times C$ array of pressure sensors uniformly distributed over the entire length and width of a standard patient bed mattress. In our problem, $R = 20$ and $C = 10$ denote the number of sensors per row and column respectively. Using information encoded in \mathbf{x} , we aim at estimating the correct 3D body pose $\mathbf{y} \in \mathbb{R}^{3 \times J}$ of a single individual with $J = 14$ body joints. Every joint and orientation is represented by a three-dimensional vector.

Given a multi-modal training dataset $\mathcal{D} = \{(\mathbf{x}, \mathbf{y})\}_{i=1}^N$ comprising of N pressure images \mathbf{x}_i and corresponding 3D joint positions \mathbf{y}_i , we present two methods to estimate 3D human pose. At first, we use hashing to learn functions to map our pressure database into a binary representation to perform a content-based pose retrieval. In an extended approach, we incorporate ConvNets as regressor to learn a mapping between a pressure image and a 3D pose. In this section, we explain both methods we use for pose estimation.

3.1 Hashing-based pose estimation

We create a look-up table containing the N pressure images $\mathbf{H} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\} \in \mathbb{R}^{N \times P}$. Here, each pressure image \mathbf{x}_i is represented as a zero-centered vector of length $P = R \times C$.

In order to dissolve ambiguities introduced by different person sizes in the database, we standardize all 3D poses \mathbf{y}_i by matching a template skeleton with J body joints and $J - 1$ segments or limbs to them [4]. Therefore, we define the pelvis as center of coordinates and compute new standardized joint positions \mathbf{y}_i^s and limb orientations \mathbf{y}_i^a for each database entry.

The proposed framework of our image based pose-retrieval method is displayed in Fig. 2. In theory, the structure enables us to directly measure the similarity between a query pressure image and the database to find the nearest neighbors. However this is computationally expensive given the size of the database. Therefore, we employ a hashing method to learn a set of hashing functions and map the pressure images into binary codes of $L = 128$ bits and create a hash table (offline hashing table construction). This mapping allows for a more efficient search, especially for real-time pose estimation. Given a query, we perform a K-nearest neighbors (K-nn) search and average the retrieved poses avoiding outlier poses and adding temporal correlation (online query search).

Offline hashing table construction

We use the unsupervised hashing method *Iterative Quantization* (ITQ) [10] to learn a similarity-preserving binary code representation of our look-up table \mathbf{H} . Given the predefined code length L , the hash look-up table is $\hat{\mathbf{B}} \in \mathbb{R}^{N \times L}$.

Our first aim is to yield a reduction of data dimensionality via the concept of Principal Component Analysis. Here, a suitable matrix \mathbf{W} is obtained by taking the top L eigenvectors of the covariance matrix $\mathbf{H}^T \mathbf{H}$. Given the projected data matrix $\mathbf{V} = \mathbf{H} \mathbf{W}$, there exists a rotation matrix $\mathbf{R} \in \mathbb{R}^{L \times L}$ that is applied to \mathbf{V} such that it minimizes the quantization loss function:

$$Q(\mathbf{B}, \mathbf{R}) = \|\mathbf{B} - \mathbf{V}\mathbf{R}\|_F^2, \quad (1)$$

where $\mathbf{B} \in \{-1, 1\}^{N \times L}$ represents the vertex of hypercube $\{-1, 1\}^L$. The reduced hash look-up table is then defined by $\hat{\mathbf{B}} = \mathbf{H}\mathbf{W}\mathbf{R}$.

Online query

Given an input query pressure image \mathbf{x}_q , a hash code \mathbf{b}_q is obtained by applying \mathbf{W} and \mathbf{R} . Later, a K-nn search is performed in the reduced hash look-up

Fig. 2 Hashing-based pipeline for patient pose estimation from pressure images. There are two stages: an offline hashing table creation and an online query which takes as input a pressure image and the output is the estimated pose.

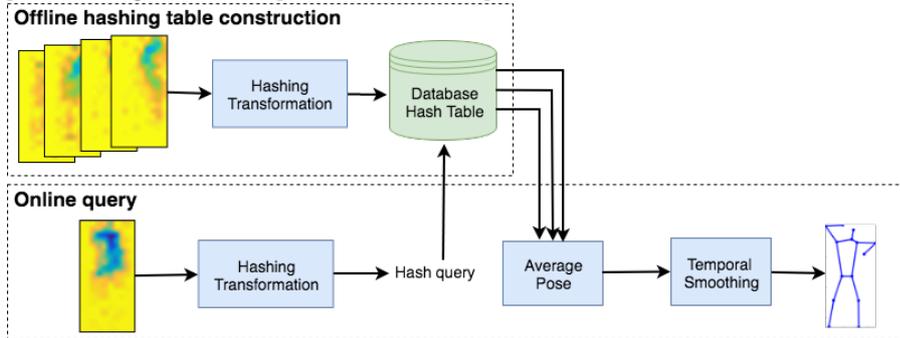


table $\hat{\mathbf{B}}$ using Hamming distance. The 3D poses based on limb orientations $\mathbf{y}_{1..k}^a$ corresponding to the K closest pressure images are retrieved.

In order to avoid corrupted or outlier poses, we discard all poses with

$$\text{abs}(\mathbf{y}_i^a - \text{median}(\mathbf{y}_{1..k}^a)) / \text{MAD} > 2.5.$$

Then, the average of the remaining retrieved limb orientation poses is computed. In addition, to provide temporal smoothness between subsequent frames, we extract a window of $2m + 1$ frames $[t - m, t + m]$ to smooth the transition and avoid corrupted static poses by employing the same strategy based on MAD. The final pose estimated is given by $\hat{\mathbf{y}}_i^s$. Using the skeleton template we compute $\hat{\mathbf{y}}_i^s$.

3.2 ConvNets-based pose estimation

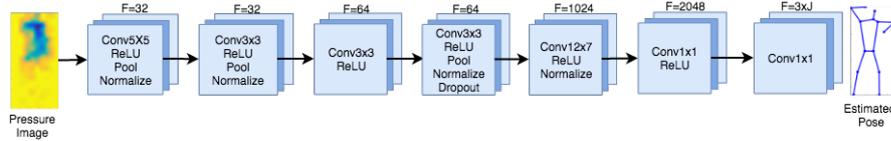
As an extension to our hashing based method, we propose to use ConvNets as regressor to estimate a human pose from a pressure image.

Let ϕ be a function representing a ConvNet used as a regressor. Given the training dataset \mathcal{D} we aim at learning the parameters θ of ϕ , in order to map an input pressure image \mathbf{x} to a set of $J = 14$ 3D joint positions \mathbf{y} . Therefore, the estimated output vector is given by $\hat{\mathbf{y}} = \phi(\mathbf{x}; \theta)$. The objective function shown in Eq.2 represents the L2 loss between the predicted values and the ground truth.

$$\mathcal{L}(\mathcal{D}, \theta) = \frac{1}{2J} \sum_{i=1}^J \|\phi_{\theta}(\mathbf{x}_i) - \mathbf{y}_i\|_2^2. \quad (2)$$

ConvNets employed as regressors, strongly depend on the detection of features. Models such as AlexNet, VGGNet-16 or ResNet, have shown strong results for classification tasks on natural images due to an efficient feature

Fig. 3 ConvNet-based method for patient pose estimation from pressure images. The input of the network is a pressure image and the output is the estimated pose. The ConvNet is represented by a cascade of 7 layers.



extraction. However, these models are deep and require large databases just like ImageNet (> 1 mio samples), in order to be able to learn the number of parameters (> 4 mio). In this work, however, we deal with small images (200 data points per image) and a limited amount of training data ($\sim 60K$ samples). This is why, we have opted for

the network model presented in [6]. The model consists on a cascade of ConvNets trained to regress a set of 2D joint positions from RGB images. The input images are already cropped according to the individual. We use the L2 loss for training. The results reported show robustness for convergence and good accuracy performing better than state of the art methods.

The network has 5 convolutional layers followed by 2 fully connected layers. The fully connected layers are presented as 1×1 convolutional layer because the operation is performed in the filters dimension (refer to Fig. 3). Following a coarse-to-fine model, the first convolution layer has a 5×5 kernel and the rest 3×3 kernels. Dropout is employed in the fourth convolutional layer to prevent overfitting.

In order to completely match our intended aim, we have slightly modified the network architecture given in [6] to regress $3 \times J$ values from one layer images. Therefore, the last layer number of filters is modified to $3 \times J$ (see Fig. 3).

4 Experiments

We have evaluated the proposed method with the multimodal database reported from [8]. This database contains recordings of 6 subjects, 2 female and 4 male, performing 5 distinct activities on a bed for 1 – 1.5 minutes each (see Table 1). The modalities contained are pressure distribution images and 3D ground-truth joint positions. According to [8], the first modality is obtained from an array of pressure sensors placed under a foam mattress. For the second modality, the patient uses a set of markers to obtain the ground-truth position of each joint using a motion capture system.

We have performed evaluations of the two presented methods, the hashing-based and the ConvNet-based method. As evaluation metric, we use mean absolute error (MAE) between the ground-truth pose \mathbf{y} and the estimated pose $\hat{\mathbf{y}}$ where the 3D joint position error is in centimeters and limb orientation

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 necessarily 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 multimodal dataset reported in [8].

in degrees. In both cases, we compute the error along the $S=J$ joints or $S=J-1$ limbs and the 3 coordinates (X, Y and Z) as shown in Equation 3.

$$MAE = \frac{1}{3S} \sum_{i=1}^S \sum_{c=1}^3 |y_{i_c} - \hat{y}_{i_c}|. \quad (3)$$

We compare our two methods with cross-validation experiments per subject and per activity. Moreover, we have performed a comparison based on orientation error per skeleton limb. In order to yield comparable results between both models, we have normalized the ConvNet output employing the same template skeleton strategy used in hashing (See Section 3.1).

In all the ConvNet experiments, the network parameters are randomly initialized. In addition, we have evaluated the use of a pre-trained network in the ConvNet-based method and the feasibility of real-time application by measuring the execution time of a query in both methods.

All evaluations have been conducted in Matlab 2017b on a 3.7GHz CPU with 8 cores and 64GB RAM. For the ConvNet method, we took as baseline the public implementation provided in [6] with MatConvNet 1.0-beta25 [25]. A Matlab implementation of the ITQ-hashing method has been provided by the original authors [10].

4.1 Cross-validation experiments

We perform a leave-one-out cross-validation for each subject and each activity. The subject cross-validation results are shown in Table 2. It is observed that the ConvNet method performs better than the hashing method for all subjects. In average, the improvement is 3.5cm in terms of joints position while the orientation is also improved in 3 degrees in average. This shows that the deep neural network dominates even this limited amount of training data. Moreover, it is invariant to the size or weight of a subject.

Subject	Hashing		ConvNet	
	Orientation(deg)	Position(cm)	Orientation(deg)	Position(cm)
S1	24.22	13.38	21.28	9.53
S2	25.37	13.87	22.74	8.71
S3	22.34	10.58	16.86	7.01
S4	17.54	9.78	15.25	6.01
S5	20.28	9.76	18.54	7.49
S6	23.17	14.02	21.05	9.24

Table 2 Joint position and limb orientation mean absolute errors (MAE) using leave-one-subject-out cross validation for hashing-based and ConvNet-based method.

Subject	Hashing		ConvNet	
	Orientation(deg)	Position(cm)	Orientation(deg)	Position(cm)
A1	29.26	14.02	17.83	7.02
A2	27.95	13.87	19.85	8.48
A3	14.52	9.82	9.29	3.79
A4	33.55	21.95	21.19	10.68
A5	19.75	10.92	14.42	1.44

Table 3 Joint position and limb orientation mean absolute errors (MAE) using leave-one-activity-out cross validation for hashing-based and ConvNet-based method.

In the activity cross-validation evaluation that is shown in Table 3, the ConvNet method performs better as well. It can be observed that in activity A4, the error in the hashing method raises to 21.95cm in comparison to 10.68cm for the ConvNet method. This performance is expected given that the hashing method is a content-retrieval search method and A4 is the only activity which includes not horizontal positions. On the other hand, the low errors between 3.5 and 8cm reported by ConvNet gives us the intuition that it can generalize better to different activities, even when the person is not in direct contact with the mattress.

4.2 Evaluation per limb

We have evaluated the performance of both methods with respect to the limbs. Table 4 shows that the ConvNet method performs better in terms of 3D position and orientation. It can be observed that in most of the limbs the error reduction obtained by the ConvNet method is considerable (~ 5 cm) except in the legs. This correlates to the fact that the range of movement of the legs during the activities recorded in the database is wider than the arms. In addition, the lowest errors reported are localized in the torso due to the low range of motion of this limb in a in-bed natural posture. Regarding the head error, it is a good example to show the importance of reporting errors not only in position but also in orientation. In this case, even though the position error seems low, the orientation error is the highest on the table. This high error is as well expected given that the head has a expanded range of mobility and is up to some extent independent form the rest of the body posture.

Method	Evaluation Mode	Head	Torso	Upper Legs	Lower Legs	Upper Arms	Lower Arms	Full Body
Hashing	position(cm)	13.47	10.23	15.10	17.98	10.90	12.09	12.20
	orientation(deg)	29.11	8.74	22.41	26.15	9.65	10.88	22.25
ConvNet	position(cm)	8.78	4.94	10.70	15.35	5.94	7.27	8.00
	orientation(deg)	28.25	9.82	23.46	26.98	11.24	12.00	19.29

Table 4 Joint position and limb orientation mean absolute errors(MAE) per limb for hashing and ConvNet method. The skeleton template is divided in the seven limb groups shown in the first row of the table. The last column reports the average of whole skeleton.

Subject	wo. Pretrained	w. Pretrained
S1	6.31	6.54
S2	6.84	7.58
S3	6.36	7.42
S4	5.97	5.79
S5	6.13	5.79
S6	9.20	9.26
Average	6.80	7.06

Table 5 Joint position mean absolute error(MAE) in cm for the ConvNet-based method trained from scratch(wo. Pretrained) and trained using a pretrained model on LSP dataset(w. Pretrained)

4.3 Evaluation based on a Pretrained Network

Deep learning approaches often rely on pre-trained models when the amount of training data is limited. We adopt this idea for our problem as well. We use a pre-trained model from [6] to initialize our model. However, please note that the pre-trained model has been build from RGB image data . When using the pre-trained model, we discard the first and last layer weights, which are randomly initialized. Later when using our own database, we perform fine-tuning. We use the same number of epochs(150) for both way of training. Table 5 shows the results of our ConvNet trained from scratch and using the pre-trained model.

It can be observed that the model using the pre-trained network performs worse than the one trained from scratch. This is somehow expected given that the network has been trained on a different modality. Furthermore, the first layers of a deep network, in general, capture universal textural features as curves and edges. Therefore, when we use a model trained on RBG images which have rich context and textural features, it fails to find similar features in a less rich domain as the pressure image. Moreover, using the 2D joint regressor network to estimate a 3D values would tend to lose information regarding the additional third dimension.

4.4 Execution Time

Finally, we have evaluated the execution time of each model to examine whether real-time execution is possible. This is an important requirement for patient monitoring. We measure the execution time for both methods given a new input pressure image. The execution time per query takes 5.9 ± 0.41 ms for the hashing. The inference time for the ConvNet is 3.41ms. Both approaches are considered as real-time for our application.

5 Discussion and conclusions

In this paper, we have presented two simple yet effective methods that yield to real-time 3D patient pose estimation from pressure sensor data. 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 (such as A1, A2, A5 in Fig. 4). On the other hand, poses where the subject has less contact with the surface (such as A4) yield to the highest error (see Fig. 4). However, even in those cases, the estimated pose does not significantly differ from the real patient posture (See Fig. 4). Moreover, we have shown that the ConvNet performs better among the two approaches. It generalizes not only to different patient size and weight, but also to positions belonging to activities not compiled in the training data. This generalization capability is useful when dealing with patients of different medical condition. Providing the pose estimation for bed-monitoring, in clinical applications, has a great potential from classification of activities, posture analysis. In addition, in our evaluations, we estimate the pose continuously, however it is in our interests to add a movement detection method. We plan to explore this path in future work.

Conflict of Interest The authors declare that they have no conflict of interest.

Ethical approval All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards. For this type of study formal consent is not required. This article does not contain any studies with animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Achilles F, Ichim AE, Coskun H, Tombari F, Noachtar S, Navab N (2016) Patient mocap: Human pose estimation under blanket occlusion for hospital monitoring applications. In: MICCAI (1), Lecture Notes in Computer Science, vol 9900, pp 491–499

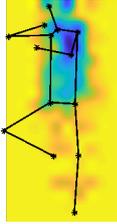
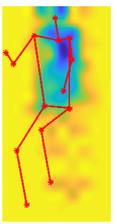
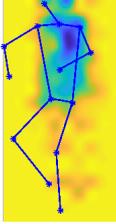
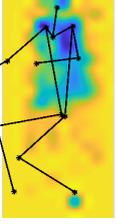
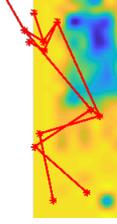
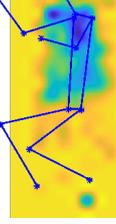
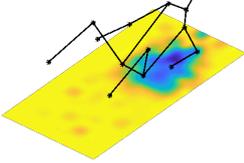
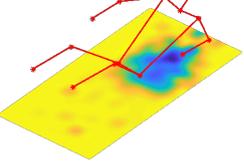
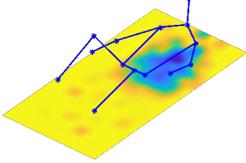
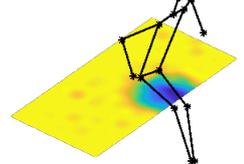
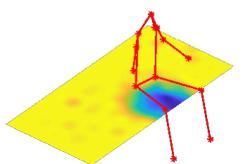
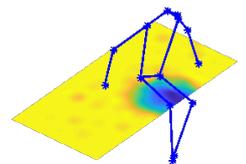
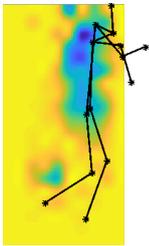
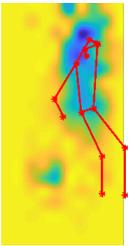
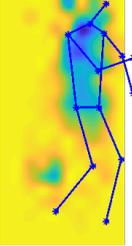
Activity	Ground Truth	Hashing	ConvNet
A1			
A2			
A3			
A4			
A5			

Fig. 4 Examples from pose estimation using hashing and ConvNet methods. The left column indicate the activity. See Table 1 for activity description

- Alaziz M, Jia Z, Liu J, Howard R, Chen Y, Zhang Y (2016) 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), pp 183–192, DOI 10.1109/CHASE.2016.13
- Auvinet E, Multon F, Saint-Arnaud A, Rousseau J, Meunier J (2011) Fall detection with multiple cameras: An occlusion-resistant method based on 3-d silhouette vertical distribution. *IEEE transactions on information technology in biomedicine* 15(2):290–300

4. Banerjee T, Enayati M, Keller JM, Skubic M, Popescu M, Rantz M (2014) Monitoring patients in hospital beds using unobtrusive depth sensors. In: Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE, IEEE, pp 5904–5907
5. Belagiannis V, Zisserman A (2017) Recurrent human pose estimation. In: Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on, IEEE, pp 468–475
6. Belagiannis V, Rupperecht C, Carneiro G, Navab N (2015) Robust optimization for deep regression. 2015 IEEE International Conference on Computer Vision (ICCV) pp 2830–2838
7. Belagiannis V, Wang X, Shitrit HB, Hashimoto K, Stauder R, Aoki Y, Kranzfelder M, Schneider A, Fua P, Ilic S, Feussner H, Navab N (2016) Parsing human skeletons in an operating room. *Mach Vision Appl* 27(7):1035–1046, DOI 10.1007/s00138-016-0792-4, URL <http://dx.doi.org/10.1007/s00138-016-0792-4>
8. Casas L, Mürwald C, Achilles F, Mateus D, Huber D, Navab N, Demirci S (2018) Human pose estimation from pressure sensor data. In: Maier A, Deserno TM, Handels H, Maier-Hein KH, Palm C, Tolxdorff T (eds) *Bildverarbeitung für die Medizin 2018*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 285–290
9. Chu X, Yang W, Ouyang W, Ma C, Yuille AL, Wang X (2017) Multi-context attention for human pose estimation. arXiv preprint arXiv:170207432 1(2)
10. Gong Y, Lazebnik S, Gordo A, Perronnin F (2013) Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(12):2916–2929, DOI 10.1109/TPAMI.2012.193
11. Grimm R, Bauer S, Sukkau J, Hornegger J, Greiner G (2012) Markerless estimation of patient orientation, posture and pose using range and pressure imaging. *International Journal of Computer Assisted Radiology and Surgery* 7(6):921–929, DOI 10.1007/s11548-012-0694-5, URL <https://doi.org/10.1007/s11548-012-0694-5>
12. Harada T, Sato T, Mori T (2001) 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, pp 3201–3207 vol.4, DOI 10.1109/ROBOT.2001.933111
13. Heydarzadeh M, Nourani M, Ostadabbas S (2016) In-bed posture classification using deep autoencoders. In: Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the, IEEE, pp 3839–3842, DOI 10.1109/EMBC.2016.7591565
14. Hoque E, Dickerson RF, Stankovic JA (2010) Monitoring body positions and movements during sleep using wisps. In: *Wireless Health 2010*, ACM, New York, NY, USA, WH '10, pp 44–53, DOI 10.1145/1921081.1921088
15. Kadkhodamohammadi A, Gangi A, de Mathelin M, Padoy N (2017) A multi-view rgb-d approach for human pose estimation in operating rooms. In: *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, IEEE, pp 363–372
16. Kehl W, Tombari F, Navab N, Ilic S, Lepetit V (2016) Hashmod: A hashing method for scalable 3d object detection. arXiv preprint arXiv:160706062
17. Kortelainen JM, Van Gils M, Pärkkä J (2012) Multichannel bed pressure sensor for sleep monitoring. In: *Computing in Cardiology (CinC), 2012*, IEEE, pp 313–316
18. Kulis B, Grauman K (2009) Kernelized locality-sensitive hashing for scalable image search. In: *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE, pp 2130–2137
19. Newell A, Yang K, Deng J (2016) Stacked hourglass networks for human pose estimation. In: *European Conference on Computer Vision*, Springer, pp 483–499
20. Papandreou G, Zhu T, Chen LC, Gidaris S, Tompson J, Murphy K (2018) Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. arXiv preprint arXiv:180308225
21. Papazov C, Burschka D (2010) An efficient ransac for 3d object recognition in noisy and occluded scenes. In: *Asian Conference on Computer Vision*, Springer, pp 135–148
22. Placht S, Stancanelli J, Schaller C, Balda M, Angelopoulou E (2012) Fast time-of-flight camera based surface registration for radiotherapy patient positioning. *Medical physics*

-
- 39(1):4–17
23. Shakhnarovich G, Viola P, Darrell T (2003) Fast pose estimation with parameter-sensitive hashing. In: null, IEEE, p 750
 24. Toms M (2000) Patient movement detection. URL <https://www.google.com/patents/US6036660>, uS Patent 6,036,660
 25. Vedaldi A, Lenc K (2015) Matconvnet – convolutional neural networks for matlab. In: Proceeding of the ACM Int. Conf. on Multimedia
 26. Wei SE, Ramakrishna V, Kanade T, Sheikh Y (2016) Convolutional pose machines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4724–4732
 27. Yu X, Zhang S, Liu B, Zhong L, Metaxas DN (2013) Large scale medical image search via unsupervised pca hashing. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on, IEEE, pp 393–398
 28. Zhang X, Liu W, Dundar M, Badve S, Zhang S (2015) Towards large-scale histopathological image analysis: Hashing-based image retrieval. IEEE Transactions on Medical Imaging 34(2):496–506