A Deep Learning Approach to 3D Human Walking Pose Estimation From SensFloor Capacitive Signals

Anonymous Author(s)

Abstract

While camera-based systems are the dominant approach for human pose estimation, they face challenges regarding privacy concerns and occlusion problems. To assess an alternative to camera-based pose estimation, this paper aims to predict 3D walking poses using SensFloor: a capacitance-based floor that registers movement activity. We analyze the potential of utilizing the floor's low-resolution signals to estimate poses and to what extent certain joint positions can be predicted. For this purpose, we collected synchronized SensFloor signals and video data, from which we extracted 3D human poses using MediaPipe to serve as ground-truth targets for training. These signals and their corresponding targets were then used for supervised training of an LSTM neural network. To estimate a person's position on the floor, we applied a clustering approach followed by a Kalman filter to smooth the trajectory. Our results demonstrate that it is possible to predict simple human walking poses using the proposed methods, establishing a proof-of-concept for an alternative way of activity monitoring.

Keywords

3D Human Pose Estimation, SensFloor, Capacitive Floor, Deep Learning, CNN-LSTM, Kalman Filter, MediaPipe

Code: https://github.com/sensfloor

1 Introduction

Human pose estimation is a fundamental task in various domains, like healthcare, rehabilitation, and gait analysis [1]. To this end, camera-based systems are a widely adopted approach due to their ease of setup and efficiency. However, the performance of such systems is highly dependent on camera placement and the field of view, making them susceptible to occlusion. Furthermore, continuous video monitoring raises significant privacy concerns, especially in sensitive environments such as elderly care, where pose monitoring could support caregivers. Consequently, alternative, privacy-preserving sensing approaches are required. Considering these issues, one promising alternative for estimating human poses is floor-based sensing technology. Generally, there are two ways to measure activity using floors: pressure- and capacitance-based. In this study, we examine SensFloor, developed by Future-Shape GmbH, as an example of a capacitance-based floor. As it is easy to install, SensFloor can be used in various domains, like healthcare, safety monitoring and retail analytics. It is composed of multiple rectangular sensing areas, hereafter referred to as patches. Each patch consists of eight triangular sensors, referred to as fields. Each field registers changes in capacitance and outputs a signal between 127 and 255 sampled at 10 Hz. The signal's intensity depends on the magnitude of the detected change. Since the human body contains mostly water, walking over the floor alters the capacitance, which SensFloor uses to detect human movement [5, 71].

SensFloor has been used in several prior studies. It has been leveraged to estimate age from walking behavior, identify individuals based on characteristic movement patterns and recognize gait patterns [3, 4, 8]. While these studies demonstrate the capability of SensFloor for tracking and behavioral analysis, they primarily address 2D localization or classification tasks rather than pose estimation. For the task of human pose estimation, different studies examined pressure-based sensors integrated in floor coverings or shoe insoles using deep-learning techniques [6, 9]. However, the feasibility of estimating full-body 3D walking poses from capacitive signals remains largely unexplored.

This study aims to determine whether 3D human walking poses can be estimated solely from SensFloor signals and to evaluate the estimation accuracy. For this purpose, we created a labeled dataset, trained a neural network to estimate poses, extracted the subjects' position from the floor, smoothed the trajectory using a Kalman filter and implemented a frontend for visualization.

2 Methods

To achieve the goal of estimating poses using a SensFloor and visualizing them, we split up the problem into two tasks: pose estimation and localization, which are illustrated in Figure 1.

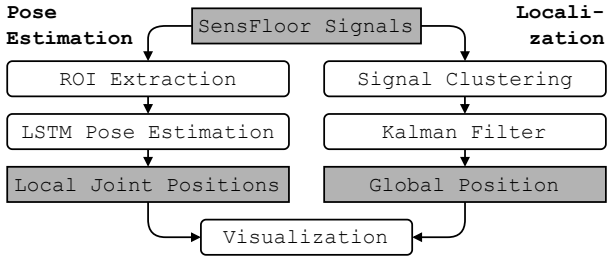

Figure 1: Overview of the SensFloor-based pose estimation pipeline — The left branch estimates the person's pose in a local, position-independent coordinate system, while the right branch localizes the person globally on the floor.

The first task of pose estimation uses a trained deep-learning model that estimates the person's joint coordinates in a local coordinate system. For that, we collected a labeled dataset of SensFloor signals and target poses. Using this dataset, we trained a Long Short-Term Memory (LSTM) model to predict human poses from sensor activations within an extracted Region of Interest (ROI). The second task of localization extracts the global position of the pose on the entire floor by signal clustering and smoothing the resulting position using a Kalman filter. Finally, the estimated pose and position are combined and visualized in a real-time simulation of the SensFloor environment.

By splitting the problem into these two tasks, the system requires only a small extracted active area of the floor for pose estimation, enabling it to be ported to floors of different dimensions.

2.1 Data Collection

To enable supervised training of the LSTM model, we collected a dataset of SensFloor signals and corresponding pose targets. Over two sessions, we recorded a total of eight hours of synchronized SensFloor signals and video footage at 15 FPS from three participants walking on the floor. Moreover, we captured some short sequences from three additional participants that were used to assess the model further. To synchronize the data, we matched each received SensFloor signal to the video frame recorded at the same time. To extract the final 3D pose targets from the video feed, we used MediaPipe's pose estimation model, which builds upon the BlazePose architecture [2]. It is worth noting that the origin of MediaPipe's pose estimates coordinate system is located at the center of the hips. This allows the decomposition of the paper's problem into the previously introduced two tasks of pose estimation relative to the hip and localization on the global floor.

2.2 Data Preprocessing

Before inputting data into the model, we preprocess it in multiple steps. First, signals below a threshold that likely correspond to noise are filtered out. This threshold is defined as $\tau = 140$ for this paper, though it could vary depending on the floor setup and its surface material. Then, the frames at which a significant SensFloor signal was measured are selected and assigned the pose of that frame. This results in a dataset containing both significant signals as inputs and corresponding poses as targets.

When loading a data sample, the input signals from the triangular sensor fields are interpolated into a grid. For that purpose, every two adjacent fields forming a rectangle are transformed into a 2×2 square where the corners are assigned the original signal value and the diagonal fields their mean. Figure 2 depicts this transformation.

Afterward, a ROI history is created, which is the temporal sequence of the floor areas with the highest activations. The history length is defined as 50 frames and the ROI as a grid of 4×4 patches. The position of the ROI is calculated by looking back 50 frames, summing the corresponding signal values for each sensor field and selecting the area of patches with the highest total signal value.

Finally, all signals are normalized to the range $[0, 1]$, where values of 127 will be 0 and values of 255 will be 1. So for parts where there are no signals, the model receives 0.

Regarding the targets, the dataset contains a subset of 13 out of 33 joints extracted by MediaPipe. We exclude eyes, ears, mouth and fingers because their contribution to the overall pose is negligible. Additionally, the heel and foot index are excluded, because when analyzing the targets, we found that MediaPipe predicted these bones unreliably to lengths of 3 to 18 cm. With an x, y and z coordinate for each target joint, this results in $N_{out} = 13 \times 3 = 39$ continuous coordinates for the model to predict.

The final dataset is split sequence-wise with 80% for training (~174k poses), 10% for validating and 10% for testing (each ~21k poses).

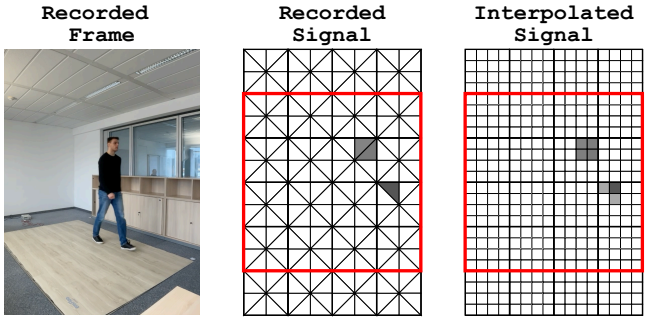

Figure 2: Visualization of SensFloor signal processing — To the left the video-frame recorded of the subject walking, in the center the raw recorded signal on the original triangular sensor grid and to the right the corresponding interpolated signal. The red square in both floor visualizations indicates the extracted ROI for this signal.

2.3 Pose Estimation Model

The model developed in this paper implements an LSTM architecture based on the classification model from [3]. We adapted the task from classifying walking modes to the regression task of estimating 3D joint coordinates. Furthermore, we added a CNN encoder prior to the LSTM to utilize the 2D spatial data of the SensFloor signals.

The CNN encoder consists of three sequential 2D convolutional layers with 3×3 kernels, increasing the channel size from 32 to 64 and then 128. Each convolution is followed by batch normalization and a ReLU activation, with 2×2 max pooling applied after the second and third layer. The resulting feature maps are flattened and passed through a linear layer to obtain a 256-dimensional embedding. This embedding is then input to the LSTM with a hidden size of 256, using a dropout of 0.2 during training. Finally, the regression head processes the last hidden state through a four-layered neural network with layer sizes $[256, 256, 128, N_{out}]$ and ReLU activations, respectively linear activation for the last layer.

The model was trained for a maximum of 40 epochs using early-stopping with patience 5 and a batch size of 32. Optimization was performed via AdamW with a learning rate of 1×10^{-4} and a weight decay of 1×10^{-2}.

To assess model performance, we utilize two metrics: Mean Per Joint Position Error (MPJPE) and Percentage of Correct Key Points (PCK) [10, 22 f.]. MPJPE quantifies the average Euclidean distance in cm between estimated and target joints, while PCK measures the percentage of estimates whose distance errors fall below a threshold, which we set at 5 cm (PCK@5) and 10 cm (PCK@10).

2.4 Loss Function

The CNN-LSTM model receives the signals in the ROI sequentially and produces N_{out} coordinate estimates. These coordinates are passed to the loss function, defined as:

$$L_{total} = L_{\text{WMSE}} + \lambda \cdot L_{\text{link}}, \tag{1}$$

where λ is a fixed scalar for weighting L_{link}. For training the model, we set $\lambda = 0.1$. The L_{WMSE} (weighted mean squared error) calculates the squared difference between the estimated joint coordinates and

their target values, weighted by specific landmark importance:

$$L_{\text{WMSE}} = \frac{1}{B \cdot J \cdot 3} \sum_{i=1}^{B} \sum_{j=1}^{J} \sum_{k=1}^{3} w_j (\hat{y}_{i,j,k} - y_{i,j,k})^2, \tag{2}$$

where B is the batch size, J the number of joints, k indexes the x, y and z coordinates of each joint, $\hat{y}_{i,j,k}$ the model's predictions and $y_{i,j,k}$ the targets. We increased the weight for the knees and feet to 5 compared to 1 for all other joints, because they are the most important components for walking. For each estimated joint pair, we furthermore calculate the Euclidean distance d, namely the bone length (e.g. of the lower leg) and use these values in the link loss function adopted from [6]. It is defined as the mean error over the set of all links (l):

$$L^l_{\text{link}}(d) = \begin{cases} k^l_{\text{min}} - d, & \text{if } d < k^l_{\text{min}} \\ d - k^l_{\text{max}}, & \text{if } d > k^l_{\text{max}} \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where k^l_{min} and k^l_{max} are the 3$^{\text{rd}}$ and 97$^{\text{th}}$ percentiles of the link lengths in the training set. The link loss is employed to enforce anatomical consistency and maintain correct proportions of body part lengths.

2.5 Localization

To integrate the local pose estimate from the model into the global coordinate system of the floor, we developed a pipeline that tracks the subject's hip position relative to the SensFloor. We extract a raw pose estimate by clustering the measured sensor activations and applying a Kalman filter to smooth the trajectory.

We identify the subject's contact points with the floor by grouping adjacent active sensor fields into clusters. The hip position is then calculated the in following way: during a step, when both feet touch the ground, two clusters can be measured. In this case, we define the hip position as the midpoint between the centroids of the two clusters. If only one cluster is detected, meaning one foot is off the ground or both feet are close together, we assume the hip is located directly above that single centroid. This logic is illustrated in Figure 3 and allows us to extract the hip position from the received signals.

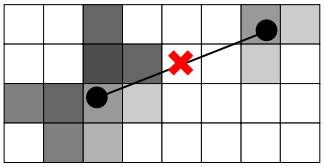

Figure 3: Global position estimation from signals — Two clusters of SensFloor activations are identified and their centroids (black dots) are computed. The final position is obtained as the midpoint between the two centroids (red cross).

Because the raw position estimates are noisy and lead to erratic, unnatural movement in the visualization, we apply a Kalman Filter to ensure a smooth trajectory. Similar to [7, 255 ff.], the filter's state vector is defined as $\mathbf{x} = [x, y, \dot{x}, \dot{y}]^T$, where x and y are the position coordinates and $\dot{x}$ and $\dot{y}$ the corresponding velocities. Due to the absence of ground-truth tracking data for exact calibration, we tuned the filter's noise parameters empirically until the trajectory visually matched the subject's actual movement in the recorded videos.

2.6 Visualization of Predicted 3D Pose

For visual evaluating and demonstrating of the pose estimation pipeline, we furthermore implemented a frontend. The estimated pose and its position are streamed to the frontend using a WebSocket. The frontend then visualizes a simulated environment of the floor and the person's pose moving on it.

3 Results

This section presents an evaluation of the results and a discussion of the model's performance, followed by an examination of the resulting trajectories from the localization pipeline.

3.1 SensFloor Pose Estimation Model

Model training reached early stopping after 22 epochs, achieving an overall performance of 5.5 cm MPJPE, 87% PCK@10 and 63% PCK@5. In the following, we visually compare an estimated pose to its target and analyze the performance across specific joints.

3.1.1 Visual Comparison of Estimated and Target Pose. Figure 4 illustrates a visual comparison between the estimated pose and the corresponding ground-truth from the test set, shown from four perspectives rotated at 0°, 90°, 180° and 270° angles. Overall, the estimated pose captures the global body posture of the target well. In particular, the global orientation of the skeleton, the hip and the left-leg joint positions align closely with the ground-truth. However, discrepancies are visible in the right lower leg, which appears shifted further back relative to the ground-truth. Furthermore, the estimated upper body exhibits a slightly stronger forward lean than the reference.

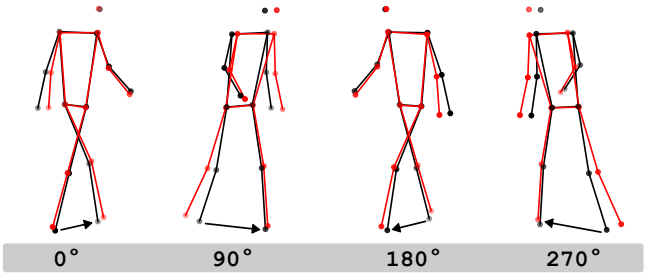

Figure 4: Comparison of a single estimated 3D pose (red) to its corresponding ground-truth pose (black) from four different perspectives. The arrows indicate the step direction.

3.1.2 Joint Estimations. Our recorded metrics for the test set, illustrated in Figure 5, mirror the results of the visual comparison. The hip was the easiest joint to estimate, with a median error of approximately 2 cm. This is because, in the poses' local coordinate system, the hips mostly rotate proximally around the origin and do not move a lot. In contrast, the wrists and ankles, as the most distal joints, move the most and performed the worst, with medians of

approximately 5 cm and 7 cm and a larger interquartile range. If the model were to always predict the mean pose, the overall error for all test targets would be roughly 23 cm for the wrists and 18 cm for the ankles. Compared to this baseline, our model represents a significant improvement.

Figure 5: Boxplot of Mean Per Joint Position Error (MPJPE) for each joint on the test set with $n = 21898$ samples. The box indicates the interquartile range and the whiskers mark the 5^{th} and 95^{th} percentile.

While most of the data shows a reasonable error, a significant portion contains higher errors. We explained this for two reasons. First, when the subject performs an unforeseen action by rapidly changing the directions or moving their body unnaturally, for example by looking at their watch, the model is unable to predict this. Second, when the subject has just entered the SensFloor, no history information is available, forcing the model to guess the stepping foot, potentially resulting in a wrong pose. Empirical tests during inference confirmed these sources of error.

Furthermore, the ankles exhibit a significantly higher error than the elbows and wrists. This finding is counterintuitive, as the proximity of the ankles to the floor would suggest a higher localization accuracy for these joints. A potential explanation may lie in the target pose extraction. MediaPipe tends to estimate the same bone lengths across subjects of varying heights. These fixed proportions may lead to a misalignment between the SensFloor signals and the corresponding joint coordinates, which most significantly impacts the joints close to the floor.

3.2 Localization and Kalman Filter Effect

In 2.5, we introduced our approach for extracting information about the person's current global position from the SensFloor activation signals. The left side of Figure 6 shows the raw clustering position trajectory we recorded during a test walk. The illustrated trajectory is highly erratic. This instability is caused by two main factors. First, the algorithm creates jumps in the estimated position as the mean shifts abruptly whenever a foot makes or breaks contact with the floor. Second, the SensFloor fields produce significant background noise, even in areas without activity. While increasing the noise signal threshold suppresses some of the noise, it is not a universal solution, as the signal intensity of people moving on the floor varies depending on the person's footwear.

However, applying the Kalman filter largely reduces these issues, as illustrated on the right side of Figure 6. By smoothing abrupt transitions and mitigating the impact of outliers, the filter produces a trajectory, that is both smooth and, according to our empirical evaluation, accurate.

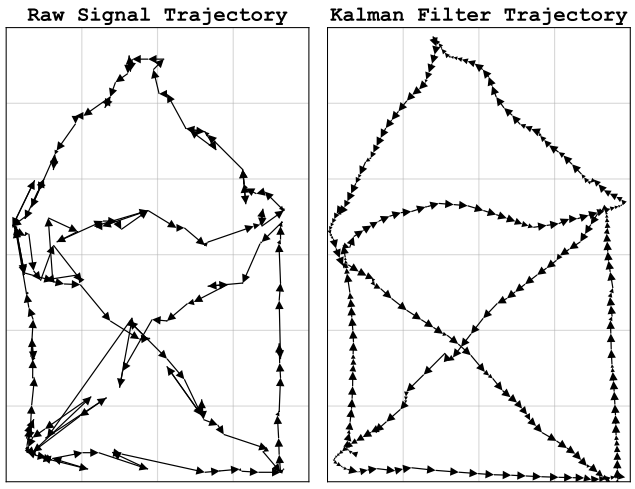

Figure 6: Comparison of raw (left) and filtered (right) localization trajectories. The arrows indicate the direction of the movement, a shorter movement results in a smaller arrow.

4 Conclusion

This study investigated the feasibility of estimating 3D walking poses from SensFloor signals. Our results indicate that simple walking poses can be predicted, which could support privacy-preserving monitoring in applications such as elderly care.

The present work has several limitations. First, the model is currently tied to the patch size it was trained on and does not transfer to other SensFloor configurations with different patch sizes. Second, the evaluated walking poses are relatively homogeneous, and the dataset contains only three male participants. Training and evaluating on more diverse movements and a broader range of participants would be necessary to assess generalization. Third, the ground-truth poses obtained via MediaPipe are themselves subject to estimation error, which places an upper bound on the achievable accuracy.

Future work should therefore improve the transfer to other SensFloor patch sizes, incorporate more diverse data and use more accurate pose targets to better quantify the limits and potential of the approach.

References

[1] Andrea Avogaro, Federico Cunico, Bodo Rosenhahn, and Francesco Setti. 2023. Markerless Human Pose Estimation for Biomedical Applications: A Survey. 5 (2023). doi:10.3389/fcomp.2023.1153160

[2] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. 2020. *BlazePose: On-device Real-time Body Pose Tracking*. doi:10.48550/arXiv.2006.10204

[3] Raoul Hoffmann, Hanna Brodowski, Axel Steinhage, and Marcin Grzegorzek. 2021. Detecting Walking Challenges in Gait Patterns Using a Capacitive Sensor Floor and Recurrent Neural Networks. 21, 4 (2021), 1086. doi:10.3390/s21041086

[4] Raoul Hoffmann, Christl Lauterbach, Jörg Conradt, and Axel Steinhage. 2018. Estimating a Person's Age from Walking over a Sensor Floor. 95 (2018), 271–276. doi:10.1016/j.compbiomed.2017.11.003

[5] C. Lauterbach, A. Steinhage, and A. Techmer. 2013. A Large-Area Sensor System Underneath the Floor for Ambient Assisted Living Applications. In *Pervasive and Mobile Sensing and Computing for Healthcare*, Subhas Chandra Mukhopadhyay and Octavian A. Postolache (Eds.). Vol. 2. Springer Berlin Heidelberg, 69–87. doi:10.1007/978-3-642-32538-0_3

[6] Yiyue Luo, Yunzhu Li, Michael Foshey, Wan Shou, Pratyusha Sharma, Tomas Palacios, Antonio Torralba, and Wojciech Matusik. 2021. Intelligent Carpet: Inferring 3D Human Pose from Tactile Signals. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Nashville, TN, USA, 2021-06). IEEE, 11250–11260. doi:10.1109/CVPR46437.2021.01110

[7] Roger R. and Labbe. 2024. Kalman and Bayesian Filters in Python. (2024). https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python

[8] M. Sousa, A. Techmer, A. Steinhage, C. Lauterbach, and P. Lukowicz. 2013. Human Tracking and Identification Using a Sensitive Floor and Wearable Accelerometers. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (San Diego, CA, 2013-03). IEEE, 166–171. doi:10.1109/PerCom.2013.6526728

[9] Atsuya Watanabe, Ratna Aisuwarya, and Lei Jing. 2025. *P2P-Insole: Human Pose Estimation Using Foot Pressure Distribution and Motion Sensors*. doi:10.48550/arXiv.2505.00755

[10] Ce Zheng, Wenhan Wu, Chen Chen, Taojiannan Yang, Sijie Zhu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. 2023. Deep Learning-based Human Pose Estimation: A Survey. 56, 1 (2023), 1–37. doi:10.1145/3603618