Neural Networks for Indoor Human Activity Reconstructions

Neural Networks for Indoor Human Activity Reconstructions

Osama Bin Tariq, *Student Member, IEEE,* Mihai Teodor Lazarescu, *Senior Member, IEEE,* and Luciano Lavagno, *Senior Member, IEEE*

***Abstract*—Low cost, ubiquitous, tagless, and privacy aware indoor monitoring is essential to many existing or future applications, such as assisted living of elderly persons. We explore how well different types of neural networks in basic configurations can extract location and movement information from noisy experimental data (with both high-pitch and slow drift noise) obtained from capacitive sensors operating in loading mode at ranges much longer that the diagonal of their plates. Through design space exploration, we optimize and analyze the location and trajectory tracking inference performance of multilayer perceptron (MLP), autoregressive feedforward, 1D Convolutional (1D-CNN), and Long-Short Term Memory (LSTM) neural networks on experimental data collected using four capacitive sensors with 16 cm x 16 cm plates deployed on the boundaries of a 3 m x 3 m open space in our laboratory. We obtain the minimum error using a 1D-CNN [0.251 m distance Root Mean Square Error (RMSE) and 0.307 m Average Distance Error (ADE)] and the smoothest trajectory inference using an LSTM, albeit with higher localization errors (0.281 m RMSE and 0.326 m ADE). 1D Convolutional and window-based neural networks have best inference accuracy and smoother trajectory reconstruction. LSTMs seem to infer best the person movement dynamics.**

***Index Terms*—Indoor localization, movement tracking, capacitive sensors, CNN, LSTM, autoregressive, multilayer perceptron.**

I. Introduction

A growing number of smart space applications rely on indoor person localization and activity recognition for safety monitoring, providing added value services, or continuous assistance. For instance, assisted living applications can lower assistance costs, improve safety and quality of life, which are increasingly important with a projected ratio between working-age and elderly people of 3.5 by 2050 [1].

Indoor person localization can rely on wearable or portable devices [2], such as the IEEE 802.11 (Wi-Fi) [3], [4] or the Bluetooth [5] standards, low power communications using the ZigBee protocol [6], radio frequency identification (RFID) [7], ultra wideband radio (UWB) [8], visible light communication [9], or audible [10] and ultrasound [11] acoustic signals.

However, tagless indoor localization [12], [13] is necessary whenever the persons may not carry or wear a device to be sensed by the localization system, such as in some smart home applications or assisted living for elderly people. Furthermore, to improve the localization system acceptance and added value, it should ensure personal privacy, be affordable, unobtrusive, easy to install, and require little or no maintenance (e.g., long battery life or use wireless power).

Many localization techniques have been proposed [14], such as pressure and load cells [15], sensing mats [16], thermal infrared [17], sound source [18], ultrasound reflections [19], [20], air pressure [21], residential power lines [22], water usage [23], optical [24], carbon dioxide [25], vital functions [26], and data fusion from various sensor types [27]. Electric field sensors [28] used for localization include capacitive tiles [29], electric resonance coupling [30], and capacitive coupling [31]. They use the conductive properties of the human body and do not require the person to carry any specific device.

Capacitive sensors can operate in loading mode [32] (see below for other modes) using only one plate [33], can be self-contained, easy to install, inconspicuous, privacy-observant, and inexpensive. They can sense [34], [35], identify [36], [37], and localize [38]–[41] persons indoor, but their sensitivity decreases steeply with the distance. Long range sensing, at distances of 10–15 times the plate diagonal, are highly susceptible to several environmental factors, such as electromagnetic and electrostatic noise, humidity, or temperature [42].

In this work, we explore how advanced data processing can improve capacitive sensor accuracy [43], [44]. We use relatively simple capacitive sensors, which are known to be more susceptible to environmental noise, to better compare the effectiveness of the data processing chains on the overall accuracy. We use several signal filtering combinations for preprocessing and then we optimize different Neural Network (NN) types through design space exploration (DSE). For NN training, validation, and testing, we use capacitive sensor data collected while a person moves arbitrarily in a $3\,\mathrm{m} \times 3\,\mathrm{m}$ experimental room, and we compare the inferred position and trajectory with the reference location acquired using an accurate ultrasound localization system.

The main contributions of this work are:

- Neural Network-based signal processing techniques for indoor person localization and tracking using small capacitive sensors operating in loading mode at long ranges (up to 10-15 times their plate diagonal);
- noise attenuation using various kinds of digital filters and neural networks (NN) for location and trajectory inference;
- comparative analysis of NN-based location and movement dynamics inference accuracy from noisy sensor data.

We believe that this is the first study of tagless human movement trajectory estimation using capacitive sensors, while previous works focused mostly on static localization.

O. Bin Tariq, MT. Lazarescu and L. Lavagno are with the Department of Electronics and Telecommunications, Politecnico di Torino, I-10129 Torino, Italy, e-mail: osama.bintariq@polito.it

Manuscript received Month XX, 20ZZ; revised Month YY, 20ZZ.

II. Related work

Our work combines long range capacitive sensors, digital filters, and neural networks to infer the position and the trajectory of a person indoors.

Capacitive sensors have many applications [35], which include tagless human body motion detection, indoor localization, person identification and biometrics. Loading mode capacitive sensor operation is demonstrated for various sensing applications in [42], inclusing human activity monitoring using sensors installed behind furniture. Similarly, capacitive sensors placed under the bed are demonstrated to monitor sleep patterns [45], and concealed sensors in ambient assisted living environments can classify the postures of several persons [46]. Mobile capacitive sensors attached to limbs are used to monitor slow movements of elderly persons or people with movement-affecting conditions that limit the effectiveness of typical accelerometer- or image-based monitoring sensors [47].

We note that in these works the sensors operate within relatively short ranges. This limits the effectiveness of the systems for tracking the free movement of a person within the space of a typical room, and also improves the signal-to-noise ratio, which simplifies the signal processing needed to reconstruct the observed physical behavior.

Person localization using capacitive sensors is proposed usually through distributed deployment of sensor or excitation electrodes within the monitored space. A proposed solution uses excitation wire or plate electrodes under the room floor and a receiver wire on a wall to achieve sensing ranges up to $2\,\mathrm{m}$ and feet localization errors up to $40\,\mathrm{cm}$ [48]. Excitation plates in or under the floor coupled through the person body with sensing electrodes on the ceiling are proposed to monitor the location, height, and posture indoor, including detection of falls [49], [50]. Localization for elderly persons monitoring in assisted living application was also proposed using impedance variations of an array of thick film sensors distributed under the floor of rooms up to $10\,\mathrm{m}$ wide [51]. Operating on similar principles, floor tiles instrumented with capacitive sensors have been repeatedly proposed for indoor person localization [52], also using data fusion [53] from capacitive sensors operating in shunt mode, pressure, vibration, and various radio frequency sensors. The sensors can be embedded in large surfaces like the floor, living room table, or beds. For inconspicuous indoor person localization, past research also considered distributed capacitive sensors made using printed electronics, which can be embedded in carpet textiles, linoleum, laminate, tiles or stone covering the floor or other large surfaces or objects in the room [54].

In general, distributed deployment of sensors or electrodes can be costly and may require significant works on home surfaces, e.g., floor, ceiling, walls, furniture. But since they usually sense within relatively close range, they have good signal-to-noise margins and can localize persons and postures with good accuracy and limited signal processing.

1D Convolutional NNs (CNNs), Long-Short Term Memory (LSTM) NNs and their variants were used in multiple sensing applications. In [55], the authors survey the use of NN architectures for human activity recognition, including 1D CNNs, Recurrent Neural Networks (RNNs), and hybrid architectures using data from various sensors to classify human activities. LSTM networks were used for indoor static localization using the magnetic and light sensors that are included in the modern smartphones [56]. They do so in close proximity to the site where they trained the system (a $6\,\mathrm{m} \times 12\,\mathrm{m}$ lab) to preserve the validity of the magnetic field calibration. The authors of [57] used Wi-Fi fingerprinting for LSTM-assisted indoor discrete localization of multiple persons in a research lab of $35.3\,\mathrm{m} \times 16.0\,\mathrm{m}$ and in an office of $55\,\mathrm{m} \times 50\,\mathrm{m}$.

We build on our previous work [58], where we used long range capacitive sensors to infer the discrete position (out of a predefined set) of a person in an experimental room using machine learning classifiers. We also improve on our previous work [59], in which we used only an autoregressive NN and digital filters to infer the person trajectory, and explored only the effects of varying the window size of the preprocessing median filter and the NN. Here we significantly extend the exploration of the NN inference performance by changing several network and filter parameters, as well as the network type (in basic configurations, to establish a performance baseline) while tracking the movement and the position of a person in a laboratory space. We also use relatively noisy data from strongly non-linear capacitive sensors (conditioned using digital filters) to test the NN capacity to filter out the noise, including through their ability to detect the underlying dynamics of the person's movement. We optimize both the NNs and the filter parameters through Design Space Exploration (DSE) and analyze both the overall error, as well as the smoothness and closeness of the inference to the actual person's trajectory.

III. Methodology, equipment and tools

We emulate in our laboratory a small (yet realistic with respect to a typical apartment room where an elderly person may live) room as an empty space of $3\,\mathrm{m} \times 3\,\mathrm{m}$. We monitor the position of the person within the room using two systems. The "target" system uses four capacitive sensors, each one with a sensing plate of $16\,\mathrm{cm} \times 16\,\mathrm{cm}$ installed at chest level in the center of a "wall" of the virtual room (as shown in Fig. 1), providing readings three times per second. The "reference" system (from Marvelmind Robotics [60]) is based on four ultrasound anchors that can localize a mobile tag with $\pm 2\,\mathrm{cm}$ accuracy at $15\,\mathrm{Hz}$.

The capacitive sensors are similar to our previous experiments [39], [58], [59], based on an LM555 circuit configured as astable multivibrator and using the sensor plate capacitance (see Fig. 2). The oscillation frequency f is inversely proportional to sensor plate capacitance C

$$f = \frac{k_c}{C} \tag{1}$$

through a constant k_c determined by the resistor values in the multivibrator circuit. Plate capacitance cannot be determined analytically for distances d between the plate and the human body that are much longer than the plate diagonal. Empirically, it is approximated proportional to the inverse of the distance at a power n ($n \approx 3$) through a constant k_d with an offset C_0

$$C \approx C_0 + \frac{k_d}{d^n}, \tag{2}$$

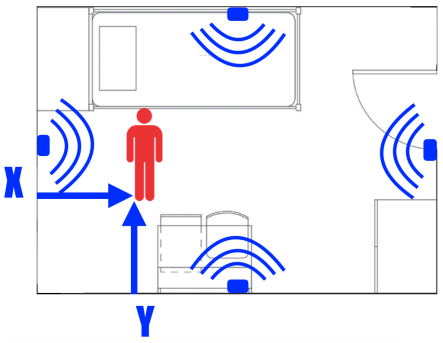


Fig. 1. Four capacitive sensors centered on the walls of a $3\,\mathrm{m} \times 3\,\mathrm{m}$ virtual room in the lab trace the position of a person moving in the space.

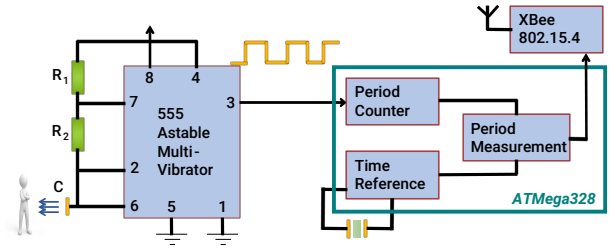


Fig. 2. Schematic of capacitive sensor front-end using an astable multivibrator to convert the plate capacitance to frequency, measured by a microcontroller

where n, k_d, and C_0 depend on multiple geometric, electric, and dielectric properties [61], [62]. We improved both the sampling rate of our previous sensor from 1 Hz to 3 Hz, to adequately track a person moving indoors, as well as its discretization error from 20 ppm to 3 ppm, further lowered to 1.5 ppm through oversampling, decimation and averaging (or about $15\,\mathrm{aF}$ while measuring a plate capacitance of roughly $10\,\mathrm{pF}$). Yet, (2) shows that sensor distance resolution changes much with the distance.

However, this sensor front-end is sensitive to both high pitch and drift environmental noise, which limits its range and stability over time. In Fig. 3, we show an example of sensor data, where the high-pitch noise is mostly visible at long range, at the top of the plot, while the drift is mostly visible at the beginning, up to around the 200 sample mark, and towards the end, especially beyond the 1200 sample mark. Environmental noise typically reduces considerably the sensing range, e.g., in laboratory tests we were able to detect a person standing at a distance up to about $1.6\,\mathrm{m}$–$1.8\,\mathrm{m}$ in front of the sensor.

Nevertheless, the higher noise susceptivity of the sensors allows us to better compare how efficiently different signal processing techniques can reject environmental noise. In future work, we plan to use the best processing chain on cleaner data collected from sensor front-ends that are more resilient to noise. For instance, instead of using period modulation to measure the capacitance of the sensor transducer, we can explore front-ends based on carrier modulation in amplitude and/or phase, as well as other techniques that are less susceptible to and reject better the environmental noise.

We characterize the localization accuracy of the ultrasound-based reference system in our environment by acquiring four times per second for five seconds the location of a person that wears the tag on the head, while standing on each one of 16

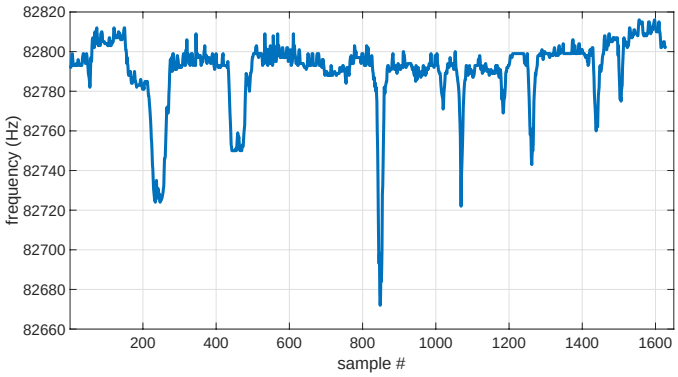


Fig. 3. Example of raw sensor data acquired at 3 Hz while a person was moving in the room. High-pitch noise is visible at the top (far end of sensor range). Slow drift is mostly visible before sample #200 and after #1200.

predefined locations inside the experimental room space. The average localization error of the system is $\pm 3.9\,\mathrm{cm}$, with a maximum error of $\pm 6.4\,\mathrm{cm}$, and a maximum standard deviation (calculated over the norm) of $\pm 0.7\,\mathrm{cm}$. We note that the absolute localization error in our setting is higher than the $\pm 2\,\mathrm{cm}$ reported by the producer, but with a good stability.

In this setting, we record concurrently the reference position of the person using the ultrasound system (ground truth) and the capacitive sensor readings (see Fig. 4). Then we

1) translate the average of the latter to zero,
2) pass it through a wide window (50 s) Median Filter (MF), to extract the slow drift,
3) pass it through a Low-Pass Filter (LPF) with a pass-band edge of 0.1 Hz and a stop-band edge of 0.6 Hz, to reduce high-pitch noise (see both traces in Fig. 5(a)),
4) and finally, we subtract the median filter output from the LPF output and
5) normalize the values to $[0, 1]$ range to use them to train and test the performance of different NN types.

Note that the best values for the window and the cutoff frequency were found via an extensive Design Space Exploration, as reported in Section IV.

In the semi-logarithmic scale plot of the inverted normalized output $(1 - y)$ of the filter block shown in Fig. 5(b), we can see a rugged but relatively flat low level as effect of the median filter reducing much of the drift visible on the top side of raw sensor output in Fig. 3. Filtering effects can also be seen in Fig. 6 comparing the frequency spectrum of the raw and the filtered sensor signals. As noted in Fig. 5(b), very low frequency components are reduced by the median filter, while components above $0.3\,\mathrm{Hz}$ are attenuated by the high-pass filter below the noise level around $-60\,\mathrm{dB}$.

Comparing the frequency spectrum of the capacitive sensors with the spectrum of the location data from the ultrasound-based reference localization system, shown in Fig. 7, we note that also the noise floor of the reference system is around $-60\,\mathrm{dB}$, and that the signal emerges above it for frequencies below $0.3\,\mathrm{Hz}$. However, while the sensor spectrum flattens around $-40\,\mathrm{dB}$ for lower frequencies, the reference signal starts to increase below $0.1\,\mathrm{Hz}$ and is about $20\,\mathrm{dB}$ stronger than the sensor signal for lower frequencies, around $0.02\,\mathrm{Hz}$.

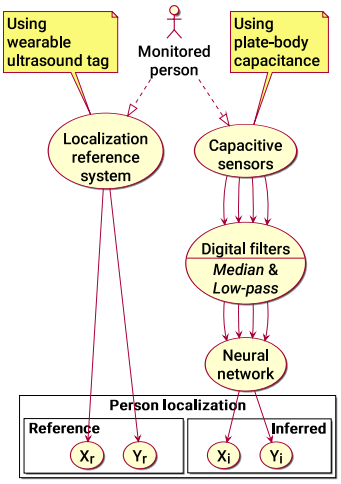


Fig. 4. Experimental data processing uses an accurate ultrasound-based reference (for training data labelling and inference testing), and the capacitive sensor processing chain with digital filters and the neural network under test.

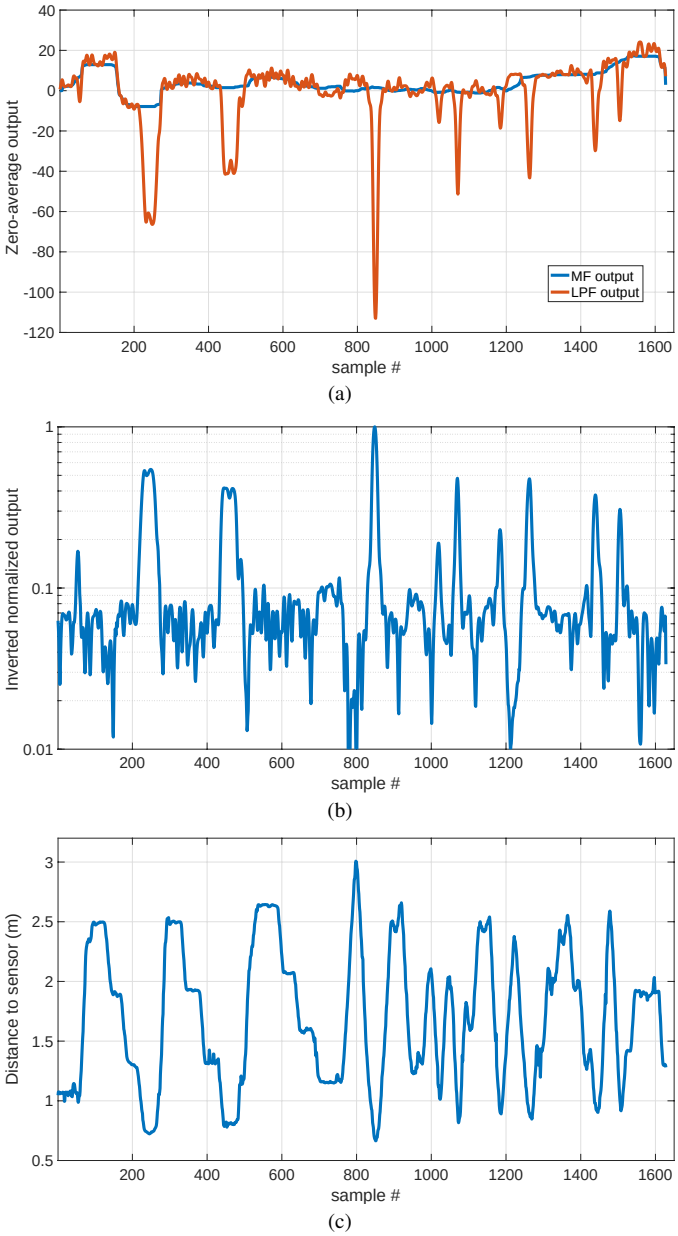


Fig. 5. Sensor output sampled at 3 Hz after (a) filtering [50 s-window median (MF) and 0.1 Hz low-pass (LPF) filters], (b) normalization (shown inverted and in semi-logarithmic scale to expose the noise), and (c) distance to person body as they roam the room

This part of the spectrum is important because most of the movement in the room during the experiment was slow, as can be expected of an elderly person, and likely contributed to the lower end of the frequency spectrum.

Comparing the filtered sensor signal in Fig. 5(b) to the plot of the distance between the person and the sensor in Fig. 5(c) (calculated from the reference system measurements), we can see a strong and well correlated sensor response when the person comes closer [the top peaks in Fig. 5(b) match the bottom peaks in Fig. 5(c)], but noise still limits sensor sensitivity at longer distances [which can be seen from the poor correlation with the distance sensor-person of the rugged lower part of sensor response in Fig. 5(b)].

Considering the high noise level of this type of sensors, we are mostly interested in how well various neural network types can extract position and trajectory information from them.

We use the Keras library with a TensorFlow [63] back-end to implement the NNs [64]. We design each NN with the same number of neurons on all hidden layers within a given network and the ReLU [65] activation function. We use the first order gradient-based optimization algorithm Adamax (with default parameters [66]), which computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients. For the LSTM networks [67], we use the default activation functions for the various gates.

IV. Design Space Exploration Results

We use two types of tests: static localization and movement tracking. For the former, we use the data that we collected in our previous classification experiment [58], namely 320 tuples collected while a person was standing in each one of the 16 positions shown in Fig. 8, i.e. $320 \times 16 = 5120$ total tuples. For the latter, we use the sensor and reference data collected while the person moves along arbitrary paths in the room for about nine minutes (1626 tuples at 3 tuples/s, see Fig. 9).

Generally, the trajectory does not come too close to the walls to better reflect the person movements in actual rooms and because sensor measurements closer to the center of the room (far from the sensors) are noisier, hence more interesting for us, because they are more difficult to interpret.

Note that in both tests we train and test the neural networks to report location estimation as a pair of X/Y coordinates, not to classify the position into a predefined set of locations.

We optimize and compare the performance of several types of neural networks in terms of Mean Square Error (MSE) and average Euclidean Distance Error (ADE) between the inferred position and the reference position (ground truth). For movement tracking, we also compare graphically the plots of the ground truth (as reported by the reference system, see Fig. 8) and the NN inference. We do this separately for the X and Y coordinates instead of 2D plots of full trajectories to

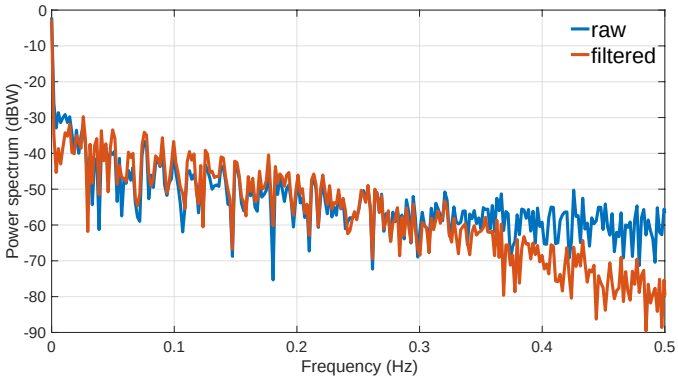


Fig. 6. Power frequency spectrum of one sensor output before ("raw") and after digital filtering ("filtered"). The median filter attenuates frequencies below 0.02 Hz and the low-pass filter attenuates frequencies above 0.3 Hz.

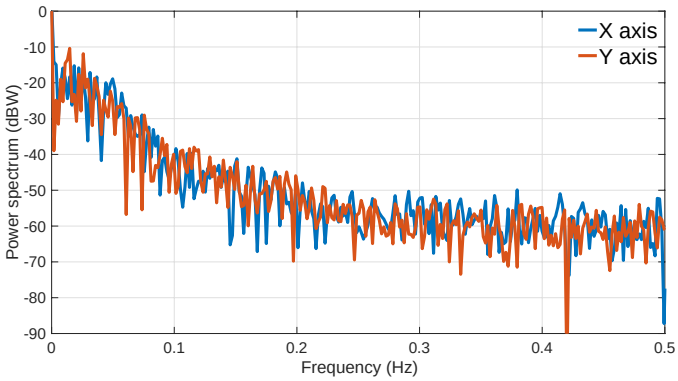


Fig. 7. Power frequency spectrum of the reference localization system output coordinates, X and Y. Below 0.3 Hz the signal rises above the noise floor and increases especially at lower frequencies, corresponding to slow movements.

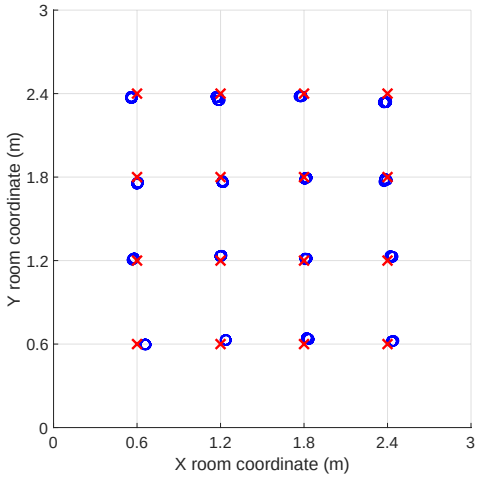


Fig. 8. Static test of the reference system in 16 locations (red 'x'). The results shows very good stability (blue circles) and adequate accuracy.

TABLE I
MEAN SQUARE ERROR (MSE) AND AVERAGE DISTANCE ERROR (ADE) FOR MULTILAYER PERCEPTRON NEURAL NETWORK INFERRING STATIC LOCATIONS

	Number of hidden layers							
	1		3		4		5	
Neurons per layer	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)
4	0.116	0.373	0.074	0.299	0.073	0.298	0.065	0.271
8	0.084	0.324	0.062	0.262	0.053	0.235	0.051	0.225
16	0.076	0.307	0.048	0.224	0.042	0.203	0.039	0.188
32	0.072	0.302	0.037	0.186	0.030	0.159	0.026	0.142
64	0.063	0.273	0.026	0.150	0.024	0.137	**0.022**	**0.124**

visualize better inference discrepancies from ground truth and to comparatively analyze the accuracy of different NNs.

We discretize sensor data (shown on top-left of Fig. 10) at 3 Hz into four-sample tuples, $S_1, \ldots, S_4$, holding one sample for each capacitive sensor. Then we concatenate the tuples in chronological order and provide them (with suitable windowing, as we will discuss later) to the tested neural networks.

A. Static position classification with multilayer perceptron neural networks

Also here we use the preprocessed experimental data from our previous classification experiment [58]. It holds 320 capacitive sensor tuples collected when the person stood in each of the 16 positions shown in Fig. 8 (320 tuples$\times$16 positions $=$ 5120 tuples) labelled with the coordinates of each position.

Because this experiment monitors static positions, we can consider the tuples independent and use a multilayer perceptron NN (see Fig. 11), which does not consider the tuple temporal sequence. Hence, we split the experimental data randomly in subsets of size 60%, 20%, and 20% to use for NN training, validation, and testing, respectively. We use the validation set to stop the NN training before it starts overfitting, i.e. when the NN inference error on the validation set starts to increase while the error on the training set continues to decrease.

We keep the size of the input layer fixed at four neurons, equal to the number of sensors. We vary the structure of the rest of the neural network from one hidden layer with four neurons up to five hidden layers with 64 neurons each (all hidden layers have always the same number of neurons). For each configuration, we train and test the NN ten times using random initializations.

Table I shows the best MSE performance of the multilayer perceptron NN. Both performance metrics (MSE and ADE) improve as either the number of hidden layers or the number of neurons per hidden layer increase, albeit with diminishing returns beyond four hidden layers with 32 neurons each.

Fig. 12 shows the location inferred by the best NNs for each one of the 16 static locations. Standard deviation is from 0.040 m to 0.227 m and we can see that most inferences are close to the actual location. We also note that inference spread appears to be higher in the upper and the right parts of the room. This may be due to higher environmental noise in that areas of the experimental space.

B. Filter optimization and trajectory tracking with multilayer perceptron neural networks

We will show later that neural networks that track the position of the person considering the past behaviour can have better accuracy than those that consider one tuple at a time, as in

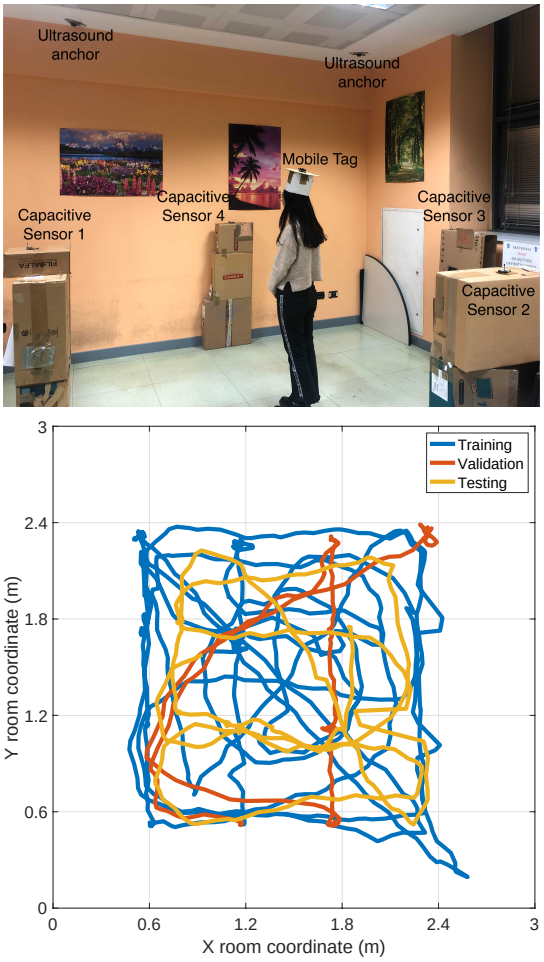


Fig. 9. Virtual room used for movement tracking experiments and person trajectory (split into segments for NN training, validation and testing)

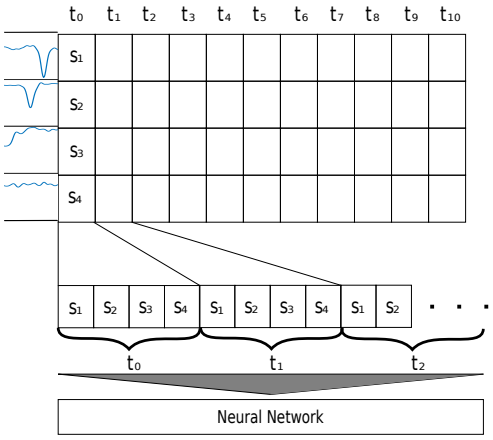


Fig. 10. Sensor data (top-left) is discretized at 3 Hz in four-sample tuples, $S_1, \ldots, S_4$, which are then concatenated in chronological order and input to the neural network with appropriate windowing.

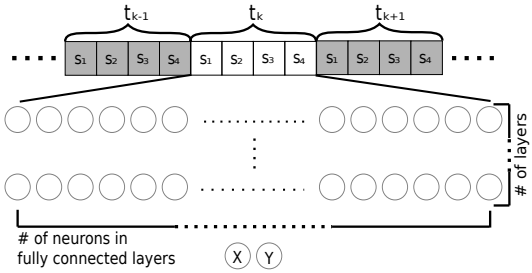


Fig. 11. Network structure and data access for the multilayer perceptron network. The input layer receives sensor data tuples from randomly selected time frames (with labels during training, not shown for readability) and reports the inferred x and y coordinates of the person.

Section IV-A. But we evaluate first the tracking performance of the multilayer perceptron neural networks (see Fig. 11), which do not consider past behaviour, to establish a baseline for the next experiments.

We split the whole trajectory of the person in the room (after applying the median and low-pass filter) in three contiguous segments 60%, 20%, and 20% long for neural network training, validation, and testing, respectively, as shown in Fig. 9. Note that the trajectory segments are different, as they would be in the case of a real-life deployment.

We use the best NN structure that we found in Section IV-A, five hidden layers with 64 neurons each. We next optimize the parameters of the filters (see Fig. 5) by analyzing the NN performance for all combinations of:

- *median filter window*: 50 s, 100 s and 150 s
- *low-pass filter pass-band edge*: 0.1 Hz, 0.2 Hz, 0.3 Hz, 0.4 Hz, and 0.6 Hz
- *low-pass filter stop-band edge*: 0.2 Hz, 0.3 Hz, 0.4 Hz, 0.5 Hz, 0.6 Hz, and 0.7 Hz.

We obtain the best NN performance (MSE 0.111 m^2 and ADE 0.405 m) for a median filter window of 50 s, and low-pass filter pass-band edge of 0.3 Hz and stop-band edge of 0.4 Hz (see the top ten results shown in Table II). The NN performance seems

TABLE II
DESIGN SPACE EXPLORATION RESULTS FOR FILTER OPTIMIZATION FOR THE BEST MULTILAYER PERCEPTRON NEURAL NETWORK MEAN SQUARE ERROR (MSE) AND THE CORRESPONDING AVERAGE DISTANCE ERROR (ADE)

Low-pass filter		**Median filter**	**Error**	
Passband edge (Hz)	Stopband edge (Hz)	Window (s)	MSE (m^2)	ADE (m)
0.3	0.4	50	**0.111**	**0.405**
0.5	0.7	50	0.115	0.418
0.2	0.6	50	0.116	0.432
0.1	0.3	50	0.121	0.429
0.6	0.7	50	0.122	0.428
0.4	0.5	50	0.122	0.431
0.1	0.6	50	0.125	0.432
0.1	0.4	50	0.126	0.445
0.2	0.6	100	0.127	0.449
0.3	0.6	50	0.128	0.447

to be more dependent on the median filter window (windows longer than 50 s generally lead to poorer NN results) than on the parameters of the low-pass filter (almost all present in the NN top ten best results). This can be explained because the amplitude of the drift can be much higher than that of the high-pitch noise (see Fig. 3).

For these parameters, we show in Fig. 13 the NN inference separately for the X and Y coordinates compared with the ground truth. We notice the ragged look of both X and Y

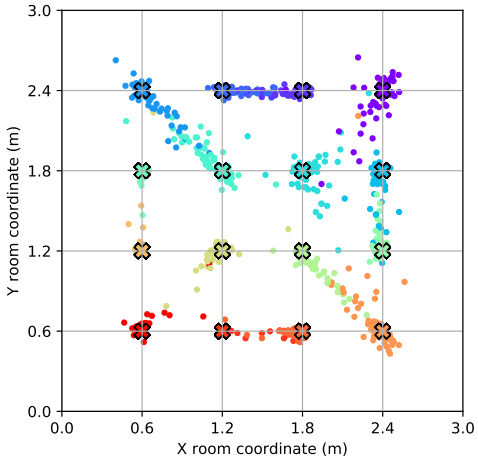
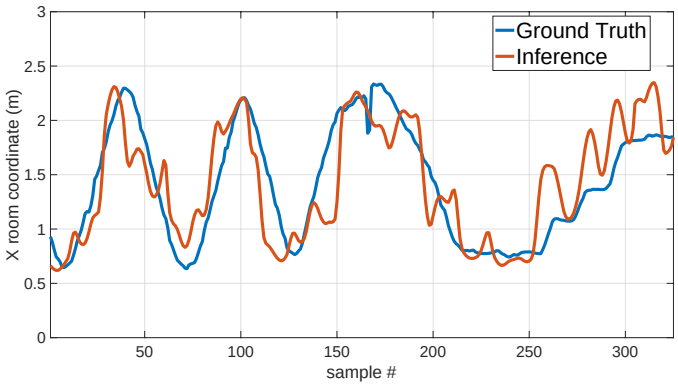
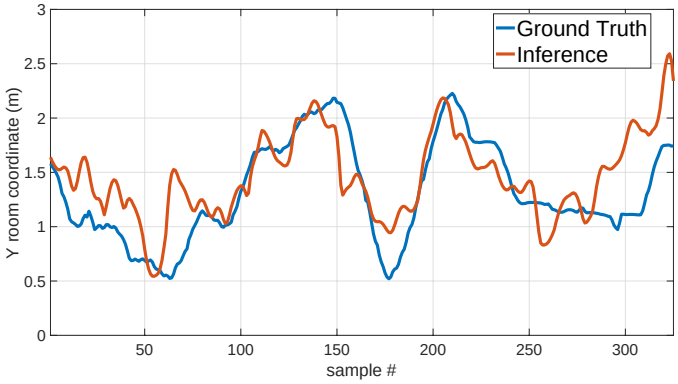


Fig. 12. Inference of 16 static positions (black 'x') using multilayer perceptron neural networks. Inference results (dots) are colored according to their reference positions.



(a)



(b)

Fig. 13. Multilayer perceptron neural network trajectory tracking inference and ground truth for the (a) X axis and (b) Y axis

inference, which seems not to smooth enough the sensor noise. We also notice increasing discrepancies in the latter part of the X and Y tracks (roughly after sample 250).

C. Trajectory tracking with autoregressive feedforward neural networks

This is the first experiment to infer the trajectory using a neural network that considers some aspects of the movement history. We choose autoregressive feedforward neural networks [68], [69] because they are non-recurrent (i.e., feedback-free) sequence-aware models that can be used to infer sequential data as a simpler alternative to recurrent neural networks (see Fig. 14). They are akin to Finite Impulse Response filters in digital signal processing, while recurrent NNs are akin to Infinite Impulse Response filters.

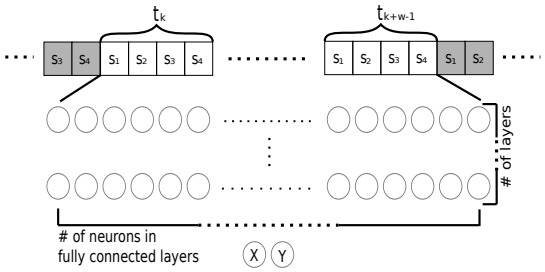


Fig. 14. Network structure and data access for the autoregressive feedforward network. The input layer receives sensor data tuples from sequential time frames that fall into network window (labelled with the x and y coordinates of the central sample during training, not shown for readability) and reports the inferred coordinates of the person.

We provide the NN with capacitive sensor tuples that fall within a temporal window and train the NN to infer the X and Y coordinates corresponding to the *middle* tuple in each window. Hence, the window gives the NN access to both past and future readings in the sensor time series. Using these, during training the NN can refine the best weights for both the past and future sensor readings (around the current position) to better reject the noise and perhaps also to learn the dynamic characteristics of person movements, such as maximum speed, acceleration, movement patterns or direction changes. Of course, the NN may also significantly overfit in a real deployment, hence we put a lot of attention to the "natural looking" aspects of our sample trajectories.

Since the NN needs a window width of samples to produce a valid inference, it will start inferring after seeing a full window of samples at the beginning of the trajectory and stop when the last sample of the trajectory enters the window. But it infers the position corresponding to the middle of the window, hence the inferred trajectory in Fig. 15 starts and stops half a window from trajectory extremes. This behavior is not a problem for our target applications, which are not particularly sensitive to delays of a few seconds. The same applies to the other NNs based on windows that are discussed later.

We implement the best network structure that we found in Section IV-A (see Fig. 14), i.e., five hidden layers with 64 neurons each. The input layer receives all sensor samples within the input window and is fully connected to the first hidden layer, like in Fig. 11.

We explore the performance of this neural network by setting the duration of its input window to 5 s, 10 s, and 15 s. They give the number of input tuples (thus the size of the NN input layer) that are seen by the NN at any moment at 3 Hz sampling rate. For example, for an input window of 5 s we have on the NN input layer

$$5\,\text{s} \times 3\,\text{tuples/s} \times 4\,\text{samples/tuple} = 60\ \text{input neurons}. \quad (3)$$

TABLE III
DESIGN SPACE EXPLORATION RESULTS FOR FILTER OPTIMIZATION FOR THE BEST AUTOREGRESSIVE FEEDFORWARD NEURAL NETWORK MEAN SQUARE ERROR (MSE) AND THE CORRESPONDING AVERAGE DISTANCE ERROR (ADE)

NN Input	**Low-pass filter**		**Median filter**	**Error**	
Window (s)	Passband edge (Hz)	Stopband edge (Hz)	Window (s)	MSE (m^2)	ADE (m)
5	0.1	0.6	50	**0.079**	0.342
10	0.2	0.6	50	0.082	0.347
10	0.3	0.6	50	0.083	**0.340**
10	0.4	0.7	50	0.085	0.365
10	0.6	0.7	100	0.086	0.371
5	0.3	0.6	50	0.092	0.358
10	0.1	0.5	100	0.092	0.380
10	0.4	0.6	50	0.092	0.373
5	0.5	0.6	50	0.093	0.358
10	0.4	0.7	100	0.093	0.395

We also vary the parameters of the low-pass and median filters in the ranges shown in Section IV-B. As can be seen from the top-ten best results shown in Table III, the best MSE is 0.079 m^2 for an ADE of 0.342 m. They show a marked improvement compared to the multilayer perceptron NN results shown in Table II, which is attributable to allowing the NN to infer the position while examining the sensor tuples of a segment of the trajectory instead of just the current tuple. We also note that the optimal length of the trajectory segment (input window) in our DSE seems to be 10 s (the performance for windows of 5 s are equal or marginally better).

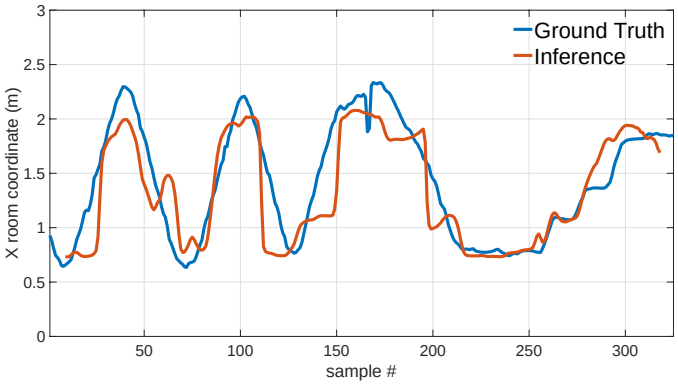
In Fig. 15 we see that the inferences of the X and Y coordinates are smoother and tend to follow closer the ground truth almost everywhere, and especially towards the end (roughly after sample 250) than the multilayer perceptron NN shown in Fig. 13.

D. Trajectory tracking with 1D convolutional neural networks

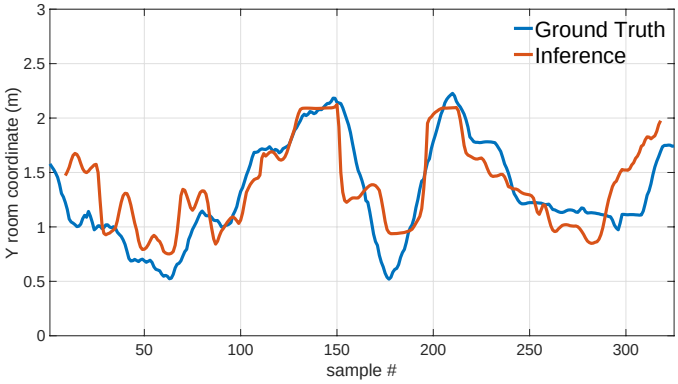
We extend the tests of neural networks that infer the trajectory based on movement history using 1D Convolutional NNs (CNNs) [70]. They are known to be effective for deriving meaningful features from fixed-length segments (input windows) of data sequences. Typical applications include sequences of sensor data (e.g., accelerometer, audio), which are similar to the ones that we have in this application.

We use 1D CNNs with the structure shown in Fig. 16. We set the window size to 5 s, which we determined to be among the best options in Section IV-C. We scan this window with several 1D convolution filters with kernels of the same size, which make one convolutional layer. Convolutional processing in our NN uses several such layers, followed by a pooling layer, and a fixed size MLP network (of two layers with 64 neurons each), before the output layer.

During the DSE, we change the number of convolutional layers (in groups of two convolutional layers followed by a 50% dropout), the convolution kernel size, and the number of filters. In each experiment, we keep constant for each network architecture the kernel size and the number of filters per layer. We use the LPF and MF parameters optimized in Section IV-B.



(a)



(b)

Fig. 15. Autoregressive feedforward neural network trajectory tracking inference and ground truth for the (a) X axis and (b) Y axis

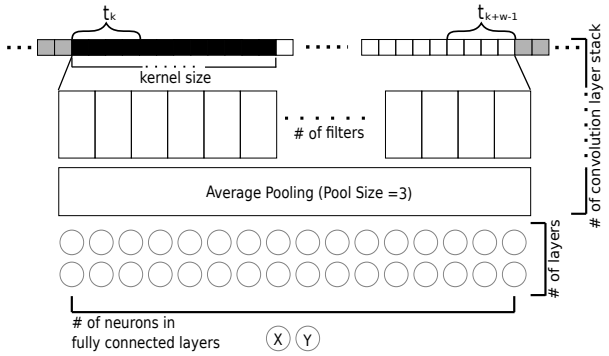


Fig. 16. Network structure and data access for the convolutional network. Each filter processes the data tuples within the kernel width, then the kernel slides one tuple to the right until the end of the input window. When done, the window moves one tuple to the right and the kernel restarts its scanning. For network training, each window is labelled (not shown for readability) with the person coordinates corresponding to the middle tuple in the window.

We train and test ten times the neural network for each combination of hyperparameters. We show in Table IV the results of the best network for each hyperparameter combination. We note that most of the best configurations have four convolutional layers, while the network performance tends to degrade for either smaller or larger number of layers. Also, for a given number of convolutional layers, the network configurations with fewer filters appear to have the best performance. The best overall network configuration has the least number of

TABLE IV
DESIGN SPACE EXPLORATION RESULTS FOR 1D CONVOLUTIONAL NEURAL NETWORKS MEAN SQUARE ERROR (MSE) AND THE CORRESPONDING AVERAGE DISTANCE ERROR (ADE) FOR DIFFERENT CONVOLUTIONAL KERNEL SIZES, NUMBER OF CONVOLUTIONAL LAYERS, AND FILTERS

	Number of filters							
	8		**16**		**32**		**64**	
Kernel size	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)
	Two 1D convolutional layers							
3	0.086	0.351	0.092	0.370	0.090	0.364	0.085	0.357
5	0.084	0.346	0.094	0.373	0.093	0.369	0.092	0.371
7	0.080	0.347	0.093	0.359	**0.078**	**0.343**	0.097	0.384
	Four 1D convolutional layers							
3	***0.063***	***0.307***	0.089	0.372	0.090	0.366	0.085	0.351
5	0.081	0.350	0.088	0.358	0.088	0.371	0.091	0.365
7	0.093	0.369	0.090	0.351	0.091	0.388	0.078	0.342
	Six 1D convolutional layers							
3	**0.078**	**0.328**	0.101	0.377	0.087	0.364	0.092	0.385
5	0.086	0.365	0.090	0.372	0.099	0.379	0.093	0.378
7	0.098	0.387	0.107	0.397	0.092	0.366	0.092	0.373

filters (eight) and the smallest convolutional kernel size (three).

We show in Fig. 17 the inference of the X and Y coordinates of the best networks for each number of convolutional layers, two, four, and six (highlighted in Table IV). We note how the six-layer network matches well the last part of the X trace, but less so in the middle. The two-layer network appears to have the highest ripples, while the four-layer seems to match best almost the whole trace, except for the last part, roughly after sample 250. On the Y trace, the four-layers network appears to stay closest to the ground truth overall.

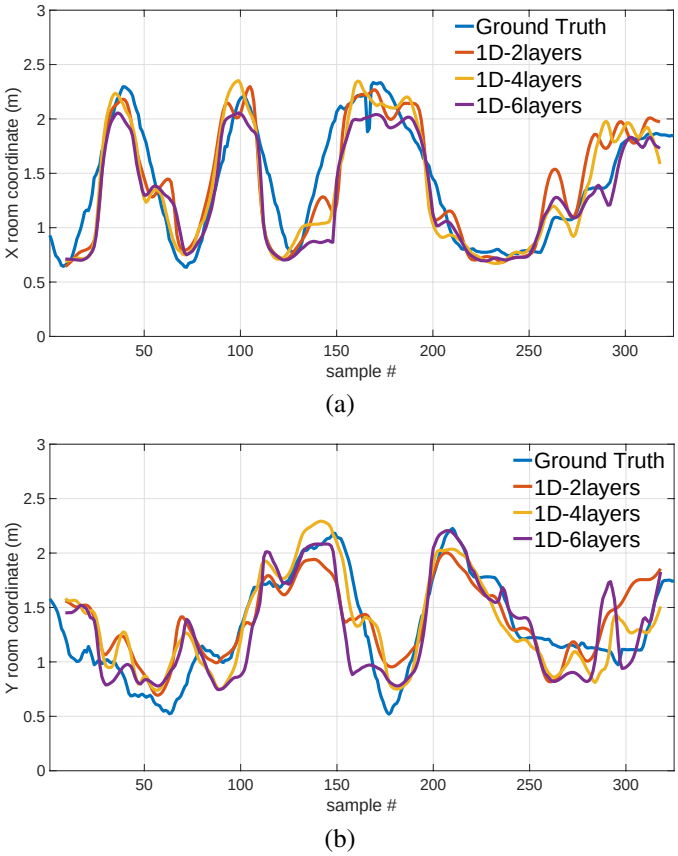


Fig. 17. Best 1D convolutional neural network trajectory tracking inferences and the ground truth for the (a) X axis and (b) Y axis for different number of convolutional layers.

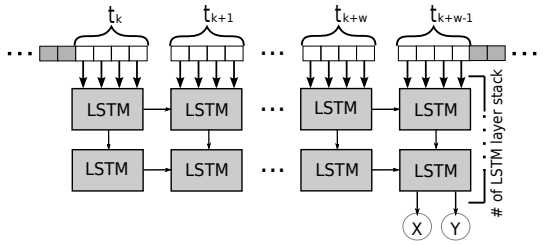


Fig. 18. Network structure and data access for the Long-Short Term Memory (LSTM) network, in which the LSTM cells in the input layer process the data tuples from the input window. Each window is labelled for training with the person coordinates corresponding to the middle tuple in the window (not shown for readability).

E. Trajectory tracking with long-short term memory networks

After testing feedforward neural networks (multilayer perceptron, autoregressive, and 1D convolutional types, discussed in previous sections), we explore recurrent neural networks. Of these, the LSTMs [71] are widely used to extract features from data sequences, such as speech or handwriting.

Fig. 18 shows the structure of our LSTM network [71]. It is a typical LSTM, in which the cells transfer the state horizontally and receive inputs either from the sensors (the first layer) or from the outputs of the previous layer.

To explore the LSTM network performance, we vary the hyperparameters known to have most influence [72], namely the number of neurons in the hidden layers and the number of hidden layers. During the DSE, we run ten times the LSTM training (with random initialization) and testing for each combination of hyperparameters, and report the best results in the top half of Table V. We note that the network performance, MSE, does not change much with the number of hidden layers or their number of neurons, and especially so for smaller numbers of neurons (8 or 16 per each hidden layer). An LSTM with one hidden layer with 16 neurons appears to perform best.

Bidirectional LSTMs (BD-LSTMs) [73] can improve the LSTM performance leveraging future samples in their inference (e.g., handwriting recognition can improve by looking also at letters after the current one). We test the BD-LSTM performance for our problem using the same DSE parameters and report the results in the second half of Table V. Performance seems to be more sensitive to hyperparameters, and especially to the number of hidden layers. The best appears again the configuration with one hidden layer with 16 neurons.

We show in Fig. 19 the inference of the X and Y coordinates of the best LSTM and BD-LSTM network (highlighted in Table V). Generally, we note very little differences between them. They both miss the first two peaks of the X coordinate and the central peak of the Y coordinate, as well as the beginning (up to sample 70 or so) and end (from around sample 250) of the Y coordinate. In some occasions BD-LSTM appears to

TABLE V
DESIGN SPACE EXPLORATION RESULTS FOR UNI- AND BI-DIRECTIONAL LONG-SHORT TERM MEMORY NETWORKS MEAN SQUARE ERROR (MSE) AND THE CORRESPONDING AVERAGE DISTANCE ERROR (ADE) WHILE VARYING THE NUMBER OF HIDDEN LAYERS AND NEURONS PER LAYER

	Internal units of LSTM layer							
	8		**16**		**32**		**64**	
Layers	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)	MSE (m^2)	ADE (m)
	Unidirectional long-short time memory neural network							
1	0.085	0.339	**0.080**	**0.325**	0.085	0.333	0.089	0.352
2	0.083	0.345	0.082	0.335	0.088	0.347	0.091	0.357
3	0.084	0.350	0.083	0.342	0.096	0.366	0.087	0.355
	Bidirectional long-short time memory neural network							
1	0.083	0.342	**0.079**	**0.326**	0.091	0.341	0.095	0.362
2	0.095	0.362	0.092	0.352	0.099	0.378	0.102	0.376
3	0.080	0.339	0.099	0.372	0.110	0.408	0.107	0.401

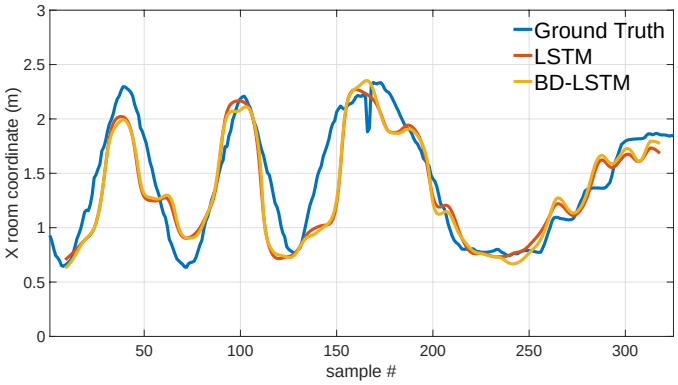
come closer than LSTM to the ground truth, such as around sample 50 on the Y trace and sample 160 on the X trace.

We note that the trajectories inferred by the LSTM network are the smoothest among all networks that we explored. They seem to reflect more closely the movement dynamics of a person, albeit with slightly higher error than the best inference (LSTM 0.079 m^2 MSE and 0.326 m ADE versus 1D convolutional 0.063 m^2 MSE and 0.307 m ADE). We intend to investigate in future work if LSTM networks indeed capture better movement dynamics and if their performance improves using less noisy readings from capacitive sensors that are more robust to environmental noise.

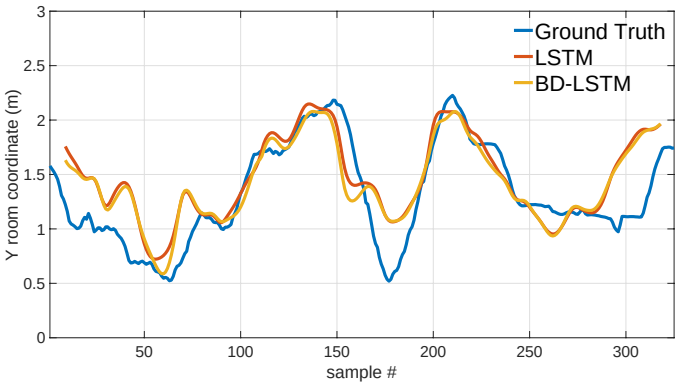
V. DISCUSSION

In Fig. 20, we can compare visually the quality of the inferred X and Y traces of the room trajectory of the best configurations of all neural network types. Their performance metrics are shown in Table I and in Table III to Table VI. In Table VI we report the correlation between the network inferences and the ground truth, as a measure of the inference replication of the actual person trajectory regardless of systematic distance errors. We also report the RMS of the first and second derivative of the inferences, as inverse measures of the speed and acceleration smoothness of the inferences, respectively (lower numbers are associated with better smoothness) [74, p. 62]. Note that the figures for the smoothness of the ground truth itself are rather high, mostly because the localization data collected from the ultrasound-based reference system has some centimeter-level jitter which we did not filter, but which seems to be filtered well by all neural networks.

The 1D convolutional network appears to follow best the X and Y components of the reference trajectory. The four-layer 1D CNN inference has the lowest MSE and the best correlation with the ground truth. The 1D CNN inference is also among the smoothest, closely matching the dynamics of the actual movement of the person, as can be seen in Fig. 20. In fact, the RMS of the first and second derivatives of the inferred location are bested only by networks in the LSTM class, as shown in Table VI.



(a)



(b)

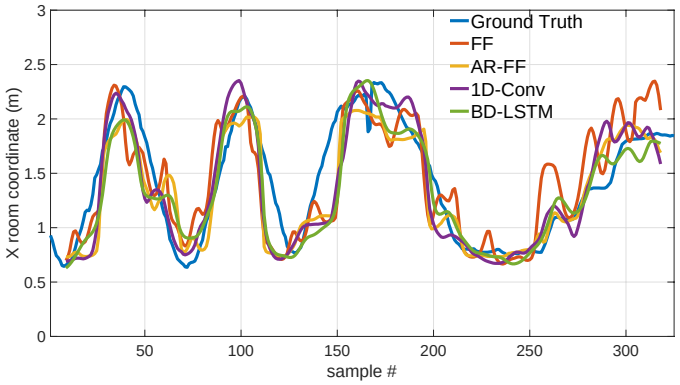
Fig. 19. Best Long-Short Term Memory (LSTM) and Bidirectional LSTM (BD-LSTM) neural network trajectory tracking inferences and the ground truth for the (a) X axis and (b) Y axis

TABLE VI
CHARACTERIZATION OF MOVEMENT INFERENCE QUALITY IN TERMS OF CORRELATION WITH THE GROUND TRUTH, AND SPEED AND ACCELERATION SMOOTHNESS CALCULATED AS THE ROOT MEAN SQUARE (RMS) OF THE FIRST AND SECOND DERIVATIVES, RESPECTIVELY.

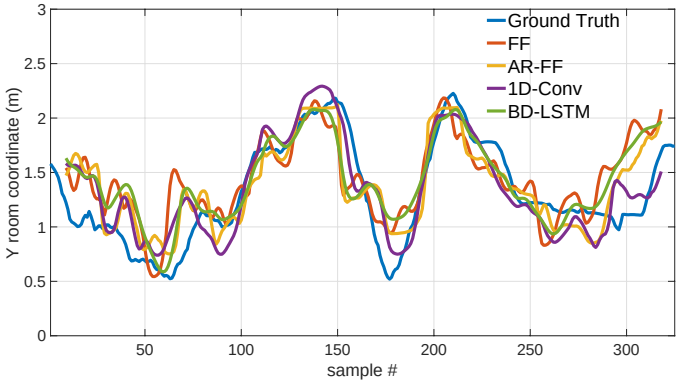
	Inference characterization		
Neural network type	**Ground truth correlation** (%)	**RMS first derivative** (m/s)	**RMS second derivative** (m/s^2)
Multilayer perceptron	77.1	0.215	0.370
Autoregressive	83.1	0.211	0.475
1D CNN (2 layers)	83.3	0.157	0.172
1D CNN (4 layers)	**87.5**	0.162	0.187
1D CNN (6 layers)	84.5	0.176	0.259
LSTM	85.0	**0.129**	**0.106**
Bidirectional LSTM	84.0	0.133	0.129
Ground truth		*0.143*	*0.333*

The inference of the recurrent networks, LSTMs, seems to be the smoothest, closely matching the actual person movement dynamics, as shown by the low RMS of first and second derivatives of the inferred location in Table I. LSTM ground truth correlation is also very good, but its MSE is higher because it does not follow well all parts of the person trajectory. We intend to investigate in future work if the discrepancy is due to the low signal-to-noise ratio of the sensor data.

Fig. 20 shows that the multilayer perceptron network infers



(a)



(b)

Fig. 20. Ground truth and best neural network trajectory tracking inferences of the (a) X axis and (b) Y axis for different types of neural networks: multilayer perceptron (FF), autoregressive feedforward (AR-FF), 1D convolutional (1D-Conv), and bidirectional long-short term memory (BD-LSTM)

a trajectory with the largest oscillations. Hence, it has the lowest correlation with the ground truth and distinctly high RMSs of the inference first and second derivatives. This can be because the network has no means to understand the physical movement dynamics because it is trained with separate points of the trajectory, which carry no dynamic information. But even in autoregressive configuration, where it is trained using segments of trajectory, the performance of the multilayer perceptron network does not improve, most likely because it lacks filtering capabilities, unlike the convolution filters of the 1D CNN or the intrinsic recurrent memory of the LSTMs. However, the multilayer perceptron networks can infer with good accuracy static positions, as shown in Table I and Fig. 12.

Besides the noise in the sensor data, we should note that the accuracy of neural network inferences was affected both by errors of the reference system and by differences in the posture of the person. As discussed in Section III, we measured the former to average at $\pm 3.9\,\mathrm{cm}$ with peaks of $\pm 6.4\,\mathrm{cm}$ in our experimental conditions. The latter can depend on head inclination (the person was wearing the mobile tag on the head) or body rotation, which can change the distance between the part of the body that is closest to the sensors for the same position of the person in the room. In fact, the human body can have complex postures and irregular shape, making it difficult to accurately define its position. Hence part of the reported neural network errors can be attributed to this intractable application domain-specific uncertainty.

TABLE VII
NUMBER OF PARAMETERS, FLOATING-POINT OPERATIONS (FLOPS), AND INFERENCE ERROR [MEAN SQUARE (MSE) AND AVERAGE DISTANCE (ADE)] FOR THE BEST NEURAL NETWORKS OF EACH TYPE

Neural network	**Parameters**	**FLOPs**	**MSE** (m^2)	**ADE** (m)
Multilayer perceptron	17090	34180	0.111	0.405
Autoregressive	20674	40382	0.079	0.342
1D CNN (2 layers) 32 filters, kernel size 7	14530	22318	0.078	0.343
1D CNN (4 layers) 8 filters, kernel size 3	7618	34078	**0.063**	**0.307**
1D CNN (6 layers) 8 filters, kernel size 3	8018	45838	0.078	0.328
LSTM	**1378**	**16800**	0.080	0.325
Bidirectional LSTM	2754	33600	0.079	0.326

In Table VII, we show the number of parameters, processing effort (estimated as the number of floating-point operations), and the inference accuracy for the best neural network of each type. The four layer 1D convolutional has the best accuracy, but needs more parameters and higher processing effort. The single layer LSTM provides very good localization estimation, the smoothest movement tracking, and also requires the fewest parameters and lowest processing effort (both important for embedded applications).

VI. CONCLUSION

We tested the inference accuracy of several neural network types, both feedforward and recurrent, while tracing the location and movement of a person using data from four capacitive sensors placed in the middle of the "walls" of a $3\,\mathrm{m}\times 3\,\mathrm{m}$ empty laboratory area. Sensor sensitivity was limited by noise level, and their stability was also affected by a slow but significant drift. While we used filters to reduce both drift and high-pitch noise, we were especially interested in how much the remaining noise affects the accuracy of the inference of person location and trajectory for various types of neural networks.

The best inference, evaluated both as mean square error and as smoothness and closeness to the actual person movement in the room, was obtained by neural networks trained on trajectory segments, processing either windows (feedforward autoregressive and 1D convolutional) or sequences (long-short term memory). The latter kind seems to capture best the movement dynamics, while the 1D convolutional network has the smallest error. Networks that consider trajectory points in isolation perform well with data collected for static positions, but have the worst trajectory inference error and do not seem to capture the movement dynamics.

It is hard to define accurately the position of the human body, especially while moving, due to its complex shapes and postures. This can explain part of the inference errors, in addition to the limited accuracy of the reference system ($\pm 3.9\,\mathrm{cm}$ average and $\pm 6.4\,\mathrm{cm}$ max in our setting).

We note that even with these noisy sensors, the best average localization error of $0.307\,\mathrm{m}$ is suitable for our main target application, namely assisted living of elderly persons.

The inference characteristics of several very distinct NN architectures, in basic configurations and processing noisy sensor data, provide a broad performance baseline that can help designing more application-specific NN architectures, which may use and tune specialized DSP blocks to better recognize the movement dynamics and reject environmental noise.

REFERENCES

[1] U. Nations, *World Population Ageing 2019 Highlights*, 2019.

[2] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Comm. Surveys & Tutorials*, 2019.

[3] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *13th USENIX Symposium on Networked Systems Design and Implementation*, 2016, pp. 165–178.

[4] E. Soltanaghaei, A. Kalyanaraman, and K. Whitehouse, "Multipath triangulation: Decimeter-level WiFi localization and orientation with a single unaided receiver," in *Proc. of the 16th int. conf. on mobile systems, applications, and services*, 2018, pp. 376–388.

[5] T. Ishihara, K. M. Kitani, C. Asakawa, and M. Hirose, "Inference machines for supervised Bluetooth localization," in *2017 IEEE Int. Conf. on Acoustics, Speech and Signal Proc.* IEEE, 2017, pp. 5950–5954.

[6] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards," *Computer communications*, vol. 30, no. 7, pp. 1655–1695, 2007.

[7] M. El-Absi, A. A. Abbas, A. Abuelhaija, F. Zheng, K. Solbach, and T. Kaiser, "High-accuracy indoor localization based on chipless RFID systems at THz band," *IEEE Access*, vol. 6, pp. 54 355–54 368, 2018.

[8] Y. Xu, Y. S. Shmaliy, Y. Li, and X. Chen, "UWB-based indoor human localization with time-delayed data using EFIR filtering," *IEEE Access*, vol. 5, pp. 16 676–16 683, 2017.

[9] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," in *Proc. of the 20th int. conf. on Mobile computing and networking*, 2014, pp. 447–458.

[10] W. Huang, Y. Xiong, X.-Y. Li, H. Lin, X. Mao, P. Yang, Y. Liu, and X. Wang, "Swadloon: Direction finding and indoor localization using acoustic signal by shaking smartphones," *IEEE Trans. on Mobile Computing*, vol. 14, no. 10, pp. 2145–2157, 2014.

[11] F. Ijaz, H. K. Yang, A. W. Ahmad, and C. Lee, "Indoor positioning: A review of indoor ultrasonic positioning systems," in *2013 15th Int. Conf. on Advanced Communications Technology*. IEEE, 2013, pp. 1146–1150.

[12] T. Kivimäki, T. Vuorela, P. Peltola, and J. Vanhala, "A review on device-free passive indoor positioning methods," *Int. J. of Smart Home*, vol. 8, no. 1, pp. 71–94, 2014.

[13] N. Pirzada, M. Y. Nayan, F. Hassan, and M. A. Khan, "Device-free localization technique for indoor detection and tracking of human body: A survey," *Procedia-Social and Behavioral Sciences*, vol. 129, no. 422-429, p. 2nd, 2014.

[14] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing Technology for Human Activity Recognition: a Comprehensive Survey," *IEEE Access*, 2020.

[15] L. Middleton, A. A. Buss, A. Bazin, and M. S. Nixon, "A floor sensor system for gait recognition," in *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*. IEEE, 2005, pp. 171–176.

[16] J. Paradiso, C. Abler, K.-y. Hsiao, and M. Reynolds, "The magic carpet: physical sensing for immersive environments," in *CHI'97 Extended Abstracts on Human Factors in Computing Systems*, 1997, pp. 277–278.

[17] X. Zhou, Q. Hao, and H. Fei, "1-bit walker recognition with distributed binary pyroelectric sensors," in *2010 IEEE Conf. on Multisensor Fusion and Integration*. IEEE, 2010, pp. 168–173.

[18] J.-M. Valin, F. Michaud, B. Hadjou, and J. Rouat, "Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach," in *IEEE Int. Conf. on Robotics and Automation*, vol. 1. IEEE, 2004, pp. 1033–1038.

[19] S. Holm and C.-I. C. Nilsen, "Robust ultrasonic indoor positioning using transmitter arrays," in *2010 Int. Conf. on Indoor Positioning and Indoor Navigation*. IEEE, 2010, pp. 1–5.

[20] Y. Nishida, T. Hori, S.-i. Murakami, and H. Mizoguchi, "Minimally privacy-violative system for locating human by ultrasonic radar embedded on ceiling," in *2004 IEEE Int. Conf. on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, vol. 2. IEEE, 2004, pp. 1549–1554.

[21] S. N. Patel, M. S. Reynolds, and G. D. Abowd, "Detecting human movement by differential air pressure sensing in HVAC system ductwork: An exploration in infrastructure mediated sensing," in *Int. Conf. on Pervasive Computing*. Springer, 2008, pp. 1–18.

[22] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the flick of a switch: Detecting and classifying unique electrical events on the residential power line," in *Int. Conf. on Ubiquitous Computing*. Springer, 2007, pp. 271–288.

[23] J. Fogarty, C. Au, and S. E. Hudson, "Sensing from the basement: a feasibility study of unobtrusive and low-cost home activity recognition," in *Proceedings of the 19th ACM symposium on User interface software and technology*, 2006, pp. 91–100.

[24] P. Hu, L. Li, C. Peng, G. Shen, and F. Zhao, "Pharos: Enable physical analytics through visible light based indoor localization," in *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, 2013, pp. 1–7.

[25] I. B. Arief-Ang, F. D. Salim, and M. Hamilton, "CD-HOC: indoor human occupancy counting using carbon dioxide sensor data," *arXiv preprint arXiv:1706.05286*, 2017.

[26] M. Zakrzewski, H. Raittinen, and J. Vanhala, "Comparison of center estimation algorithms for heart and respiration monitoring with microwave Doppler radar," *IEEE Sensors J.*, vol. 12, no. 3, pp. 627–634, 2011.

[27] D. Hauschildt and N. Kirchhof, "Improving indoor position estimation by combining active TDOA ultrasound and passive thermal infrared localization," in *2011 8th Workshop on Positioning, Navigation and Communication*. IEEE, 2011, pp. 94–99.

[28] X. Tang and S. Mandal, "Indoor occupancy awareness and localization using passive electric field sensing," *IEEE Trans. on Instrumentation and Measurement*, vol. 68, no. 11, pp. 4535–4549, 2019.

[29] A. Braun, H. Heggen, and R. Wichert, "CapFloor–a flexible capacitive indoor localization system," in *Int. Competition on Evaluating AAL Systems through Competitive Benchmarking*. Springer, 2011, pp. 26–35.

[30] S. Nakamura, S. Ajisaka, K. Takiguchi, A. Hirose, and H. Hashimoto, "Electric-field resonance coupling between human and transmitter for human position estimation system," in *ICCAS 2010*. IEEE, 2010, pp. 109–114.

[31] A. Ropponen, M. Linnavuo, R. Sepponen *et al.*, "LF indoor location and identification system," *Int. J. on Smart Sensing and Intelligent Systems*, vol. 2, no. 1, pp. 94–117, 2009.

[32] J. Smith, T. White, C. Dodge, J. Paradiso, N. Gershenfeld, and D. Allport, "Electric field sensing for graphical interfaces," *IEEE Computer Graphics and Applications*, vol. 18, no. 3, pp. 54–60, 1998.

[33] R. Wimmer, "Capacitive sensors for whole body interaction," in *Whole Body Interaction*. Springer, 2011, pp. 121–133.

[34] A. Braun, R. Wichert, A. Kuijper, and D. W. Fellner, "Capacitive proximity sensing in smart environments," *J. of Ambient Intelligence and Smart Environments*, vol. 7, no. 4, pp. 483–510, 2015.

[35] T. Grosse-Puppendahl, C. Holz, G. Cohn, R. Wimmer, O. Bechtold, S. Hodges, M. S. Reynolds, and J. R. Smith, "Finding common ground: A survey of capacitive sensing in human-computer interaction," in *Proc. of the 2017 CHI Conf. on Human Factors in Computing Systems*. ACM, 2017, pp. 3293–3315.

[36] J. Iqbal, A. Arif, O. B. Tariq, M. T. Lazarescu, and L. Lavagno, "A contactless sensor for human body identification using RF absorption signatures," in *2017 IEEE Sensors Applications Symposium (SAS)*. IEEE, 2017, pp. 1–6.

[37] J. Iqbal, M. T. Lazarescu, O. B. Tariq, A. Arif, and L. Lavagno, "Capacitive sensor for tagless remote human identification using body frequency absorption signatures," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 4, pp. 789–797, Apr. 2018.

[38] A. Arshad, S. Khan, A. Z. Alam, R. Tasnim, T. S. Gunawan, R. Ahmad, and C. Nataraj, "An activity monitoring system for senior citizens living independently using capacitive sensing technique," in *2016 IEEE Int. Instrumentation and Meas. Tech. Conf. Proc.* IEEE, 2016, pp. 1–6.

[39] A. Ramezani Akhmareh, M. Lazarescu, O. Bin Tariq, and L. Lavagno, "A tagless indoor localization system based on capacitive sensing technology," *Sensors*, vol. 16, no. 9, p. 1448, 2016.

[40] J. Iqbal, M. T. Lazarescu, O. B. Tariq, and L. Lavagno, "Long range, high sensitivity, low noise capacitive sensor for tagless indoor human localization," in *2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, Jun. 2017, pp. 189–194.

[41] J. Iqbal, M. T. Lazarescu, A. Arif, and L. Lavagno, "High sensitivity, low noise front-end for long range capacitive sensors for tagless indoor human localization," in *2017 IEEE 3rd International Forum on Research and Technologies for Society and Industry (RTSI)*, Sep. 2017, pp. 1–6.

[42] R. Wimmer, M. Kranz, S. Boring, and A. Schmidt, "A capacitive sensing toolkit for pervasive activity detection and recognition," in *Fifth IEEE Int. Conf. on Pervasive Computing and Comm.* IEEE, 2007, pp. 171–180.

[43] A. Shareef, Y. Zhu, and M. Musavi, "Localization using neural networks in wireless sensor networks," in *Proc. of the 1st int. conf. on Mobile Wireless Middleware, Operating Systems, and App.* ICST, 2008, p. 4.

[44] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, 2017.
[45] M. Djakow, A. Braun, and A. Marinc, "MoviBed-sleep analysis using capacitive sensors," in *Int. Conf. on Universal Access in Human-Computer Interaction*. Springer, 2014, pp. 171–181.
[46] T. A. Große-Puppendahl, A. Marinc, and A. Braun, "Classification of user postures with capacitive proximity sensors in AAL-environments," in *Int. Joint Conf. on Ambient Intelligence*. Springer, 2011, pp. 314–323.
[47] M. Haescher, D. J. Matthies, G. Bieber, and B. Urban, "CapWalk: a capacitive recognition of walking-based activities as a wearable assistive technology," in *Proc. of the 8th ACM Int. Conf. on Pervasive Technologies Related to Assistive Environments*, 2015, pp. 1–8.
[48] M. Valtonen, J. Maentausta, and J. Vanhala, "Tiletrack: Capacitive human tracking using floor tiles," in *2009 IEEE Int. Conf. on Pervasive Computing and Communications*. IEEE, 2009, pp. 1–10.
[49] M. Valtonen, L. Kaila, J. Mäentausta, and J. Vanhala, "Unobtrusive human height and posture recognition with a capacitive sensor," *J. of Ambient Intelligence and Smart Env.*, vol. 3, no. 4, pp. 305–332, 2011.
[50] M. Valtonen, T. Kivimäki, and J. Vanhala, "Capacitive 3D user tracking with a mobile demonstration platform," in *Proc. of the 16th Int. Academic MindTrek Conf.*, 2012, pp. 61–63.
[51] R. Henry, L. Matti, and S. Raimo, "Human tracking using near field imaging," in *2008 Second Int. Conf. on Pervasive Computing Technologies for Healthcare*. IEEE, 2008, pp. 148–151.
[52] T. Kivimäki, T. Vuorela, M. Valtonen, and J. Vanhala, "Reliability of the TileTrack capacitive user tracking system in smart home environment," in *ICT 2013*. IEEE, 2013, pp. 1–5.
[53] N.-W. Gong, S. Hodges, and J. A. Paradiso, "Leveraging conductive inkjet technology to build a scalable and versatile surface for ubiquitous sensing," in *Proc. of the 13th int. conf. on Ubiquitous computing*, 2011, pp. 45–54.
[54] C. Lauterbach and A. Steinhage, "Sensfloor®-a large-area sensor system based on printed textiles printed electronics," in *Proc. AAL-Kon.*, 2009.
[55] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.
[56] X. Wang, Z. Yu, and S. Mao, "Deepml: Deep lstm for indoor localization with smartphone magnetic and light sensors," in *2018 IEEE Int. Conf. on Communications*. IEEE, 2018, pp. 1–6.
[57] Z. Chen, H. Zou, J. Yang, H. Jiang, and L. Xie, "WiFi Fingerprinting Indoor Localization Using Local Feature-Based Deep LSTM," *IEEE Systems J.*, 2019.
[58] O. B. Tariq, M. T. Lazarescu, J. Iqbal, and L. Lavagno, "Performance of machine learning classifiers for indoor person localization with capacitive sensors," *IEEE Access*, vol. 5, pp. 12 913–12 926, 2017.
[59] O. B. Tariq, M. T. Lazarescu, and L. Lavagno, "Neural network-based indoor tag-less localization using capacitive sensors," in *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM Int. Symposium on Wearable Computers*. ACM, 2019, pp. 9–12.
[60] "Starter Set HW v4.9." [Online]. Available: https://marvelmind.com/product/starter-set-hw-v4-9/
[61] L. K. Baxter, "Capacitive sensors," *Design and Applications*, 1997.
[62] H. Nishiyama and M. Nakamura, "Capacitance of a strip capacitor," *IEEE Trans. on Components, Hybrids, and Manufacturing Technology*, vol. 13, no. 2, pp. 417–423, 1990.
[63] M. A. et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: https://www.tensorflow.org/
[64] F. Chollet *et al.*, "Keras," 2015. [Online]. Available: https://github.com/fchollet/keras
[65] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. of the 27th int. conf. on machine learning*, 2010, pp. 807–814.
[66] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[67] F. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *IET Conf. Proc.*, pp. 850–855(5), January 1999.
[68] T. J. Sejnowski and C. R. Rosenberg, "Parallel networks that learn to pronounce english text," *Complex sys.*, vol. 1, no. 1, pp. 145–168, 1987.
[69] C. Freeman, R. Dony, and S. Areibi, "Audio environment classication for hearing aids using artificial neural networks with windowed input," in *2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing*. IEEE, 2007, pp. 183–188.
[70] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *arXiv preprint arXiv:1905.03554*, 2019.
[71] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[72] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.
[73] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
[74] J. Ramsay, G. Hooker, and S. Graves, *Functional data analysis with R and MATLAB*. New York, NY: Springer, 2009.

Osama Bin Tariq received the MS degree in Electronic engineering with specialization in embedded systems from the Politecnico di Torino, Italy where currently he is pursuing the Ph.D. degree with the department of Electronic and Telecommunications engineering. His research interests include artificial intelligence and machine learning applications, indoor localization, and high-level synthesis.

Mihai Teodor Lazarescu received the Ph.D. degree in Electronics and Communications from Politecnico di Torino (Italy) in 1998, where he serves now as Assistant Professor. He was Senior Engineer at Cadence Design Systems and founded several startups. He co-authored more than 60 scientific publications, four books, and international patents. His research interests include design tools for reusable WSN platforms, sensing, indoor localization, and data processing for IoT, low power embedded design, high-level HW/SW co-design, and high-level synthesis.

Luciano Lavagno received the Ph.D. degree in EECS from the University of California at Berkeley, Berkeley, CA, USA, in 1992. He was the architect of the POLIS HW/SW co-design tool. From 2003 to 2014, he was an Architect of the Cadence CtoSilicon high-level synthesis tool. Since 1993, he has been a Professor with the Politecnico di Torino, Italy. He co-authored four books and more than 200 scientific papers. His research interests include synthesis of asynchronous circuits, HW/SW co-design, high-level synthesis, and design tools for wireless sensor networks.