**Engineering Applications of Neural Networks manuscript No.**
(will be inserted by the editor)

# Indoor Human Activity Recognition Using High-Dimensional Sensors and Deep Neural Networks

**Baptist Vandersmissen[1] · Nicolas Knudde[2] · Azarakhsh Jalalvand[1] · Ivo Couckuyt[2] · Tom Dhaene[2] · Wesley De Neve[1,3]**

Received: date / Accepted: date

**Abstract** Many smart home applications rely on indoor human activity recognition. This challenge is currently primarily tackled by employing video camera sensors. However, the use of such sensors is characterized by fundamental technical deficiencies in an indoor environment, often also resulting in a breach of privacy. In contrast, a radar sensor resolves most of these flaws and maintains privacy in particular. In this paper, we investigate a novel approach towards automatic indoor human activity recognition, feeding high-dimensional radar and video camera sensor data into several deep neural networks. Furthermore, we explore the efficacy of sensor fusion to provide a solution in less than ideal circumstances. We validate our approach on two newly constructed and published data sets that consist of 2347 and 1505 samples distributed over six different types of gestures and events, respectively. From our analysis, we can conclude that, when considering a radar sensor, it is optimal to make use of a three-dimensional convolutional neural network that takes as input sequential range-Doppler maps. This model achieves 12.22% and 2.97% error rate on the gestures and the events data set, respectively. A pre-trained residual network is employed to deal with the video camera sensor data and obtains 1.67% and 3.00% error rate on the same data sets. We show that there exists a clear benefit in combining both sensors to enable activity recognition in the case of less than ideal circumstances.

**Keywords** activity recognition · deep neural networks · high-dimensional sensors · sensor fusion

## 1 Introduction

Indoor activity recognition is an essential feature for future smart homes, with applications ranging from advanced security systems to health monitoring tools. A video camera is a powerful sensor when it comes to identifying humans or recognizing actions [24,37]. However, despite the ubiquitous availability of this sensor, it is characterized by a number of fundamental deficiencies in an indoor environment, like the inability to function properly when the view is blocked or when lighting conditions are unfavorable. Furthermore, the indoor use of video cameras results in a breach of privacy. In contrast, a radar device preserves visual privacy, while being unaffected by poor lighting conditions. Moreover, it can deal with obstructing elements and it even allows for through-the-wall sensing [38].

While a camera device passively operates by measuring light streams captured by a lens, a radar device transmits an electromagnetic signal over a certain line of sight (LOS). Thanks to the well-known Doppler effect, essential information such as velocity and range can be extracted from the reflection of every target in this LOS. In addition, separately moving parts are characterized by their own Doppler signal. The superposition of all these Doppler signals can be summarized by a so-called micro-Doppler (MD) signature [5].

In this paper, we present deep machine learning approaches for indoor human activity recognition, using a frequency-modulated continuous-wave (FMCW) radar

Baptist Vandersmissen
E-mail: baptist.vandersmissen@ugent.be
[1] Department of Electronics and Information Systems
Ghent University–imec
Technologiepark-Zwijnaarde 122, 9052 Gent, Belgium
[2] Department of Information Technology
Ghent University–imec
Technologiepark-Zwijnaarde 126, 9052 Gent, Belgium
[3] Center for Biotech Data Science
Ghent University Global Campus
119 Songdomunhwa-ro, Yeonsu-gu, Incheon, Korea

and a video camera. In this context, we train different complex models for each modality and the combination thereof (i.e., we develop both single- and fusion-based approaches). That way, we are able to combine the strengths of both sensors and provide a robust solution for indoor human activity recognition in different environments. Accordingly, wherever necessary privacy can be maintained in sensitive areas by disabling the video camera sensor and predict activities solely based on a radar-based model.

We test our approaches on two different activity data sets. One data set focuses on small gestures, representing hand-based motions and is highly relevant for the creation of intelligent human-machine interfaces. The second data set focuses on events that occur in daily life. Typical examples of such events are standing up or leaving a room. This data set is relevant towards the creation of smart health monitoring systems. Due to the lack of publicly available data sets that contain indoor activities recorded by both a radar and camera sensor, we have created these data sets and make them publicly available. Each data set contains six different activities performed by nine different subjects.

To summarize, the main contributions of our research efforts are as follows:

1. We propose robust classification models that are independent of sensor placement and room setup, while being highly effective at predicting fine- and coarse-grained activities, hereby employing a low-power radar.
2. We compare six different DNN-based architectures on different input modalities that originate from high-dimensional sensors. We show that a three-dimensional CNN taking as input subsequent range-Doppler maps and a 34-layer residual CNN is optimal for radar and video data, respectively.
3. We study the fusion of video- and radar-based model to achieve a complimentary approach which is effective in imperfect circumstances.
4. By publishing the data sets, we aim to facilitate future follow-up research and benchmarking.

The rest of the paper is organized as follows. In Section 2, we briefly review related work in the area of video- and radar-based activity recognition. Section 3 provides an overview of the utilized sensors. In Section 4 and Section 5, we describe the basics of DNNs and the proposed approaches, respectively. In Section 6, we outline the experimental setup used to validate our approach, and in Section 7, we provide an in-depth discussion of our experimental results. Finally, we present our conclusions in Section 8.

## 2 Related Work

Activity recognition is a widely-studied and relevant topic applicable to many daily challenges. The authors of [15] define an action as *"the most elementary human-surrounding interaction with a meaning"*. The term activity is looked upon as a sequence of more rudimentary actions. However, both terms are interchangeably used in literature. In general, action or activity data sets range from coarse and clearly discernible actions such as *Brushing Teeth* and *Basketball Dunk* (part of the UCF101 data set[1]) to more subtle gestures such as *Thumbs Up* and *Thumbs Down* (part of the Jester data set[2]).

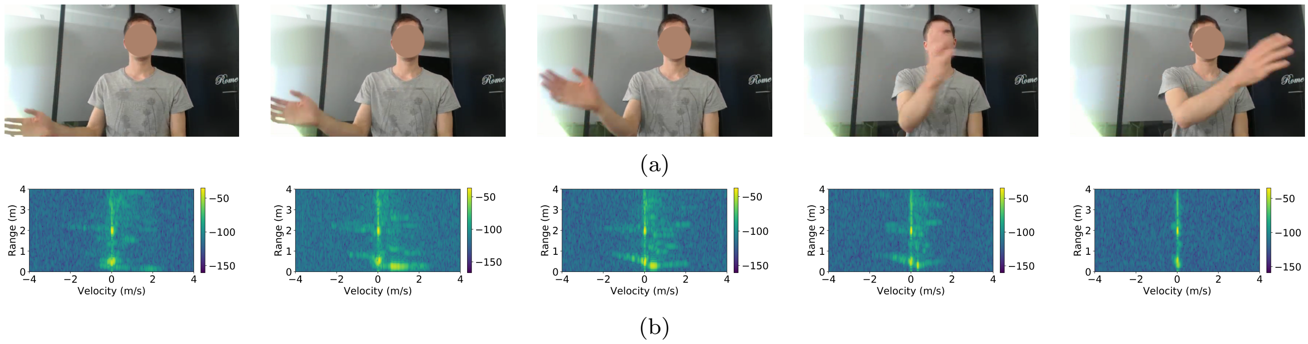
The sensor most frequently used to tackle the challenge of activity recognition is a video camera. DNNs have been predominantly employed to acquire state-of-the-art performances on these data sets. A pioneering study by [17] attempts to train a three-dimensional convolutional neural network (CNN) to exploit the temporal structure of video data. Specifically, a number of different architectures are tested with the aim of fully exploiting the temporal and spatial information on a YouTube-based data set with 487 sports-related classes. In [25], these ideas are extended by investigating smart temporal pooling techniques, as well as using long short-term memory (LSTM) networks with the aim of leveraging longer temporal sequences. These studies are followed by a plethora of research efforts that build upon these ideas to develop accurate solutions for activity recognition [8, 10, 29, 31]. The authors of [26] investigate a video-based approach for hand gestures recognition. To that end, they show that it is crucial to also explicitly learn features along the temporal dimension. As opposed to the initial attempt of [17], the relatively recent release of significantly large activity-related video data sets have enabled the effective training of three-dimensional residual networks [13].

A different type of sensor that is becoming increasingly popular is radar. This sensor can compensate for many of the disadvantages a video camera suffers from. Use cases that have been tackled using radar devices range from security applications trying to detect violent intents [11, 28] to elderly monitoring applications that aim at detecting walking behavior or people falling [12, 22, 34]. The authors of [19, 33] attempt to recognize gestures using DNNs and a radar device. They achieve an accuracy of 87% and 93% for eleven and ten different gestures, respectively.

The combination of both a video and radar sensor has been less investigated, mainly receiving some at-

[1] http://crcv.ucf.edu/data/UCF101.php

[2] https://20bn.com/datasets/jester



(a)

(b)

Fig. 1: Sequence showing one sample of gesture *Swiping left* for (a) video recording and (b) radar recording. The five RD maps shown at the bottom display the velocity ($x$-axis) in relation to the range ($y$-axis). Color scale: the power of the reflected signal (in dB).

tention from the automotive field. In that regard, multiple sensors are for example combined to increase the efficacy of tracking multiple objects in a particular environment [6, 35]. The combination of both sensors can also be employed for detecting hazardous situations in vehicles [9] or monitoring an environment to help navigate visually impaired people more safely [23].

In this study, and setting us apart from the research efforts reviewed above, we compare and combine the use of two different high-dimensional sensors as input for multiple DNNs, with the aim of automatically recognizing a wide range of indoor human activities. To that end, we significantly extend upon the work of [27], in which the subject of automatic activity recognition using a radar and video camera sensor is briefly explored. Specifically, Polfliet et al. [27] only partly focuses on activity recognition using a radar and camera sensor by constructing a limited activity data set consisting of 540 samples distributed over three *events*. Due to the low number of activities and samples in the data set, only a limited analysis of the effects of combining both sensors is given. In this paper, we investigate the efficacy of employing each individual sensor for the use cases at hand and we extensively analyze the potential benefit of combining both sensors on two newly created and large data sets.

## 3 High-Dimensional Sensors

In this study, two different high-dimensional sensors are employed, namely an FMCW radar and a video camera.

An FMCW radar works through emitting a modulated electromagnetic wave towards moving or static targets. The transmitted radiation that scatters on these targets is intercepted by the receiving antenna and can deliver rich information. Based on the time delay, phase shift, or frequency shift, valuable properties such as distance, velocity, size, and orientation can be extracted from the different targets [3]. Specifically, a target in the LOS of the radar moving at constant speed will induce a constant Doppler frequency shift. Coherently, with the translation of the main body, multiple smaller moving parts result in micro-motion dynamics. Such dynamics generate Doppler modulations on the reflected signal, defined as the MD effect [4].

A 77 GHz FMCW radar can be produced at low cost while being relatively power efficient. The disadvantage of this lower power consumption is a degraded signal-to-noise (SNR) ratio [21]. The reflected signal is typically processed by applying a two-dimensional Fourier transform, resulting in range-Doppler (RD) maps that show range and velocity information of all objects in the LOS of the radar [5]. In Fig. 1, an example of the gesture *Swiping left*, which is recorded with a video and radar sensor, is shown. Specifically, Fig. 1b shows five sequential RD maps. The $x$-axis in these RD maps represents the Doppler dimension, also referred to as Doppler channels throughout this paper, while the $y$-axis represents the range dimension. The zero Doppler channels contain the reflections of all static objects in the room and thus result in higher power. To obtain an MD signature, RD maps are summed over the range dimension and concatenated over time. Fig. 2 shows the MD signature of a sample that represents the activity *Shaking*, with the $x$-axis representing the time dimension and the $y$-axis denoting the Doppler dimension.

A video camera works by measuring light rays coming in through a lens. These incoming light rays are turned into electrical signals by for example a CCD (Charge-Coupled Device) or CMOS (Complementary Metal-Oxide Semiconductor) image sensor. In this study, we simply made use of a full HD webcam. Fig. 1a shows a sequence of five images, capturing the gesture *Swiping left*.

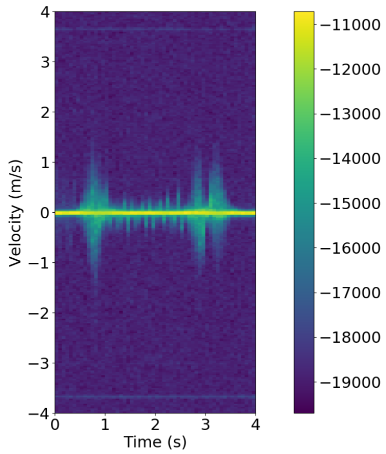


Fig. 2: MD signature displaying the gesture *Shaking*. The $x$-axis represents the time dimension, while the $y$-axis represents the velocity. Color scale: accumulated power levels (in dB) after summing over each RD map.

## 4 Deep Machine Learning

Deep learning or hierarchical learning is a subfield of machine learning that aims at the automatic construction of tailored features based on a stack of nonlinear operations. In particular, algorithms in the field of deep learning aim at automatically creating feature hierarchies, typically through the use of multi-layered feed-forward neural networks (FFNNs) [2]. A FFNN consists of a chain of functions that allow the learning of increasingly complex concepts by stacking many simpler functions:

$$f(\boldsymbol{x}) = f^L(f^{L-1}(\dots f^1(\boldsymbol{x}))), \tag{1}$$

$$f^\ell(\boldsymbol{x}) = \sigma(\mathbf{W}^\ell \boldsymbol{x} + \mathbf{b}^\ell), \qquad \forall \ell \in \{1..L\}, \tag{2}$$

where $\boldsymbol{x}$ represents an input vector, $L$ denotes the number of layers in the network, $\sigma$ represents a piece-wise nonlinear function, and $\mathbf{W}^\ell$ and $\mathbf{b}^\ell$ describe the layer-specific weights and biases, respectively. The piece-wise nonlinear operation $\sigma$ is commonly chosen to be the rectifier linear unit (ReLU) [32], and where this function is defined as follows: $\mathrm{ReLU}(x) = \max(0, x)$.

In this work, we focus on deep convolutional neural networks (DCNNs), long short-term memory networks (LSTMs), and a combination thereof. DCNNs make use of neurons that are only locally connected and that share weights. This means that convolutional filters work on small local receptive fields of input data in a sliding-window fashion [20]. This specialized kind of neural network has a grid-like topology. Different filters evolve to become specific feature detectors, for instance ranging from low-level color and edge detectors in early layers to high-level object detectors in later layers [36]. The essential difference with a standard FFNN is the use of convolutions instead of plain matrix multiplications.

In Fig. 3, an example of a two-dimensional convolution is shown with a kernel of size 2 × 2 and stride 1[3]. The mathematical operation of such a convolution is defined as follows:

$$\mathbf{S}_{ij} = (\mathbf{X} * \mathbf{K})_{ij} \tag{3}$$

$$= \sum_m \sum_n \mathbf{X}_{i+m,j+n} \mathbf{K}_{mn}, \tag{4}$$

with $\mathbf{S}$ denoting the resulting feature map, $\mathbf{X}$ a two-dimensional input, and $\mathbf{K}$ a kernel $\in \mathbb{R}^{m \times n}$. Compared to a regular FFNN, Eq. 2 can be modified as follows:

$$f_j^\ell(\mathbf{X}) = \sigma(\mathbf{X} * \mathbf{W}_j^\ell + \mathbf{b}_j^\ell), \forall \ell \in \{1..L\}, \tag{5}$$

with $f_j^\ell$ depicting the $j$-th feature map of layer $\ell$.

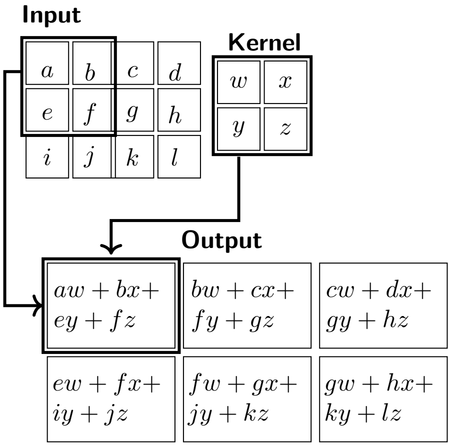


Fig. 3: Example of a two-dimensional convolutional operation. A 2 × 2-sized kernel is convolved over a 3 × 4-sized input with zero padding. The operation of each element is exactly described in the resulting output feature map.

Residual neural networks, a specific type of neural networks, are widely used to facilitate effective learning while enabling very deep architectures [14]. In Fig. 4, a basic building block is shown, which forms the foundation of a residual network. Specifically, the convolutional layers inside such a basic building block are explicitly modeled to fit a residual mapping. The original mapping is recast into $F(\mathbf{x}) + \mathbf{x}$. It is hypothesized that it is easier to optimize the residual mapping than to optimize the original, unreferenced mapping [14]. A residual neural network consists of a series of such basic blocks.

[3] From a strict point-of-view, we are dealing with a cross-correlation, as the kernel is not flipped.

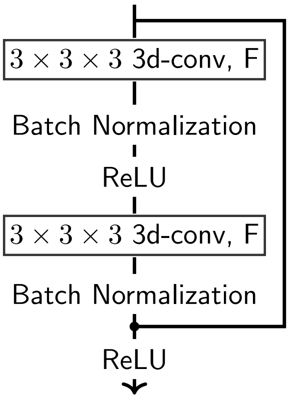


Fig. 4: Schematic overview of the basic building block of a residual neural network.

An LSTM network is optimally suited to model dynamic processes [16]. Such a network, which consists of so-called LSTM cells, belongs to the family of recurrent neural networks (RNN). RNNs differ from regular FFNNs in that they contain feedback loops. These feedback loops encode contextual information of a temporal sequence. Given a certain input sequence $\boldsymbol{x} = (x_1, x_2, \ldots, x_T)$, with $\mathbf{x_t}$ a feature vector given at time $t$, the hidden states of a recurrent layer $\boldsymbol{h} = (h_1, h_2, \ldots, h_T)$ and the outputs $\boldsymbol{y} = (y_1, y_2, \ldots, y_T)$ can be obtained as follows:

$$h_t = \sigma(W_{ih}x_t + W_{hh}h_{t-1} + b_h), \tag{6}$$
$$y_t = W_{ho}h_t + b_o, \tag{7}$$

where the $W$ terms denote weight matrices (e.g., $W_{ih}$ is the input-hidden weight matrix), the $b$ terms denote bias vectors (e.g., $b_h$ is the hidden bias vector), and $\sigma$ is the hidden layer activation function, typically the logistic sigmoid function.

As depicted in Fig. 5, the LSTM architecture uses memory cells to store and output information, allowing it to better discover long-range temporal relationships. The hidden sequence $\mathbf{h}$ of an LSTM cell is computed as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i), \tag{8}$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f), \tag{9}$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \tag{10}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o), \tag{11}$$
$$h_t = o_t \tanh(c_t) \tag{12}$$

where $\sigma$ is the logistic sigmoid function, and $i$, $f$, $o$, and $c$ are the input gate, forget gate, output gate, and cell activation vectors, respectively. By default, the value stored in the LSTM cell $c$ is maintained, unless it is added to by the input gate $i$ or diminished by the forget gate $f$. The output gate $o$ controls the emission of the memory value from the LSTM cell.

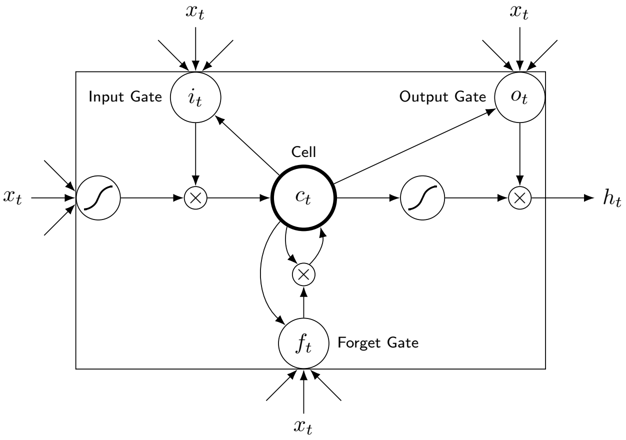


Fig. 5: Graphical display of an LSTM unit, showing the relationship between the different gate connections and the memory cell $c_t$.

## 5 Proposed Approach

The goal of the research effort presented in this paper is to identify activities by leveraging a low-power FMCW radar and a video camera sensor in an indoor environment. The key research questions we attempt to answer are:

1. can we accurately recognize activities with a different granularity using a low-power 77 GHz FMCW radar,
2. what is the most accurate input representation and network architecture to recognize activities given this radar device,
3. what is the added value when combining models based on a radar and video sensor in less than ideal circumstances?

In this section, we discuss the different preprocessing steps and machine learning algorithms used to address the aforementioned questions.

In Fig. 6, a schematic overview is given of the proposed approach. The different steps are defined as follows: (a) a single subject is captured synchronously by a low-power radar and a video camera while performing an activity. This activity can either be an event (representing an activity containing larger movements) or a gesture (representing specific hand-oriented motions), (b) the recorded signals are processed and result in RD maps and RGB images for the radar and camera sensor, respectively, (c) fragments of $k$ seconds of data are used as input for separate deep neural networks per modality and activity category, (d) via late fusion, the predictions of each sensor-specific network are combined to compensate for weaknesses of both, and (e) predictions are outputted over six activities.

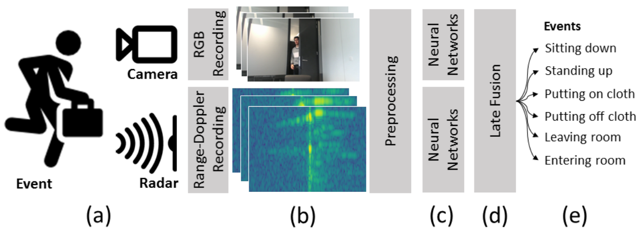


Fig. 6: Schematic overview of the proposed approach for the *events* data set. The proposed approach is similar when considering the *gestures* data set.

## 5.1 Recording & Preprocessing

In this work, an FMCW radar device produced by INRAS [1] is used. We employ the radar in Single Input Single Output (SISO) mode and the recording parameters are given in Table 1.

Table 1: Recording parameters of the FMCW radar. Fine-grained capturing of detailed movements is possible thanks to a range and velocity resolution of 10 cm and 2 cm/s, respectively.

| Waveform Parameters | | Sensing Parameters | |
|---|---|---|---|
| Center freq. | 77 GHz | Range resolution | 10 cm |
| Chirp bandwidth | 1.5 GHz | Velocity resolution | 2 cm/s |
| Chirp duration | 256 µs | Ambiguous range | 38.4 k m |
| Sampling freq. | 2 GHz | Ambiguous velocity | 13.68 k m/h |

An RD map is obtained by applying a 2-dimensional Fourier transform, subsequently converting the absolute value of the signal to decibels (dB). This results in an RD map containing 256 Doppler channels, representing velocities from −3.8 m/s to 3.8 m/s, and 160 range channels, representing a range varying from 0.5 m to 4.5 m. The MD signature is computed by summing the RD maps over the range dimension, thus containing the same 256 Doppler channels per time unit. The time dimension is represented by the frequency for which an RD map is produced by the radar device. In this case, a total of 256 chirps are emitted, with each chirp having a duration of 256 µs, thus resulting in approximately 15 frames per second (FPS). Similar to [30], we remove the three middle static Doppler channels, representing objects with zero velocity as these primarily consist of room characteristics.

A full HD webcam device is used to record each activity in the visual domain. To that end, the camera is positioned on top of the radar device, with both sensors recording synchronously. To decrease the amount of data, the frame resolution is reduced to 341 × 256 pixels and, like the FMCW radar, the speed of recording is set at 15 FPS.

## 5.2 Neural Network Architectures

DNNs are not only well suited to deal with noisy data but also have the ability to automatically infer features from raw data [2]. Both properties are crucial elements to answer the research questions that have been put forward, given that we are dealing with challenging radar and video camera data. In what follows, we describe the designed neural network architectures to predict the listed activities. We describe five different architectures that output predictions based on radar input and one architecture that outputs predictions based on video input.

### *5.2.1 Radar-based Classification*

Fig. 7 shows the exact architecture of each of the five radar-based networks. Each network consists of a combination of either convolutional, pooling, LSTM, or fully-connected layers. Dropout is applied to any layer (except the last) that consists of trainable weights, with an increasing rate depending on the proximity to the final layer. Each convolutional and fully-connected layer is followed by an Exponential Linear Unit (ELU) nonlinearity operation. The ELU non-linearity is defined as follows:

$$\mathrm{ELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(\exp(x) - 1) & \text{if } x \leq 0 \end{cases}, \tag{13}$$

with $x \in \mathbb{R}$ representing the input and $\alpha$ a predefined parameter greater than zero. As described in [7], ELU non-linearities possess improved learning characteristics as opposed to other non-linearities. Indeed, an effect similar to batch normalization, but with a lower computational complexity, is accomplished by pushing mean unit activitations closer to zero, thanks to the negative values allowed by this non-linearity. The last fully-connected layer uses a softmax non-linearity to produce outcome probabilities for each target class.

The architectures *LSTM*, *1d-CNN-LSTM*, and *2d-CNN* take as input two-dimensional MD signatures.

More precisely, the *LSTM* network extracts features from the Doppler data by applying a fully-connected layer of size 64. The time dimension is handled by applying a bidirectional recurrent layer consisting of 32 LSTM units. Finally, the network makes a prediction based on two fully-connected layers of size 128 and 6, respectively. In the *1d-CNN-LSTM*, the first fully-connected layer of the *LSTM* network is replaced by three one-dimensional convolutional layers that attempt to extract features from the Doppler data. The convolutional layers possess 8, 16, and 32 filters of size 3, respectively. Each convolutional layer is followed by a non-overlapping one-dimensional pooling layer of size 3. The *2d-CNN* network does not use a recurrent layer but attempts to jointly extract features from the Doppler and time dimensions through the use of four two-dimensional convolutional layers with 8, 16, 32, and 64 filters, respectively. In order to reduce the input dimensions, each convolutional layer is followed by a non-overlapping two-dimensional pooling layer of size 2 × 2.

The architectures *2d-CNN-LSTM* and *3d-CNN* take three-dimensional RD maps as input. The *2d-CNN-LSTM* network attempts to combine feature extraction through the use of convolutional layers with a recurrent layer to handle the time dimension, similar to the *1d-CNN-LSTM* network. More precisely, the one-dimensional convolutional and pooling layers are replaced by their two-dimensional counterparts. The *3d-CNN* is similar to the *2d-CNN* network but increasing the number of dimensions with one for both the convolutional and pooling layers.

By investigating these five networks, we aim at understanding the influence of the different nature of the input data on automatic activity recognition, so to be able to develop an adequate solution.

#### *5.2.2 Video-based Classification*

The video-based model is a three-dimensional CNN consisting of 34 layers with a residual structure, pretrained on the Kinetics-400 data set [18]. This model achieves a top-1 accuracy of 60.1% over 400 classes [13]. As described in Table 2, we employ a slightly modified form of this network, taking as input a stack of sequential RGB images, while giving as output predictions over six classes. Spatial downsampling is performed by conv1, conv3_1, conv4_1, and conv5_1 with a stride of two. Temporal downsampling is performed in conv3_1, conv4_1, and conv5_1 with the same stride. This network is referred to as *3d-ResCNN* throughout this paper.

Table 2: Specifications of the *3d-ResCNN* architecture. The basic building block represents the core of the residual network and has been previously explained in Section 4. $F$ represents the number of filters learned in each convolutional layer of a block. The last fully-connected layer contains six output neurons and is followed by a softmax non-linearity function.

| Layer | Specifications |
|---|---|
| conv1 | $7 \times 7 \times 7$ 3d-conv, 64 |
| conv2_x | 3 basic building blocks, $F = 64$ |
| conv3_x | 4 basic building blocks, $F = 128$ |
| conv4_x | 6 basic building blocks, $F = 256$ |
| conv5_x | 3 basic building blocks, $F = 512$ |
| pool | global avg. pooling |
| fc | fully-connected, 6, softmax |

It has been shown in [13] that deeper residual networks obtain marginally better accuracy results on the above mentioned Kinetics data set. However, taking into account our limited set of categories, we deem ResNet-34 to consist of the optimal performance-to-size ratio. By pretraining this network on a vast data set, we enable the learning process to efficiently jump local minima and quickly converge to a near-optimal solution.

## 6 Experimental Setup

In this section, we describe the characteristics of the constructed data sets, our approach towards learning, and the way we evaluated the proposed methods.

### 6.1 Data Sets

In this study, we investigate a multi-sensor- and neural network-based approach towards automatic human activity recognition. Accordingly, we develop and evaluate the proposed solutions in two relevant application domains. To that end, we constructed two realistic and extensive data sets. The first data set concerns fine-grained activities, namely gestures that are performed with any of two hands. As described before, this category is tailored towards the development of advanced human-machine interfaces. The second category entails coarse-grained activities that are useful to develop smart health monitoring tools. The two data sets are referred to as *gestures* and *events*, respectively.

In order to construct two data sets that can be deemed large, we have recorded nine different subjects in two different environments. Both environments entail a meeting room in which the sensor set up is directed towards the exit. The gestures are performed while sitting on a chair in front of both sensors. The sensor setup

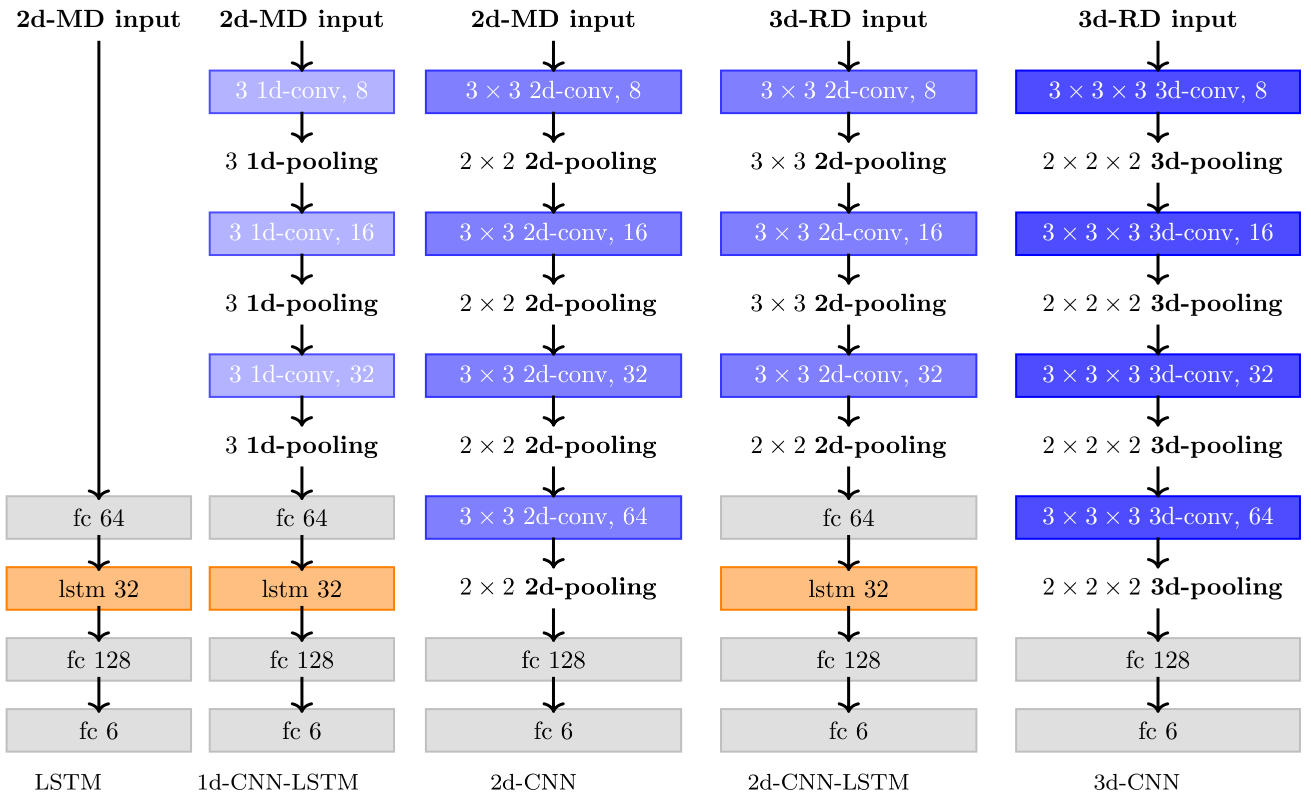
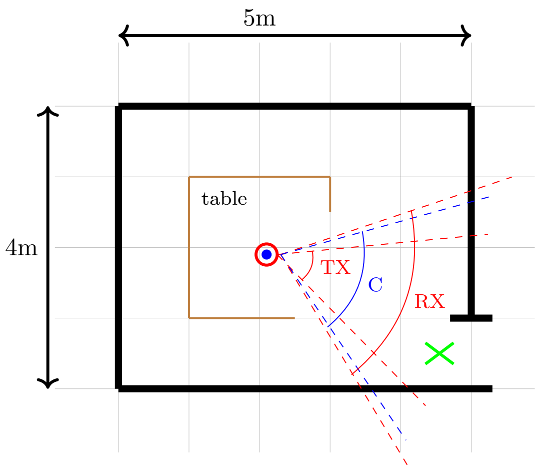


Fig. 7: Schematic diagram of five different neural network architectures. The architectures *LSTM*, *1d-CNN-LSTM*, and *2d-CNN* take two-dimensional MD signatures as input. The two subsequent architectures, *2d-CNN-LSTM* and *3d-CNN*, take three-dimensional RD maps as input.



visualization of a recording environment. The radar and video camera sensors are depicted by a red and blue circle, respectively. The radar LOS is characterized by the receiving beamwidth (RX) covering 76.5° and the transmitting beamwidth (TX) covering 51°. The horizontal LOS of the video camera (C) covers 70°. The green cross denotes a possible target.

is conceptually displayed in Fig. ??. The camera is positioned on top of the radar device, with both sensors recording synchronously. The camera device covers a horizontal field of view of 70° and the radar sensor has a receiving beamwidth of 76.5° in combination with a transmitting beamwidth of 51°. Table 1 shows the specifications of the employed radar.

Every subject was repeatedly recorded in a continuous way for seven minutes, during which they performed all gestures and events. Multiple recordings were performed per subject, alternating between the two recording environments. These recordings were labeled by segmenting the video-based streams into one of twelve activities. However, it should be noted that not all subjects were able to perform the same number of recordings. Moreover, each subject performs the different activities at different speeds and pausing intervals. This uncontrolled approach allows for less generic and more diverse activity recordings since the length of an activity is not predetermined, nor the order in which these should be performed. As a result, the data sets are characterized by non-equal distributions of the number of activities per subject. In Table 3, an overview of the count per activity can be found, along with the average duration of each activity. In Table 4, an overview of the number of all gestures and events per subject is given.

Table 3: Overview of all recorded activities.

| Activity | Abbr. | Total | Avg. duration |
|---|---|---|---|
| Drumming | D | 390 | 2.92s (±0.94) |
| Shaking | S | 360 | 3.03s (±0.97) |
| Swiping Left | $S_l$ | 436 | 1.60s (±0.27) |
| Swiping Right | $S_r$ | 384 | 1.71s (±0.31) |
| Thumb Up | $T_u$ | 409 | 1.85s (±0.37) |
| Thumb Down | $T_d$ | 368 | 2.06s (±0.42) |
| Entering Room | E | 221 | 3.01s (±0.73) |
| Leaving Room | L | 224 | 3.94s (±0.78) |
| Sitting Down | $S_d$ | 342 | 1.98s (±0.31) |
| Standing Up | $S_u$ | 344 | 1.65s (±0.28) |
| Clothe | C | 195 | 5.62s (±1.76) |
| Unclothe | U | 179 | 4.97s (±1.09) |

Our data sets contain 3852 activities in total, taking on average 2.56 s per activity, subdivided in 1505 event-related activities and 2347 gesture-related activities. Our data sets thus contain a total of 2.74 hours of effectively annotated activity data distributed over 12 classes. As is depicted in Table 3, the extent of time in which each activity is performed differs significantly per activity class. On the one hand, gestures such as *Swiping Left* or *Swiping Right* and *Thumb Up* or *Thumb Down* are performed in 2 s or less. On the other hand, certain events such as *Clothe* or *Unclothe* can require up to 5 s or more. Moreover, there is a large intra-class variability for the duration of certain activity classes, as shown by the standard deviation. These properties add to the diversity of the constructed data sets and to the challenging nature of the research questions we set out to answer. Both data sets are made publicly available under the name HARRad (Human Activity Recognition with a Radar) to facilitate further research[4].

## 6.2 Learning

Our models are trained on GeForce GTX 980 and Titan X graphics cards. We used the PyTorch[5] library to implement and test our different approaches. Gradients are computed over minibatches of size 64 for both the radar- and video-based models. We use the Adam optimizer with a learning rate of $10^{-3}$ for all non-pretrained radar-based models and $10^{-4}$ for the pretrained video-based models. The best validation loss is used after training for 500 and 50 epochs for the radar- and video-based models, respectively.

Table 4: Number of recorded events and gestures per subject $S_i$, with $i \in \{1 \ldots 9\}$.

| | Gestures | | | | | | Events | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | D | S | $S_l$ | $S_r$ | $T_u$ | $T_d$ | E | L | $S_d$ | $S_u$ | C | U |
| $S_1$ | 49 | 40 | 44 | 22 | 41 | 37 | 20 | 20 | 29 | 27 | 30 | 22 |
| $S_2$ | 89 | 80 | 99 | 92 | 83 | 80 | 78 | 79 | 83 | 88 | 73 | 71 |
| $S_3$ | 44 | 43 | 48 | 46 | 35 | 38 | 14 | 14 | 33 | 33 | 13 | 14 |
| $S_4$ | 33 | 34 | 35 | 35 | 32 | 32 | 32 | 33 | 51 | 50 | 33 | 33 |
| $S_5$ | 40 | 40 | 45 | 46 | 52 | 47 | 22 | 22 | 43 | 43 | 14 | 14 |
| $S_6$ | 45 | 45 | 46 | 47 | 58 | 52 | 15 | 16 | 32 | 34 | 8 | 8 |
| $S_7$ | 46 | 40 | 72 | 62 | 72 | 47 | 28 | 28 | 45 | 42 | 17 | 12 |
| $S_8$ | 17 | 15 | 20 | 23 | 17 | 19 | 5 | 5 | 6 | 6 | 5 | 3 |
| $S_9$ | 27 | 23 | 27 | 11 | 19 | 16 | 7 | 7 | 20 | 21 | 2 | 2 |

The MD and RD data are min-max normalized and the video data are rescaled to the interval $[0, 1]$. For training of any radar-based model, random shifting inside an activity sample is performed when this sample is longer than the selected classification time range. This is done to increase data diversity and to enable the learning of robust models. When the activity is shorter than this time range, the last frame is repeated. As discussed in Section 5, the static Doppler channels are removed from both the MD signatures and the RD maps. Furthermore, we quartered the dimensions of each RD map to $40 \times 63$ by applying linear interpolation. No other data augmentation techniques are used for these models. The video-based models make use of random square crops of size $112 \times 112$ after having resized the frames to $170 \times 128$. Random horizontal flipping, brightness, and saturation augmentations are also applied.

The computational complexity of training the proposed models significantly depends on the size of the model and the input data. The models that are based on MD input take on average 1.2 s to complete one epoch. Each of these models is trained for 500 epochs which results in a training time of around ten minutes. The RD-based models take around 5.4 s and thus train for 45 minutes to complete the same number of epochs. In contrast, the video-based model is significantly larger and takes around 78 s to complete one epoch. However, since this model is pretrained, convergence can be attained in 50 epochs taking around 65 minutes.

## 6.3 Evaluation

We report the error rate, which is defined as the number of wrongly classified samples compared to the total number of samples. A sample is defined as a set of consecutive frames in which a subject performs an activity

[4] The data sets are publicly available at: *https://www.imec-int.com/en/harrad*

[5] https://pytorch.org

Table 5: Error rate for leave-one-subject $S_i$-out cross-validation ($\overline{S}$), with $i \in \{1 \ldots 9\}$, and stratified random split ($RS$) for gestures and events, feeding MD signatures (a, b, and c), RD maps (d and e), and RGB images (f) as input to various DNNs. Networks (a) to (f) refer to *LSTM*, *1d-CNN-LSTM*, *2d-CNN*, *2d-CNN-LSTM*, *3d-CNN*, and *3d-ResCNN*, respectively. The lowest radar-based error rates are highlighted in bold.

| | Gestures | | | | | | | Events | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| input | Micro-Doppler | | | Range-Doppler | | RGB | | Micro-Doppler | | | Range-Doppler | | RGB | |
| network | (a) | (b) | (c) | (d) | (e) | (f) | (f) | (a) | (b) | (c) | (d) | (e) | (f) | (f) |
| $S_1$ | 25.18 | 25.32 | 25.32 | **9.30** | 10.73 | 1.43 | 23.89 | 14.19 | 13.96 | **13.29** | 14.41 | **13.29** | 15.09 | 23.42 |
| $S_2$ | 18.99 | 19.18 | 22.75 | 6.05 | **5.61** | 0.25 | 16.06 | 8.05 | **6.36** | 9.18 | 7.98 | 9.18 | 5.16 | 11.23 |
| $S_3$ | 30.18 | 28.48 | 28.74 | 12.60 | **11.68** | 4.86 | 17.98 | 2.48 | **1.65** | 2.48 | 2.75 | 2.75 | 6.06 | 21.49 |
| $S_4$ | 38.97 | 31.67 | 31.34 | 19.73 | **16.92** | 0.50 | 18.91 | 5.89 | 5.46 | 7.90 | 4.60 | **2.30** | 2.16 | 13.07 |
| $S_5$ | 32.96 | 30.62 | 31.73 | 15.93 | **12.84** | 4.57 | 20.49 | 1.27 | **1.05** | 2.95 | **1.05** | **1.05** | 2.95 | 14.56 |
| $S_6$ | 30.83 | 27.65 | 29.01 | 17.86 | **15.81** | 1.93 | 19.11 | 3.24 | 3.54 | 3.54 | **1.18** | 1.47 | 2.65 | 25.07 |
| $S_7$ | 31.66 | 28.52 | 29.11 | **12.09** | 15.04 | 1.18 | 12.68 | 5.43 | 4.65 | 5.81 | 4.07 | **3.88** | 2.13 | 12.40 |
| $S_8$ | 32.13 | 38.74 | 32.43 | **17.12** | 17.42 | 6.31 | 28.53 | 3.33 | **2.22** | 4.44 | **2.22** | **2.22** | 5.56 | 25.56 |
| $S_9$ | 30.89 | 30.89 | 31.71 | **13.82** | 18.97 | 3.25 | 30.62 | 3.39 | 0.56 | **0.00** | **0.00** | **0.00** | 1.69 | 16.95 |
| $\overline{S}$ | 30.20 | 29.01 | 29.13 | **13.89** | **13.89** | 2.70 | 20.92 | 5.28 | 4.38 | 5.51 | 4.25 | **4.02** | 4.83 | 18.19 |
| $RS$ | 15.28 | 12.50 | 15.56 | **2.50** | 5.28 | 0.28 | 16.67 | 4.72 | 4.17 | 6.67 | 4.17 | **2.78** | 3.61 | 15.28 |

from beginning to end. The label of a sample is predicted based on a fragment of $k$ seconds, temporally cropped from the middle frames of the sample. Two different methods are applied to correctly evaluate our different approaches. The first is leave-one-subject-out cross-validation for which we report the average validation error rate over all splits. Since we have an unbalanced distribution of the number of labels per subject, we also report training, validation, and test error rate for a stratified randomized split, with each activity having a fixed number of 20 and 50 samples in the validation and test set, respectively. This ensures that there is no over- or under-representation of classes in the validation or test set. Similar to the training procedure, we extend samples that contain less than the required number of consecutive frames by repeating the last frame.

## 7 Results

In this section, we give a detailed overview of a number of experiments. First, six different network architectures are analyzed, quantifying the effect of each model on the effectiveness of activity recognition. Second, the sample length is investigated in order to determine the optimal amount of temporal data that are necessary for both the gestures and events data set. Third, we analyze the combination of a video- and radar-based model, measuring the effect of sensor fusion. Finally, we give an overview of the best performing model and its configuration, also providing a number of additional insights.

### 7.1 Analysis of Micro-Doppler as Input Modality

We analyze the effectiveness of using MD signatures as the input for three different DNN architectures. These architectures are described in more detail in Section 5. They are defined as (a) *LSTM*, (b) *1d-CNN-LSTM*, and (c) *2d-CNN*. The results for the cross-validation ($\overline{S}$) and random stratified split ($RS$) are listed in Table 5. The sample length of all MD inputs is fixed to 2 s or 30 frames.

Table 5 shows that we cannot observe a clear difference among the three networks for both the gestures and the events data set. However, a significant difference can be noted when comparing the effectiveness between the two separate data sets. While the best performing network for predicting gestures achieves an error rate of 29.01%, the best performing network for predicting events achieves an error rate of 4.38%. In both cases, this is the *1d-CNN-LSTM* network. Therefore, we can conclude that MD signatures provide sufficient information to tackle clear and distinct activities but fail to grasp more fine-grained movements that occur frequently in smaller gestures.

### 7.2 Analysis of Range-Doppler as Input Modality

We analyze the use of RD maps as input for two different networks, namely (d) *2d-CNN-LSTM* and (e) *3d-CNN*. By maintaining the range dimension, we hypothesize that these models will be able to better recognize fine-grained activities such as gestures. Table 5 shows the results obtained for both networks. Indeed, the use of RD maps directly enables the use of more advanced networks that are able to take into account the extra information in an effective way. This is proven by comparing the three MD-based networks to the two RD-based networks. In general, for the gestures data set, the error rate is decreased by more than 50% to 13.89% and 5.89% for the $\overline{S}$ and $RS$ evaluation methods, respectively. This difference in error rate is not observed for the events data set, as MD-based networks already achieved error rates down to 4.38%. In this case, the error rate is improved to 4.02% by the *3d-CNN* network. Furthermore, there is no significant difference between the use of a three-dimensional CNN compared to a two-dimensional CNN that integrates an LSTM layer.

Regarding the difference between the two evaluation methods ($\overline{S}$ and $RS$), we can conclude that there is a significant difference in error rate when focusing on the gestures data set. It is clearly beneficial to allow the network to learn directly from the specific way a subject performs different gestures. We notice an absolute improvement of more than 11% and 8% on the error rate when considering the *2d-CNN-LSTM* and *3d-CNN* networks, respectively. Again, this difference is not significantly noticeable in the case of the events data set as these different events are more general and less person-specific.

## 7.3 Analysis of Video Frames as Input Modality

Notwithstanding the reluctance to use video cameras in an indoor environment because of privacy concerns, it is the primary sensor to tackle the challenge of activity recognition. Moreover, privacy concerns are less relevant in professional environments, which often already deploy video cameras for various applications such as video conferencing or security measures.

In this experiment, we analyze the effectiveness of learning a model to recognize six gestures or six events based on RGB input data. As can be seen in Table 5, using a video camera in normal circumstances significantly outperforms the use of a radar device in the case of the gestures data set. Specifically, the *3d-ResCNN* achieves an error rate of 2.70% and 0.28% on the $\overline{S}$ and $RS$ evaluation methods, respectively. However, this neglects suboptimal settings such as dimly lit environments or obstructing elements in front of the camera sensor. Moreover, the video-based *3d-ResCNN* can take advantage of having been pretrained on a vast online data set, which aids the training of this model. Regarding the events data set, the video-based model achieves similar results as the radar-based models. It is clear that a radar is in this case the most viable option to solve the challenge of activity recognition.

To illustrate the weaknesses of using a video camera as the primary sensor, we repeat the same experiment after artificially darkening the data. For this experiment, we do not retrain the model but test its capacity to deal with these artificial data based on its originally learned weights. We state that this is a fair comparison since radar-based models likewise do not need to be retrained for dimly-lit or dark circumstances. The video frames are darkened by lowering the RGB values by 60%. In Table 5, we can observe that there is a degradation of effectiveness when such modifications are applied to the data. Specifically, the effectiveness obtained for $\overline{S}$ degrades from an error rate of 2.70% to an error rate of 20.92% and from 4.83% to 18.19% for the gestures and the events data set, respectively. These values are significantly worse than the radar-based variants, where the best models achieve an error rate of 13.89% and 4.02% for the two respective data sets.

## 7.4 Analysis of Sample Length

We analyze the optimal sample length for both the gestures and the events data set. Given the average length of a gesture (see Table 3), we hypothesize that the ideal classification length is below 2 s. In case of the events data set, a longer sample length should be more effective. For this experiment, we consider the best performing *3d-CNN* network to measure the influence of the sample length for RD maps. The following results are obtained by evaluating with the cross-validation ($\overline{S}$) method. We assume the optimal sample length measured based on radar data will act as an upper boundary for the error rate produced by the video-based model. This assumption is based on the fact that single static video frames by themselves already contain rich information that can be employed to accurately predict the performed activity. Therefore, the *3d-ResCNN* network is not considered in this experiment.

For the gestures data set, Fig 8 allows observing that the optimal sample length is 20 frames, corresponding to a duration of 1.33 s. For the events data set, we can see that the optimal sample length is significantly longer. The lowest error rate is achieved when using a sample length that is in-between 50 and 60 frames, which corresponds to a duration that is in-between 3.33 s to 4 s. These findings can be attributed to the general observation that gestures correspond to short swift movements, while events can consist of smaller sub-actions that take place over a longer period of time. For all subsequent experiments, we make use of the optimal sample length, which is 20 and 50 frames for gestures and events, respectively.

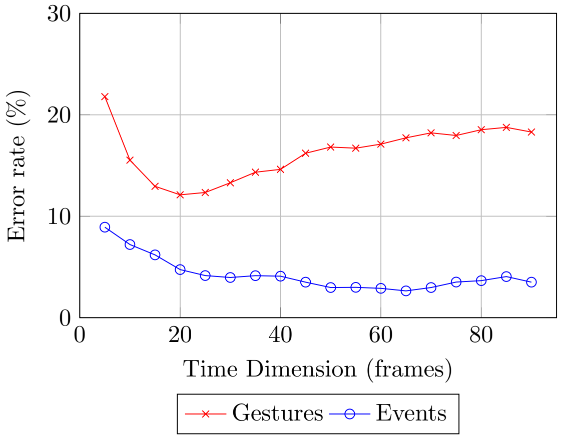


Fig. 8: Analysis of the optimal sample length (in frames) for both the gestures and the events data set. The measured sample length runs from 0 s to 6 s. The results are based on the *3d-CNN* network, taking RD maps as input, and where this network is evaluated by the cross-validation method $(\overline{S})$.

## 7.5 Analysis of Model Complexity

In this section, we analyze the complexity of the models regarding the number of trainable parameters and time efficiency to evaluate one sample. The following results are obtained by executing the networks on a Titan X graphics card. When using the default input sizes of the MD and the RD data in combination with a sample length of 2 s, the MD-based networks *LSTM*, *1d-CNN-LSTM*, and *2d-CNN* possess 36.2 k, 38.3 k, and 148.2 k trainable parameters, respectively. The RD-based networks *2d-CNN-LSTM* and *3d-CNN* contain 52.4 k and 123.0 k trainable parameters, respectively. The prediction of one sample takes on average 3 ms for the MD-based models and 5 ms for the RD-based models. In terms of computational complexity, we conclude there is a negligible difference among the radar-based networks. Moreover, the proposed radar-based models are not restricted based on time efficiency.

In contrast, the residual CNN that is employed for the video-based predictions possesses 63.5 M trainable parameters and takes 20 ms to predict one sample. The large number of trainable parameters shows the necessity to train this network by starting from a set of pre-trained weights, in order to allow effective finetuning of the weights using our constructed data sets. Although, this model takes significantly more time to predict one sample in comparison to the radar-based networks, it is still able to provide real-time predictions at a speed of 50 samples per second.

## 7.6 Sensor Fusion

In this section, we analyze the effectiveness of fusing the predictions of the best performing models, one for each sensor. To that end, we apply late fusion and average the predictions returned by each sensor-specific model. Regarding the radar-specific model, we make use of the *3d-CNN* network, whereas the video-based input is handled by the *3d-ResCNN* model. We use 20 and 50 frames as the input sample length for both data sets. Fig. 9 shows the results of fusion based on the clean data (*Fused*) and the artificially darkened data (*Fused**).

In ideal circumstances and for professional environments without privacy concerns, we can observe that a video sensor outperforms a radar sensor; there is no added value in fusing its predictions with a radar-based model. The fusion of both sensors achieves an error rate that is 7% higher in absolute terms in comparison with the video-based model for the gestures data set, while it achieves similar results in the case of the events data set. However, the combined use of both sensors becomes credible when taking into account the added value of a radar sensor, which can function properly when the effectiveness of a video camera sensor strongly or completely degrades. These results can be read from Fig. 9, where the *Video** category shows the degraded effectiveness of a video sensor in less than ideal circumstances. The *Fused* category shows an absolute improvement of 3.5% and 0.20% over using solely a radar sensor for the gestures and the events data set, respectively. The *Fused** category shows an even more obvious benefit of using both sensors in less than ideal circumstances.

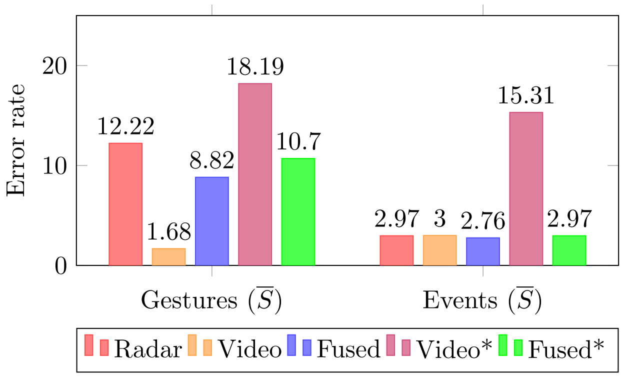


Fig. 9: Average error rates on the leave-one-subject-out cross-validation splits $(\overline{S})$ for both data sets.

Table 6: Error rate for the $\overline{S}$ and $RS$ evaluation methods for the best performing models. In the case of the *Video** and *Fused** categories artificially darkened video data is used as input data for the *3d-ResCNN* model.

| | **Gestures** | | | **Events** | | |
|---|---|---|---|---|---|---|
| | $\overline{S}$ | $RS$ | | $\overline{S}$ | $RS$ | |
| | valid | valid | test | valid | valid | test |
| Radar | 12.22 | 4.17 | 6.00 | 2.97 | 1.67 | 4.56 |
| Video | 1.67 | 0.00 | 1.89 | 3.00 | 3.61 | 3.22 |
| Fused | 8.82 | 3.89 | 3.89 | 2.76 | 1.67 | 4.22 |
| Video* | 18.19 | 10.00 | 10.78 | 15.31 | 10.84 | 8.22 |
| Fused* | 10.70 | 4.17 | 5.11 | 2.97 | 1.94 | 4.33 |

## 7.7 Main Results

In this section, we list the best performing model configuration based on our previous analyses. The exact results can be found in Table 6. We conclude that using the *3d-CNN* network results in the best performing set up, with 20 and 50 consecutive RD maps as input for the gestures and the events data set, respectively. Specifically, this network takes as input $64 \times 1 \times k \times 40 \times 63$-dimensional matrices, with 64 representing the batch size, $k$ equaling 20 or 50 depending on the activity data set, and $40 \times 63$ representing an RD map after resizing and removal of the static Doppler channels. Using the cross-validation evaluation method, this radar-based model achieves an error rate of 12.22% and 2.97% on the events and the gestures data set, respectively. By evaluating on the *RS* test set, the advantage of using person-specific gesture information when training a model becomes evident, given that the error rate lowers to 6.00%. Regarding the video-based model, we achieve an error rate of 1.67% and 3.00% for both data sets, resulting from fine-tuning a pretrained 34-layered residual CNN on clean video sensor data. This error rate increases to 18.19% and 15.31% when testing the same model on artificially darkened video data. Our fusion results show that there is a clear advantage of complementing a video sensor with a radar sensor, even in non-privacy sensitive environments.

Table 7 displays the summed confusion matrices for predictions on each $S_i, i \in \{1 \dots 9\}$ split for both data sets using the *3d-CNN* model. It can be noted that in the case of the gestures data set, the most static activities, namely *Thumb Up* and *Thumb Down*, are confused among each other and are the least accurately recognized. Similar confusion exists between the activities *Clothe* and *Unclothe*. This can be attributed to the very similar nature of both events.

Table 7: The resulting confusion matrix for both data sets after summing the confusion matrices of all splits of the leave-one-subject-out cross validation. The predictions are obtained by the *3d-CNN* radar-based network.

| | Predicted Label | | | | | | | Predicted Label | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| True Label | D | S | $S_l$ | $S_r$ | $T_d$ | $T_u$ | | $S_u$ | $S_d$ | G | E | C | U |
| D | **356** | 16 | 1 | 2 | 5 | 10 | $S_u$ | **342** | 2 | 0 | 0 | 0 | 0 |
| S | 5 | **353** | 0 | 0 | 0 | 2 | $S_d$ | 1 | **337** | 1 | 0 | 3 | 0 |
| $S_l$ | 0 | 3 | **425** | 5 | 0 | 3 | G | 0 | 0 | **224** | 0 | 0 | 0 |
| $S_r$ | 0 | 1 | 12 | **362** | 1 | 8 | E | 0 | 0 | 0 | **221** | 0 | 0 |
| $T_u$ | 12 | 4 | 3 | 2 | **295** | 52 | C | 0 | 1 | 0 | 0 | **175** | 19 |
| $T_d$ | 5 | 2 | 1 | 6 | 56 | **339** | U | 0 | 1 | 3 | 0 | 16 | **159** |
| | **Gestures** | | | | | | | **Events** | | | | | |

In Appendix A, we test the efficacy of our proposed approach on an integrated system that combines both data sets. More specifically, we show that similar results can be obtained using the same networks and configurations when making predictions over the combined data sets of *gestures* and *events*.

## 8 Conclusions

In this paper, we have proposed a novel approach towards automatic indoor human activity recognition, using deep neural networks that take as input data originating from radar and video camera sensors. To that end, we have constructed two data sets that consist of 2347 and 1505 samples distributed over six different types of gestures and events, respectively. When regarding the radar sensor, we concluded that it is optimal to use a three-dimensional CNN that takes as input 20 and 50 sequential RD maps for the gestures and the events data set, respectively. These models achieve 12.22% and 2.97% error rate on the gestures and the events data set, respectively. In the case of privacy-sensitive environments, we suggest to only employ the radar-based solution that can operate in an effective and efficient manner without the need for video cameras. When regarding the camera sensor, we make use of a pretrained residual CNN and obtain 1.67% and 3.00% error rate on the same data sets. In ideal and non-privacy sensitive circumstances, it is optimal to make use of a video camera sensor. However, there is a clear benefit of combining both sensors to enable activity recognition in the case of non-ideal circumstances such as dark environments or in the case the view of a video camera sensor is partially blocked. By artificially darkening the camera sensor data, the effectiveness of these models significantly worsens to 18.19% and 15.31% for both data sets.

By applying late fusion to the predictions obtained from each model, the benefit of using both sensors becomes obvious. To summarize, we successfully built a solution to automatically recognize gestures and events in a realistic scenario, taking advantage of both an FMCW radar and a video camera sensor.

**Acknowledgements** The research activities described in this paper were funded by Ghent University-imec, the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

## Conflict of Interest

The authors declare that they have no conflict of interest.

## References

1. INRAS GmbH (2017). URL `http://www.inras.at`. Accessed 20 Jun. 2017
2. Bengio, Y., Goodfellow, I.J., Courville, A.: Deep learning. Nature **521**(7553), 436–444 (2015)
3. Brooker, G.M.: Understanding millimetre wave FMCW radars. In: 1st international Conference on Sensing Technology, pp. 152–157 (2005)
4. Chen, Q., Tan, B., Chetty, K., Woodbridge, K.: Activity recognition based on micro-doppler signature with inhome Wi-Fi. In: IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 1–6 (2016)
5. Chen, V.C., Li, F., Ho, S.S., Wechsler, H.: Micro-Doppler effect in radar: phenomenon, model, and simulation study. IEEE Transactions on Aerospace and Electronic Systems **42**(1), 2–21 (2006)
6. Cho, H., Seo, Y., Kumar, B.V.K.V., Rajkumar, R.R.: A multi-sensor fusion system for moving object detection and tracking in urban driving environments. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1836–1843 (2014)
7. Djork-Arné, C., Thomas, U., Sepp, H.: Fast and accurate deep network learning by exponential linear units (elus). CoRR **abs/1511.07289** (2015)
8. Donahue, J., Hendricks, L.A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 677–691 (2017)
9. Eshed, O.B., Ashish, T., Sujitha, M., Mohan M., T.: On surveillance for safety critical events: In-vehicle video networks for predictive driver assistance systems. Computer Vision and Image Understanding **134**, 130–140 (2015)
10. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
11. Fioranelli, F., Ritchie, M., Griffiths, H.: Classification of unarmed/armed personnel using the netrad multistatic radar for micro-doppler and singular value decomposition features. IEEE Geoscience and Remote Sensing Letters **12**(9), 1933–1937 (2015)
12. Gurbuz, S.Z., Clemente, C., Balleri, A., Soraghan, J.J.: Micro-Doppler-based in-home aided and unaided walking recognition with multiple radar and sonar systems. IET Radar, Sonar Navigation **11**(1), 107–115 (2017)
13. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3d CNNs retrace the history of 2d CNNs and imagenet? In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6546–6555 (2018)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016)
15. Herath, S., Harandi, M., Porikli, F.: Going deeper into action recognition. Image and Vision Computing **60**, 4–21 (2017)
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
17. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1725–1732. IEEE Computer Society, Washington, DC, USA (2014)
18. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, A., Suleyman, M., Zisserman, A.: The kinetics human action video dataset. CoRR **abs/1705.06950** (2017)
19. Kim, Y., Toomajian, B.: Hand gesture recognition using micro-doppler signatures with convolutional neural network. IEEE Access **4**, 7125–7130 (2016)
20. LeCun, Y., et al.: Generalization and network design strategies. In: Connectionism in perspective, vol. 19. Elsevier (1989)
21. Lee, J., Li, Y.A., Hung, M.H., Huang, S.J.: A fully-integrated 77-GHz FMCW radar transceiver in 65-nm CMOS technology. IEEE Journal of Solid-State Circuits **45**(12), 2746–2756 (2010)
22. Liu, L., Popescu, M., Skubic, M., Rantz, M., Yardibi, T., Cuddihy, P.: Automatic fall detection based on Doppler radar motion signature. In: 5th International Conference on Pervasive Computing Technologies for Healthcare and Workshops, pp. 222–225 (2011)
23. Long, N., Wang, K., Cheng, R., Yang, K., Bai, J.: Fusion of millimeter wave radar and rgb-depth sensors for assisted navigation of the visually impaired. In: Millimetre Wave and Terahertz Sensors and Technology XI, vol. 10800, p. 1080006. International Society for Optics and Photonics (2018)
24. McLaughlin, N., Martinez del Rincon, J., Miller, P.: Recurrent convolutional network for video-based person reidentification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1325–1334 (2016)
25. Ng, J.Y.H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4694–4702 (2015)
26. Pigou, L., van den Oord, A., Dieleman, S., Van Herreweghe, M., Dambre, J.: Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. International Journal of Computer Vision **126**(2), 430–439 (2018)
27. Polfliet, V., Knudde, N., Vandersmissen, B., Couckuyt, I., Dhaene, T.: Structured inference networks using high-dimensional sensors for surveillance purposes. In: Engineering Applications of Neural Networks (EANN), pp. 1–12. Springer International Publishing (2018)

28. Ritchie, M., Fioranelli, F., Borrion, H., Griffiths, H.: Multistatic micro-Doppler radar feature extraction for classification of unloaded/loaded micro-drones. IET Radar, Sonar and Navigation **11**(1), 116–124 (2017)
29. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: IEEE International Conference on Computer Vision (ICCV) (2015)
30. Vandersmissen, B., Knudde, N., Jalalvand, A., Couckuyt, I., Bourdoux, A., De Neve, W., Dhaene, T.: Indoor person identification using a low-power FMCW radar. IEEE Transactions on Geoscience and Remote Sensing **56**(7), 3941–3952 (2018)
31. Varol, G., Laptev, I., Schmid, C.: Long-term temporal convolutions for action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(6), 1510–1517 (2018)
32. Vinod, N., Geoffrey E., H.: Rectified linear units improve restricted boltzmann machines. In: J. Fürnkranz, T. Joachims (eds.) 27th International Conference on Machine Learning (ICML), pp. 807–814. Omnipress (2010)
33. Wang, S., Song, J., Lien, J., Poupyrev, I., Hilliges, O.: Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In: 29th Annual Symposium on User Interface Software and Technology (UIST), pp. 851–860. ACM, New York, NY, USA (2016)
34. Wu, M., Dai, X., Zhang, Y.D., Davidson, B., Amin, M.G., Zhang, J.: Fall detection based on sequential modeling of radar signal time-frequency features. In: IEEE International Conference on Healthcare Informatics (ICHI), pp. 169–174. IEEE Computer Society, Washington, DC, USA (2013)
35. Wu, X., Ren, J., Wu, Y., Shao, J.: Study on target tracking based on vision and radar sensor fusion. Tech. rep., SAE Technical Paper (2018)
36. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European Conference on Computer Vision, pp. 818–833. Springer (2014)
37. Zhang, H.B., Zhang, Y.X., Zhong, B., Lei, Q., Yang, L., Du, J.X., Chen, D.S.: A comprehensive survey of vision-based human action recognition methods. IEEE Sensors **19**(5), 1005 (2019)
38. Zhao, M., Li, T., Abu Alsheikh, M., Tian, Y., Zhao, H., Torralba, A., Katabi, D.: Through-wall human pose estimation using radio signals. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7356–7365 (2018)

## A Indoor Human Activity Recognition on Combined Data Set

In this study, we developed a deep learning approach towards automatic indoor human activity recognition. Moreover, this approach is validated on two separate data sets that are both applicable in a different domain. For the sake of completeness, we explore the efficacy of an integrated system that is capable of predicting the correct activity when dealing with a combined data set of *gestures* and *events*. To that end, both data sets are merged and the *3d-CNN* and *ResCNN* networks are employed for the radar and camera sensors, respectively. The combined data set consists of 3852 samples distributed over 12 different activities. Table 4 lists the total number of samples per activity. Similar to the experiments performed in Sections 7.2 and 7.3, the sample length is set to 2 s or 30 frames.

Table 8 shows the obtained results of both the radar- and video-based model. The results suggest that our developed approach is valid for the combined data set. The radar-based *3d-CNN* achieves 14.40 % and 6.67 % error rate on the cross-validation and random split evaluation approach, respectively. These results are similar to those obtained on the *gestures* data set (c.f., Section 7.2). Similarly, the video-based *ResCNN* network obtains 3.52 % and 2.70 % error rates for $\overline{S}$ and *RS*, respectively.

Furthermore, an experiment is conducted that shows the benefit of fusing both sensors. More precisely, artificially darkened frames (denoted by the $*$ operator) are used as input for the video-based model. This input has a clear negative effect on the error rate of the *ResCNN* network since it degrades by nearly 20 % and 13 % for $\overline{S}$ and *RS*, respectively. However, through the combined use of both sensor-specific networks this effect is not pronounced in the late fusion approach (*Fused**). The performance of this approach only degrades by 2 % in comparison with the use of clean RGB data. Moreover, the fused approach that uses artificially darkened video data still outperforms the radar-only approach by a margin of 2 %.

Table 8: Results for leave-one-subject $S_i$-out cross-validation ($\overline{S}$), with $i \in \{1 \ldots 9\}$, and stratified random split (*RS*) for the combined data set. The *Fused* approach makes use of late fusion of the probabilities of each sensor-specific network. The * operator depicts the use of artificially darkened RGB input frames.

| | Combined Data Set | | | | |
|---|---|---|---|---|---|
| | *Radar* | *Video* | *Fused* | *Video** | *Fused** |
| $S_1$ | 14.87 | 9.54 | 14.35 | 44.97 | 14.96 |
| $S_2$ | 10.59 | 2.38 | 8.78 | 24.96 | 9.82 |
| $S_3$ | 12.98 | 3.73 | 11.56 | 26.13 | 12.44 |
| $S_4$ | 11.55 | 1.00 | 9.47 | 27.87 | 10.85 |
| $S_5$ | 16.28 | 3.35 | 13.71 | 31.39 | 15.97 |
| $S_6$ | 16.63 | 3.45 | 15.02 | 30.30 | 17.57 |
| $S_7$ | 12.79 | 2.02 | 7.37 | 15.53 | 9.98 |
| $S_8$ | 23.64 | 4.02 | 19.39 | 47.75 | 23.40 |
| $S_9$ | 10.99 | 2.20 | 6.96 | 26.01 | 9.71 |
| $\overline{S}$ | 14.40 | 3.52 | 11.46 | 30.54 | 13.86 |
| *RS* | 6.67 | 2.70 | 5.83 | 18.33 | 6.88 |