CNN-based Sensor Fusion Techniques for Multimodal Human Activity Recognition

Sebastian Münzner
Karlsruhe Institute of Technology, Germany
uxcno@student.kit.edu

Philip Schmidt
Robert Bosch GmbH
Corporate Research, Germany
philip.schmidt@de.bosch.com

Attila Reiss
Robert Bosch GmbH
Corporate Research, Germany
attila.reiss@de.bosch.com

Michael Hanselmann
Robert Bosch GmbH
Corporate Research, Germany
michael.hanselmann@de.bosch.com

Rainer Stiefelhagen
Karlsruhe Institute of Technology, Germany
rainer.stiefelhagen@kit.edu

Robert Dürichen
Robert Bosch GmbH
Corporate Research, Germany
robert.duerichen@de.bosch.com

ABSTRACT

Deep learning (DL) methods receive increasing attention within the field of human activity recognition (HAR) due to their success in other machine learning domains. Nonetheless, a direct transfer of these methods is often not possible due to domain specific challenges (e.g. handling of multimodal sensor data, lack of large labeled datasets). In this paper, we address three key aspects for the future development of robust DL methods for HAR: (1) Is it beneficial to apply data specific normalization? (2) How to optimally fuse multimodal sensor data? (3) How robust are these approaches with respect to available training data? We evaluate convolutional neuronal networks (CNNs) on a new large real-world multimodal dataset (RBK) as well as the PAMAP2 dataset. Our results indicate that sensor specific normalization techniques are required. We present a novel pressure specific normalization method which increases the F_1-score by ~ 4.5 percentage points (pp) on the RBK dataset. Further, we show that late- and hybrid fusion techniques are superior compared to early fusion techniques, increasing the F_1-score by up to $3.5\,pp$ (RBK dataset). Finally, our results reveal that in particular CNNs based on a shared filter approach have a smaller dependency on the amount of available training data compared to other fusion techniques.

Author Keywords

Human Activity Recognition, Deep Learning, Sensor Fusion

ACM Classification Keywords

H.1.2, I.5 User/Machine Systems: Pattern Recognition

INTRODUCTION

Regular physical activity has been associated with many health benefits, from maintaining or even enhancing physical fitness to reducing the risk of different diseases. There exist recommendations of how much and what type of physical activity individuals should do [6]. Particularly for elderly populations, practicing and promoting physical activity is important to keep functional independence, as addressed e.g. in the physical activity monitoring for ageing people (PAMAP) platform [3]. A main goal of HAR is to ensure the right quality and quantity of physical activity by continuous monitoring. Recent progress in wearable technology makes small, lightweight, low-cost, multimodal, and accurate sensor units available. Therefore, unobtrusive and mobile activity monitoring has become reasonable.

The recognition of basic physical activities (such as walk or cycle) and postures (such as sit or stand) is well researched, showing that good performance can be achieved even with just one accelerometer and simple machine learning methods [9]. Nevertheless, many practical challenges remain, including: sensor displacement [2], personalisation [16], robustness in real-life scenarios [14], fusion of different sensor modalities [12], and reducing the required amount of labelled training data [19]. The latter is very relevant in HAR as labelling cannot be performed on the raw data (as in e.g. image classification). Labelling requires additional effort (e.g. recording activities with a synchronized video) and is consequently very time consuming. Existing, widely-used datasets (such as the Opportunity dataset [17], the Skoda dataset [22], or the PAMAP2 dataset [15]) are usually relatively small ($\leq$12 subjects), focus on specific user groups (e.g. young researchers), or specific scenarios (e.g. home environment, assembly line). Therefore, the evaluation of new algorithms (such as DL methods) with respect to the amount of required data is of high relevance.

The majority of currently used HAR algorithms is based on standard machine learning classifiers such as (boosted) decision trees [16] or support vector machines [12], in combination with templates or features of time- and frequency-domain. In other research fields such as image and speech classification, these machine learning methods were signifi-

ISWC '17, September 11–15, 2017, Maui, HI, USA

ACM 978-1-4503-5188-1/17/09...$15.00
https://doi.org/10.1145/3123021.3123046

cantly outperformed by DL methods [8]. A key difference of DL methods is that no explicit feature engineering has to be performed as it is part of the optimization process.
Recently, the first papers appeared evaluating the potential of DL techniques for HAR. Zeng *et al.* [23] showed the potential of CNNs and investigated basic parameters (such as max pooling, weight decay, or dropout). They reported an accuracy of up to 88.19% on the Skoda dataset which is $4.41\,\%$ better compared to the best conventional machine learning approach. The results of Hammerla *et al.* [5] indicate an increased accuracy of CNNs for periodic activities such as walking and running, while recurrent neuronal networks (RNNs) outperformed CNNs for short activities such as gestures. Further improvements on the Skoda and Opportunity dataset were achieved by Ordóñez *et al.* [12] by combining CNNs and RNNs layers. The same authors investigated the transfer of learned features between different recognition domains, sensor positions and modalities [13]. They conclude that the training time can be reduced by up to 17% through transfer learning.
In this paper, we investigate three important aspects for the future usage of DL methods in the context of HAR, namely:

1. **Data normalisation:** We evaluate the influence of different normalization techniques and investigate if modality specific techniques are required.
2. **Sensor fusion:** Typical HAR datasets contain multimodal sensor data. This data can be fused at different stages within a DL architecture. We explore the effect of different early and late fusion techniques on the prediction accuracy.
3. **Robustness:** We investigate the classification robustness of different DL models depending on the amount of available training data.

To the best of our knowledge, this is the first paper focusing on these aspects in the context of HAR. As most publicly available datasets are relatively small, we evaluated our experiments on a newly recorded dataset containing 31 subjects. It was acquired in a hospital and covers a wide range of age. For better comparison, the fusion techniques were also evaluated on the PAMAP2 dataset [15]. The paper focuses on CNN-based DL architectures, as both datasets contain primarily continuous activities.

METHODS

Dataset

The dataset was acquired at the Robert Bosch Hospital (RBK) in cooperation with Bosch Healthcare Solutions GmbH and approved by the ethic committee of the University of Tübingen. It includes 31 subjects with a mean age of 62 years (range: 22-87 years; sex: 18 female, 13 male). We refer to this dataset as RBK dataset. Each participant was asked to complete a predefined course consisting of different daily life activities such as walking inside and outside, stair climbing, passive and active sitting (e.g. reading a magazine), or laying. Additionally, the course contained flexible elements, for instance, cutting vegetables, writing a letter, or putting on/off a jacket. The activity walking and stair climbing depends on the physical condition of the subject and was not always possible. The average duration of one recording was $26.7\,min$ ($> 13\,h$ in total). The ground truth was acquired using a simultaneously recorded video. All activities were classified into 7 classes: $C = \{$*walking*, *stair climbing*, *standing*, *sitting*, *laying*, *transitions*, *detached*$\}$. The *transition* class contains all transitions between the postures standing, sitting, and laying. The *detached* class mainly includes periods where the sensor was not attached to the subject (e.g. laying on table) and a few instances with activities not fitting in one of the residual classes.
Data were acquired from six sensor nodes attached at both wrist and ankle positions of each subject as well at the left and right side of the waist. Each node recorded 3D acceleration (ACC) and gyroscope (GYRO) data via a BNO055 sensor (sampled at 100 Hz; ACC: $14\,bit$ resolution, range $\pm 8\,g$; GYRO: $16\,bit$ resolution, range $\pm 2000\ \mathrm{deg}/s^2$). Additionally, air pressure was acquired via a BME280 sensor (sampled at $1\,Hz$, $16\,bit$ resolution, interpolated to $100\,Hz$). Therefore, we have $n_c = 7$ input channels per sensor: $n_{c,P} = 1$ channel pressure, and $n_{c,A} = n_{c,G} = 3$ channels for ACC and for GYRO, respectively. In this study, we use only data of the right wrist sensor node, as it is the most reliable one and as we are focusing on an minimal intrusive setup to increase the practical relevance.
In comparison to publicly available datasets [15, 17, 22], this dataset differentiates through: (1) a larger number of subjects, (2) high diversity with respect to age, and (3) explicit inclusion of time periods when the sensor was not attached to the subject.
We included the PAMAP2 dataset [15] in our evaluation for further comparison. The activities of the protocol subset were grouped in 7 classes (lie, sit, stand, (nordic-)walking, stair climbing, sport (run, cycle, rope jump), house work (vacuum, iron)). Only data of the wrist sensor (ACC, GYRO) and heart rate monitor were used. No pressure data was available.

Class C	*walk*	*stair*	*stand*	*sit*	lay	*transition*	*detached*
% data	20.4	2.1	33.8	10.5	7.1	1.1	25

Table 1. Class distribution and number of available training samples within dataset.

Baseline Algorithm & Evaluation Scheme

As baseline algorithm, we used a Random Forest (RF) classifier with 200 trees, a well known and widely used standard machine learning approach in the HAR community. We evaluated it with once using only time domain features (e.g., mean, max/min, integrals, correlation, ratio, zero-crossings) and once using time and frequency domain features (e.g., peak frequency, short time Fourier transform).
We used a sliding window approach containing $n_w = 128$ samples (approx. $1.3\,s$) as input for the baseline and DL algorithms. The window was shifted by 10 samples. The ground truth of each window was defined by majority vote of all 128 labels within the current window. The total number of available samples is $390k$ training samples. Table 1 shows the distribution of the 7 classes within the dataset. In order to

counter the class imbalance, the amount of samples per activity is equalised. This is done by duplicating samples of underrepresented classes. The same procedure has been applied to the PAMAP2 dataset.
For keeping the variance of the dataset (considering age and gender of subjects), the validation set is composed of at least one male and one female, as well as one young and one elderly person (in total three subjects). The same criteria apply to the test set (in total four subjects). Both validation and test set are kept fixed through the work due to computational reasons.
For the PAMAP2 dataset, the same network parameters (e.g., size of the sliding window) were used. Subject 2 was used as test set; subjects 3 and 4 were used as validation set and remaining subjects as training set.
The macro F_1-score, which is the unweighted mean of the F_1-scores for the different labels, is employed as evaluation metric.

Deep Learning Architecture

The models, presented below are implemented in Theano using Lasagne [4]. A TITAN X GPU, 1392 MHz clock speed and 12 GB RAM, was used for training and testing. We focus on CNN structures in our investigations, as the results of Hammerla *et al.* [5] indicate a high accuracy of CNNs for continuous periodic activities compared to RNN. A CNN is a multi layer architecture that comprises several so-called convolutional layers. The final layer then consist of a fully connected layer followed by a soft-max layer with n_o output neurons. During training, a mapping of the inputs (e.g., time-series data from a gyroscope sensor) to a class label (e.g., "walking") is learned by optimizing the parameters of all layers with respect to a loss function. The output of a layer can be defined as [12]:

$$a_j^{(l+1)} = \sigma\left(b_j^l + \sum_{k=1}^{n_f^l} W_{jk}^l * a_k^l\right), \tag{1}$$

where $a_j^{(l+1)}$ denotes the feature map j in layer $l+1$, σ is a non-linear activation function, n_f^l is the number of convolutional filters in layer l. The weights of the convolution filter and the bias vector are denoted by W_{jk}^l and b_j^l. As we work with multidimensional time-series data, we define W_{jk}^l as a two dimensional matrix of size $\mathbb{R}^{d \times f}$, where d is the number of considered channel dimensions and f is the temporal size of the filter. In our work, we chose a leaky rectifying linear unit (LReLU) [20] as non-linear function σ. At the input level ($l = 1$), the number of filters is $n_f^1 = 1$ and a_1^1 is of size $\mathbb{R}^{n_c \times n_w}$, with n_c being the total number of input channels. The convolution filters are moved by $s_{stride} = 1$.
The fully connected layers consist of n_{fc} neurons which have weighted connections to each previous output neuron. The output is defined via a LReLU transformation function.
In order to obtain a prediction in the end, the last layer needs to transform its output to a probability distribution. This is achieved by the following soft-max operation:

$$\xi(a_j) = \frac{e^{a_j}}{\sum_{c=1}^{C} e^{a_c}} \text{ for } j = 1, \ldots, C. \tag{2}$$

The loss function was chosen to be cross entropy. To allow for an efficient computation, gradients are iteratively calculated per batch using the Adam approach [1] which is known for its fast convergence with learning rate of 0.001. A batch is a random collection of n_b samples.
In a pre-study we investigated different two and three layer network architectures as well as their sensitivity with respect to other network parameters: the learning rate (varied from 0.0001 to 0.1), number of filters n_f^l (16 to 128), filter size f in time dimension (3 - 7), dropout layer [18], stride (1 - 2) and training functions (cross entropy, hinge loss). From this study we choose the two best performing models, namely one with two convolutional layers (2L-CNN: $n_f^1 = n_f^2 = 32$), and one with three layers (3L-CNN: $n_f^1 = 24, n_f^2 = 32, n_f^3 = 64$). The optimal temporal filter size was found to be $f = 3$. The number of considered filter channels d depends on the selected fusion technique. The convolutional layers are followed by a fully connected layer with $n_n = 256$ neurons, dropout ($p = 0.5$), and a soft-max layer with $n_o = C$ nodes at the end. In order to determine convergence, training ended when there was no improvement on the validation set within ten consecutive epochs. The epoch with the highest validation score was chosen.

Normalization Techniques

To investigate the optimal sensor fusion technique, we consider three normalization techniques:
Z-normalization (zNorm): As performed in many machine learning papers [11, 24], the standard normalization technique is z-normalization. The value x_i of channel i is normalized by:

$$x_i' = (x_i - \mu_i)/\sigma_i, \tag{3}$$

where μ_i and σ_i refers to the mean and standard deviation over all available training data for channel i.
Batch normalization (BN): Batch normalization is a normalization technique within the network in form of a layer [7]. The layer is inserted between the linear transformation and the non-linearity unit. These perform z-normalization on the output of the previous layer with the mean μ_B and standard deviation σ_B of the current batch B. Afterwards, the values are scaled by γ and shifted by β. These parameters are learned along with the original model parameters. The output y_j is defined as follows:

$$\hat{x_j} = (x_j - \mu_B)/\sigma_B \tag{4}$$

$$y_j = \gamma \hat{x_j} + \beta. \tag{5}$$

Pressure mean subtraction (PMS): Values of an atmospheric pressure sensor have a large variance due to different weather conditions. To compensate this effect, we investigated a special pressure normalization technique. The idea is to subtract the mean of the current pressure window before

applying a standard z-normalization. This reduces the influence of different weather conditions between various measurement days and enhances short-term changes as they occur during e.g. stair climbing.

Multimodal Sensor Fusion

When dealing with multimodal sensor data in a DL architecture, it is possible to fuse the data at different stages of the network. In principle, it can be distinguished between the fusion of channels within one sensor modality or across multiple sensor modalities. We investigate the four approaches whose number of parameters are shown for the two layer model in Table 2:

Early fusion (EF): This approach fuses all channels of all sensor modalities in the first convolutional layer. The first dimension of the filter size is equal to the number of channels ($d = n_c$). This method was successfully used by Neverova *et al.* [11] and is displayed in Figure 1(a). As all channels are fused into one dimension after the first layer, the number of final parameters to be learned is relatively small (see Table 2), which requires less computation time.

Sensor-based late fusion (SB-LF): In contrast to EF, this approach splits the data by sensor and fuses the data after the convolutional layers with a fully connected layer. This procedure allows to independently tailor a sensor-specific pipeline to the necessities of each modality. So, for instance, for each modality Martinez *et al.* [10] designed a separate pipeline which are fused by a single layer perceptron. We refer to this approach as SB-LF and our implementation is shown in Figure 1(b) with $d_A = n_{c,A}$, $d_G = n_{c,G}$, and $d_P = n_{c,P}$.

Channel-based late fusion (CB-LF): This approach is equal to the previous one with one difference: Instead of dividing the input by sensor, each channel is handled separately ($d = 1$). For example, Zeng *et al.* [23] split the acceleration signal into three input signals, compute features with convolutional layers, and fuse them at a later stage. Therefore, we analogously refer to that design as CB-LF and our layout is outlined in Figure 1(c). This approach results in the highest number of parameters (see Table 2).

Shared filters hybrid fusion (SF-HF): Equivalent to CB-LF, this fusion technique uses filters of dimension $1 \times f$. However, the same filters are used for all input channels. Even though the outputs are not merged explicitly, the filters are shared over all channels, and hence influence each other. This implementation is related to the one of Yang *et al.* [21]. The idea behind this concept is that the same pattern appears across all channels. Our implementation is shown in Figure 1(d). Close inspection reveals the similarity between EF and SF-HF. However, these approaches differ in the filter dimensionality in the first convolutional layer as well as their number of parameters (see Table 2).

Layer	conv1	conv2	FC	Soft-max
Parameters	W^1, b^1	W^2, b^2	W^{fc}, b^{fc}	W^{sm}, b^{sm}
EF	704	4,096	1,016,064	1,799
SB-LF	768	12,288	3,047,680	
CB-LF	896	28,672	7,110,912	
SF-HF	128	4,096	7,110,912	

Table 2. Overview of number of parameters per layer in different sensor fusion techniques, exemplary shown for the optimal two layered CNN architecture with $n_f = 32$ in both convolutional layers (denoted as conv1 and conv2) and $n_n = 256$ in the fully connected layer (FC).

RESULTS

Data Normalization

First, we address the effect of different normalization techniques on the F_1-score. We investigate four techniques: zNorm only and combinations of zNorm with either BN, PMS, or BN and PMS.

Figure 2 displays the F_1-scores (and standard deviations) for the investigated normalization techniques on the RBK dataset. The bar plots were obtained by averaging the performance of the different two and three layered fusion models (EF, SB-LF, CB-LF, SF-HF) for a given normalization. In order to reduce the influence of the random initialized weights each model was trained five times.

Figure 2 indicates that using zNorm only or in combination with BN results in an average F_1-score of $\sim 60.7\,\%$ with a standard deviation over $\geq 1.8\,pp$. The average F_1-score increases by $4.0\,pp$ (and standard deviation decreases below $< 1.5\,pp$), using either zNorm+PMS or zNorm+BN+PMS.

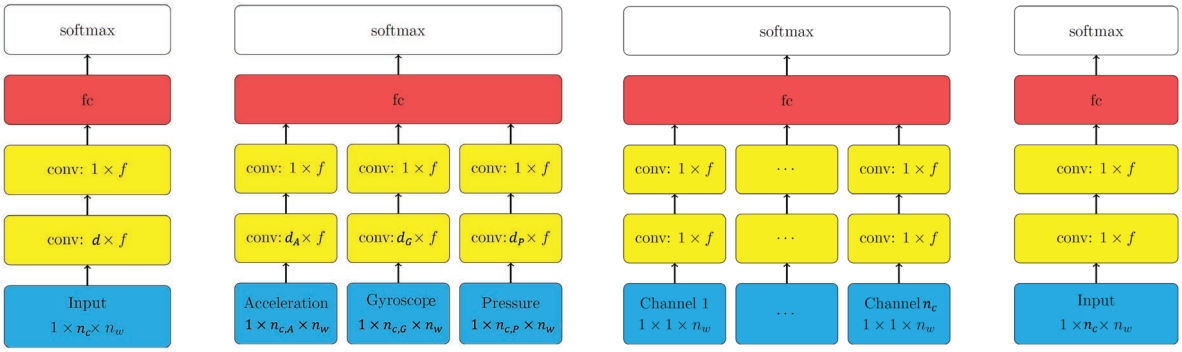
Sensor Fusion

As shown in Figure 2, applying zNorm+PMS with or without additional BN leads on average to the highest F_1-scores. Consequently, we focus on these two normalization techniques to further investigate the performance of the different fusion models (EF, SB-LF, CB-LF and SF-HF). For the different fusion techniques, Figure 3 and Figure 4 show the mean F_1-scores with and without BN, respectively. Each model was trained five times. As baseline, the F_1-scores of RF classifiers using either time (RF(T)) or time and frequency domain (RF(TF)) features are shown in the plots, too.

Looking at Figure 3, it becomes apparent that all CNNs outperform the RF(T) by at least $1.5\,pp$. In case of RF(TF), the deeper three layered CNNs have a similar performance. Training deeper models also increases the F_1-scores by $1.0\,pp$ on average compared to the two layered CNNs. However, the three layered models have on average a slightly increased standard deviation. Figure 3 indicates that SB-LF, CB-LF, and SF-HF achieve comparable good average F_1-scores for three layered CNNs.

In Figure 4, the average F_1-scores achieved by the models trained with BN are displayed. Again, all CNNs outperform the RF(T) baseline algorithm and three layered CNNs perform on average better than the two layered CNNs (except for EF). Further, the three layered SF-HF outperforms on average the RF(TF) baseline algorithm by $\approx 1pp$. In contrast to Figure 3, there is a clear difference between the fusion techniques visible (especially for the three layered CNNs). Using a three layered hybrid SF-HF fusion model with BN leads to the highest average F_1-score of $67.5\,\%$.

Similar tendencies can be observed when evaluating the different CNN fusion models on the PAMAP2 dataset (Table 3). On average SF-HF outperforms the residual techniques. In



(a) Early fusion (b) Sensor-based late fusion (c) Channel-based late fusion (d) Shared filters hybrid fusion

Figure 1. Schematic illustration of the four investigated sensor fusion techniques. Networks consist of two convolutional (conv) layers with filter size f in time dimension, one fully connected (fc), and a soft-max layer.

case of 2L-CNN, the average F_1-score of SF-HF is $11.6pp$ higher compared to EF.

Robustness

Dependency on amount of training data

To evaluate the robustness of different sensor fusion techniques, first, we investigate the influence of the amount of available training data on the classification outcome. For this purpose, CNNs were trained on only 12 - 75 % of the training data on the RBK dataset. The training data was split subject-wise into subsets corresponding to either 12 % (only for the evaluation of CNN models with 12 % labeled data) or 25 % (for the evaluation of CNN models with $\geq$ 25 % labeled data) of the original data. The single subsets and all different combinations of them were evaluated.

Exemplary, the average F_1-score for a three layered CNN model with SF-HF and zNorm+PMS+BN normalization is shown in Figure 5. Similar trends could be observed

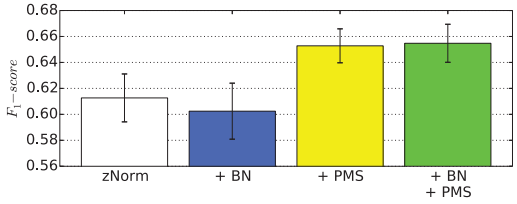


Figure 2. Average F_1-score and standard deviation for different normalization techniques on the RBK dataset: z-normalization only (zNorm) or in combination with either BN, PMS, or BN and PMS. Results are averaged over two and three layered CNNs and different sensor fusion techniques.

Fusion technique		EF	SB-LF	CB-LF	SF-HF
2L-CNN	mean	0.74	0.76	0.81	**0.86**
	std	0.013	0.043	0.045	0.034
3L-CNN	mean	0.74	0.81	0.81	**0.85**
	std	0.016	0.052	0.05	0.029

Table 3. Average F_1-scores achieved on PAMAP2 for different sensor fusion models (EF, SB-LF, CB-LF and SF-HF) with two and three layered CNNs using zNorm+BN normalization.

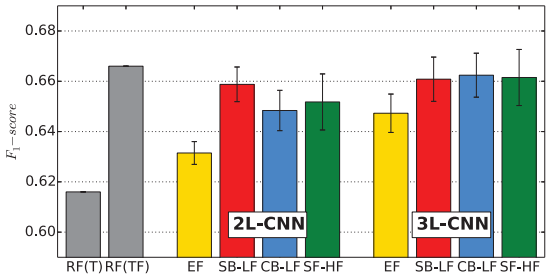


Figure 3. Average F_1-scores of different sensor fusion models for two and three layered CNNs using zNorm+PMS normalization on the RBK dataset. Whiskers indicate standard deviation due to five times repetition. RF shown for comparison.

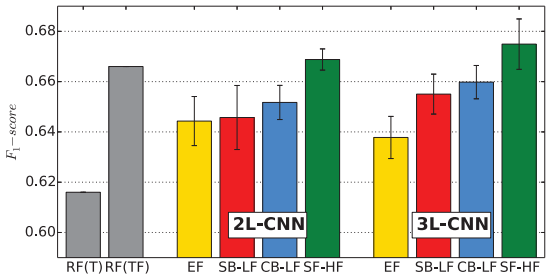


Figure 4. Average F_1-scores of different sensor fusion models for two and three layered CNNs using zNorm+PMS+BN normalization on the RBK dataset. Whiskers indicate standard deviation due to five times repetition. RF shown for comparison.

for the two layered models and for other normalization techniques. Depending on the amount of labeled data, the average F_1-score decreases on average by $10.0\,pp$ for CNN models, by $15.9\,pp$ for RF(T), and by $17.6pp$ for RF(TF). In general, the F_1-score of the CNNs is on average between $4 - 10\,pp$ higher than the one obtained by the RF(T). All CNNs outperform RF(TF) for evaluations with $\leq$ 25% labeled data. The individual CNN fusion models perform quite similar (deviation in F_1-score $< 4\,pp$). Overall, the SF-HF model achieved the highest F_1-score, regardless of the amount of training samples.

Dependency on randomly initialized weights

To investigate the influence of the initial weights independently on the different training folds, we selected the folds resulting in the highest F_1-score for each investigated subset. The evaluation was performed for the models CB-LF without BN and SF-HF with BN. Each model was trained five times on the RBK dataset. Figure 6 displays the F_1-scores depending on the amount of labeled data in a box plot. The results indicate that the standard deviation increases for decreasing amount of available data, in particular for datasets with $< 75\,\%$ of data. For SF-HF with BN, the range between the upper and lower whiskers increases from $0.9\,pp$ using $100\,\%$ of labeled training data to $8.9\,pp$ using $12\,\%$ of the data.

DISCUSSION

The presented F_1-scores for the newly acquired RBK dataset are lower than the results reported on publicly available datasets [15, 17, 22]. The reasons for this are manifold. First, the primarily used dataset has a higher variability due to a larger number of participants, a very diverse age profile (the oldest participant was 87 years), and the flexible elements in the dataset (e.g., cutting vegetables). Second, our evaluation focuses on the use of a single (wrist) sensor instead of multiple sensors distributed across the body. The main motivation for this was that we wanted to use a minimal intrusive setup. Using also the information of the other sensors should further improve the results as it would lead to a more complete picture of what task the subject is performing. Third, the ground truth included not only standard daily living activities (e.g., walking) but also unusual labels like *detached* and *transitions*.

Figure 7 shows the confusion matrices for a RF classifier and the CNN models CB-LF without BN and SF-HF approach utilizing BN. It becomes obvious that in particular the class *transition* is difficult to classify (F_1-score is $\leq 43\,\%$), which results in a lower overall F_1-score. In contrast to the other activity classes, transitions between e.g. sitting and standing are relatively short and less frequent (see Table 1). Hence only very few representative number of training samples are generated for this class. Moreover, intraclass variance for the transition labels are large as they can be performed in different ways (e.g., with or without the use of an armrest). The results of [5] indicate that such short term events might be better predicted by recurrent neural networks (RNNs). The residual activity classes such as *walking* or *laying* have, partly, very high F_1-scores (Figure 7) which are comparable to other publications.

On average the mean F_1-scores of the PAMAP2 dataset are higher compared to the results of the elderly care dataset, which was to be expacted as primarily data of young people was acquired. In contrast, doing multiple repetitions reveals a higher standard deviation especially for late and hybrid fusion techniques for the PAMAP2 dataset. Reasons are the lower number of available training data as well as lower number of training subjects.

In our investigations, first, we focused on the effect of using different normalization techniques. Our results indicated that using sensor specific normalization techniques, in particular for the pressure sensor, are crucial also for DL methods. The physical motivation for this procedure is that the data was recorded on different days (with different weather conditions). The pressure changes caused by different activities are relatively small and are consequently overlapped and partly covered by the pressure variations due to weather changes. Therefore, an individual preprocessing step had to be done to emphasize local changes in a given window. These findings can be seen as a motivation to further investigate sensor specific normalization. Especially in the broader context of HAR, if e.g. the dataset contains beside inertial sensors other modalities such as photoplethysmography or skin conductivity the normalization becomes very relevant.

Next, we compared different early and late fusion techniques and trained models of different depth. Although the deeper models had a slightly larger standard deviation than the shallower ones, their average performance was better (on the RBK dataset). This increased standard deviation may be explained by the larger amount of parameters of the deeper models giving rise to different local minima. In terms of fusion techniques the late and hybrid fusion models clearly outperformed EF models. This can be explained by reconsider-

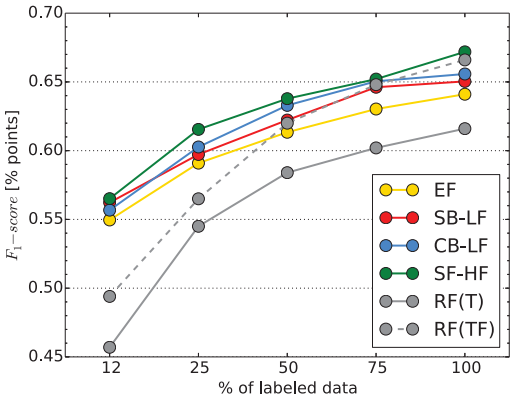


Figure 5. Average F_1-scores on the RBK dataset using different training data folds of three layer CNNs incorporating BN for different sensor fusion techniques (and baseline method) depending on the amount of available training data.

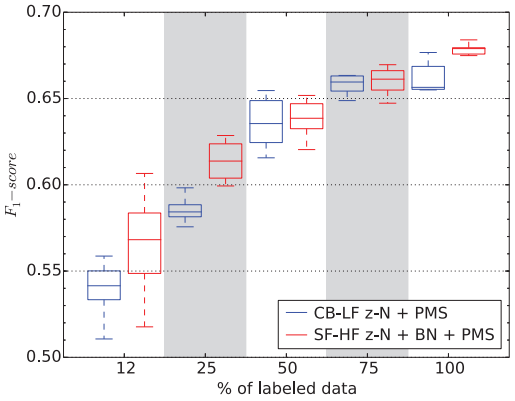
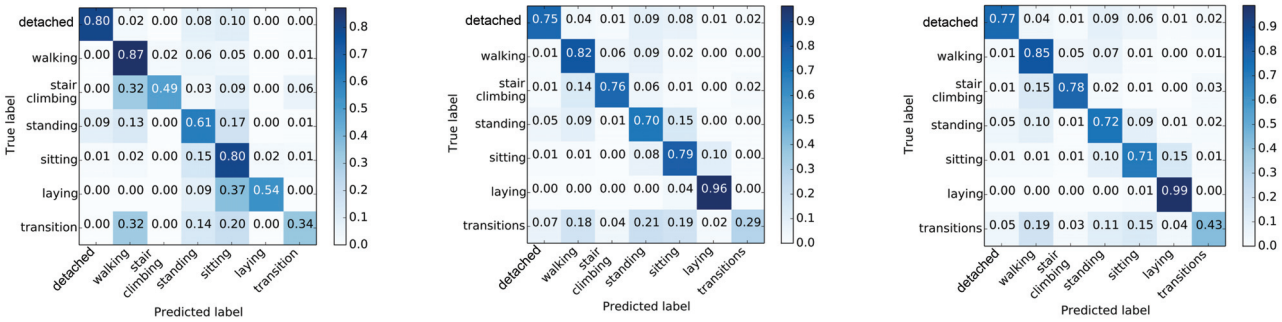


Figure 6. Comparison of F_1-scores of multiple initialized CB-LF without BN and SF-HF with BN models depending on the amount of available training data on the RBK dataset.



(a) Random Forest with time and frequency features

(b) Channel-based late fusion without BN

(c) Shared filters hybrid fusion with BN

Figure 7. Confusion matrix for different trained models (RBK dataset)

ing their architecture. CB-LF models are based on the idea to learn convolutional filters which are tailored specifically for each sensor modality or channel. Metaphorically speaking, each branch of the CNN becomes an expert for its inputs and extracts the most descriptive features. These optimal features are then fused by the fully connected layer giving rise to the good F_1-score. The SF-HF approach, which relies on the assumption that similar patterns appear across multiple channels, performs well, especially when BN layers are used. In contrast, the EF model combines all sensor inputs in the first layer. This early sensor fusion seems to hinder the model from learning long-ranged interactions. In addition, the EF model holds the smallest number of parameters which might pose limitations on its expressive abilities.

Our findings are supported by the results achieved on the public available PAMAP2 dataset, where SF-HF is on average superior compared to the residual fusion techniques (Table 3). Similar tendencies can be observed when comparing the results of Zeng *et al.* [23] (using a CB-LF model) and Yang *et al.* [21] (using a SF-HF model). They evaluated their models on the Opportunity dataset [17], which contains also more complex activities.

When comparing late and hybrid fusion techniques, Figure 3 and 4 reveal large differences due to BN (especially for 3L-CNNs). In case of not using BN, the average F_1-score of SF-HF is comparable to SB-LF and CB-LF. Applying BN, the average F_1-score of SF-HF increases by $1.5\,pp$. This effect can be explained as BN layers reduce the internal covariate shift. The SF-HF approach uses the same filters for all channels. However, as the channels have values within different ranges, this can result in a shift of the channel based mean and variance. Applying BN normalizes the feature maps of each sensor channel to zero mean and unit variance, which allows us to apply the same filters to all channels. This hypothesis is supported as the average F_1-score of CB-LF is not effected by the use of BN. The performance boost of the CNNs incorporating BN layers comes at the expense that these models require roughly a factor 1.5 more training time per epoch.

In case of SB-LF, the regularization effect of BN can be observed as it was pointed out by Ioffe *et al.* [7]. Compared to CB-LF and SF-HF, SB-LF has the lowest number of parameters. Using BN has a regularization effect on the models, which results in a decrease of the F_1-score of SB-LF models.

The amount of training data was found to have a strong effect on the performance (Figure 5). Interesting to note is that the relation between F_1-score and training data follows roughly a $log(x)$ rule. For the models trained with only a small fraction of the training data, the standard deviation between the performance of the individual runs is increased. This was to be expected as e.g. $12\,\%$ of labelled data means that only data of three subjects was used. Comparing the decrease of the F_1-score for the RF and CNN models reveals that the dependency of RF on the amount of available data is stronger. This shows that CNN models are also capable of learning meaningful representations with low amount of data. In Figure 6, we investigate the robustness of CB-LF and SF-HF models with respect to the initial selected weights. The box plots indicate that the variance increases with decreasing amount of data, in particular for SF-HF. If only few training data is available, we recommend the training of multiple SF-HF models and selection of the best model based on the performance on the validation dataset. An alternative approach is transfer learning which can be used to reduce the influence on randomly initialised weights. Ordóñez *et al.* [13] demonstrated how weights can be pretrained on data from different locations or domains and, later, only be fine-tuned on the target dataset.

Deep learning methods are known to be quite resource hungry and so far it is often hard to meet their computational demands on embedded devices. However, with the recent technological advances in the field of tensor processing units (TPUs) we expect the expect mobile devices to handle deep networks in the near future.

CONCLUSION & OUTLOOK

In this paper, we addressed the problems of normalization and fusion of mutimodal sensor HAR data. Our results show that sensor specific normalization increases the prediction accuracy of the CNNs. We presented a pressure specific normalization technique which increased the average F_1-score by ~ 4.5 pp in case of the RBK dataset. In the context of multimodal HAR, further normalization techniques should be investigated which focus on other modalities such as physiological sensors.

Comparing different sensor fusion techniques revealed that late and hybrid fusion techniques outperform early fusion

techniques. The best performing fusion technique was the SF-HF model, achieving the highest F_1-score on both the RBK and the PAMAP2 dataset. As discussed, the optimal fusion of a HAR setup using multiple sensors (e.g. wrist and waist sensors) has to be investigated further.

ACKNOWLEDGMENTS

We thank Prof. Dr. med. Becker, Dr. U. Lindemann, Dr. J. Klenk, and L. Schwickert of the geriatric department of the Robert Bosch Hospital, and Bosch Healthcare Solutions for the planning, acquisition and labeling of the RBK dataset.

REFERENCES

1. Ba, J., and Kingma, D. Adam: A method for stochastic optimization, 2015.
2. Baños, O., Damas, M., Pomares, H., et al. A benchmark dataset to evaluate sensor displacement in activity recognition. In *Proceedings of 14th Int. Conference on Ubiquitous Computing (UbiComp)* (2012), 1026–1035.
3. Bleser, G., Steffen, D., Reiss, A., Weber, M., Hendeby, G., and Fradet, L. *Personalized Physical Activity Monitoring Using Wearable Sensors*. LNCS. 2015, 99–124.
4. Dieleman, S., Schlter, J., Raffel, C., et al. Lasagne: First release., Aug. 2015.
5. Hammerla, N. Y., Halloran, S., and Ploetz, T. Deep, convolutional, and recurrent models for human activity recognition using wearables. In *Proceedings of 25th Int. Joint Conference on Artificial Intelligence* (2016), 1533–1540.
6. Haskell, W. L., Lee, et al. Physical activity and public health: Updated recommendation for adults from the American College of Sports Medicine and the American Heart Association. *Medicine and Science in Sports and Exercise 39*, 8 (Aug. 2007), 1423–34.
7. Ioffe, S., and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR abs/1502.03167* (2015).
8. Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., 2012, 1097–1105.
9. Lee, M.-h., Kim, J., Kim, K., et al. Physical activity recognition using a single tri-axis accelerometer. In *Proceedings of World Congress on Engineering and Computer Science (WCECS)* (2009).
10. Martinez, H. P., Bengio, Y., and Yannakakis, G. N. Learning deep physiological models of affect. *IEEE Computational Intelligence Magazine 8*, 2 (2013), 20–33.
11. Neverova, N., Wolf, C., Lacey, G., et al. Learning human identity from motion patterns. *arXiv preprint arXiv:1511.03908* (2015).
12. Ordóñez Morales, F. J., and Roggen, D. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors 16*, 1 (2016), 115.
13. Ordóñez Morales, F. J., and Roggen, D. Deep convolutional feature transfer across mobile activity recognition domains, sensor modalities and locations. *Proceedings of IEEE 20th Int. Symposium on Wearable Computers (ISWC)* (2016).
14. Reiss, A., Hendeby, G., and Stricker, D. A novel confidence-based multiclass boosting algorithm for mobile physical activity monitoring. *Personal and Ubiquitous Computing 19*, 1 (2015), 105–21.
15. Reiss, A., and Stricker, D. Introducing a new benchmarked dataset for activity monitoring. In *Proceedings of IEEE 16th Int. Symposium on Wearable Computers (ISWC)* (2012), 108–109.
16. Reiss, A., and Stricker, D. Personalized mobile physical activity recognition. In *Proceedings of IEEE 17th Int. Symposium on Wearable Computers* (2013).
17. Roggen, D., Calatroni, A., Rossi, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *Proceedings of 7th Int. Conference on Networked Sensing Systems (INSS)* (2010), 233–240.
18. Srivastava, N., Hinton, G., Krizhevsky, A., et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15* (2014), 1929–1958.
19. Stikic, M., van Laerhoven, K., and Schiele, B. Exploring semi-supervised and active learning for activity recognition. In *Proceedings of IEEE 12th Int. Symposium on Wearable Computers* (2008), 81–88.
20. Xu, B., Wang, N., Chen, T., and Li, M. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853* (2015).
21. Yang, J. B., Nguyen, M. N., San, P. P., et al. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Proceedings of the 24th Int. Joint Conference on Artificial Intelligence* (2015), 25–31.
22. Zappi, P., Lombriser, C., Stiefmeier, T., et al. Activity recognition from on-body sensors: Accuracy-power trade-off by dynamic sensor selection. In *Wireless Sensor Networks*, vol. 4913. Springer Berlin Heidelberg, 17–33.
23. Zeng, M., Nguyen, L. T., Yu, B., et al. Convolutional neural networks for human activity recognition using mobile sensors. In *Proceedings of 6th Int. Conference on Mobile Computing, Applications and Services (MobiCASE)*, IEEE (2014), 197–205.
24. Zheng, Y., Liu, Q., Chen, E., et al. Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. *Frontiers of Computer Science 10*, 1 (2016), 96–112.