

Human Activity Recognition using Attribute-Based Neural Networks and Context Information

Stefan Lüdtke^{1*}, Fernando Moya Rueda^{2*}, Waqas Ahmed¹, Gernot A. Fink² and Thomas Kirste¹

¹Institute of Visual & Analytic Computing, University of Rostock, Germany

²Department of Computer Science, TU Dortmund University, Germany

{stefan.luedtke2, waqas.ahmed, thomas.kirste}@uni-rostock.de,
{fernando.moya, gernot.fink}@tu-dortmund.de

Abstract

We consider human activity recognition (HAR) from wearable sensor data in manual-work processes, like warehouse order-picking. Such structured domains can often be partitioned into distinct *process steps*, e.g., packaging or transporting. Each process step can have a different prior distribution over *activity classes*, e.g., standing or walking, and different system dynamics.

Here, we show how such context information can be integrated systematically into a deep neural network-based HAR system. Specifically, we propose a hybrid architecture that combines a deep neural network—that estimates high-level movement descriptors, *attributes*, from the raw-sensor data—and a shallow classifier, which predicts activity classes from the estimated attributes and (optional) context information, like the currently executed process step.

We empirically show that our proposed architecture increases HAR performance, compared to state-of-the-art methods. Additionally, we show that HAR performance can be further increased when information about process steps is incorporated, even when that information is only partially correct.

1 Introduction

The accurate recognition of human activities from sensor data is an important task for many applications, like healthcare, sports, or for developing situation-aware assistive technologies. As another example, activity recognition is relevant for the analysis and optimization of manual-work processes, like packaging in a warehouse, see Fig. 1, the main motivation for our work [Reining *et al.*, 2018b; Niemann *et al.*, 2020bl].

Human Activity Recognition (HAR) from movement data is challenging due to the intra- and inter-subject variability of human movement. Recently, deep neural networks have been used successfully for multichannel time-series HAR. These networks combine the feature extraction and classification in an end-to-end approach [Ordóñez and Roggen, 2016; Zeng *et al.*, 2014; Ronao and Cho, 2015; Hammerla *et al.*, 2016; Yao *et al.*, 2018].

*These authors contributed equally to the paper.

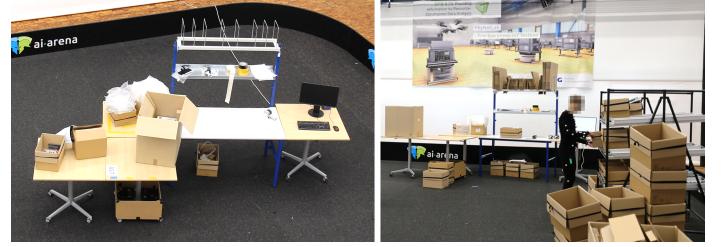


Figure 1: Physical laboratory set-up of a warehouse packaging work station.

They learn non-linear and temporal relations of basic, complex, and highly dynamic human movements directly from raw-inertial sensor data. These transformations are more discriminative with respect to human action classes than hand-crafted features [Zeng *et al.*, 2014; Hammerla *et al.*, 2016].

However, such deep neural networks are usually purely data-driven—they cannot directly make use of prior knowledge that is often available in highly structured application domains. For example, manual work processes, like order picking or packaging, are structured into distinct high-level *process steps*, as shown in Fig. 4. Each of these steps has a unique prior distribution over activity classes and unique system dynamics. For example, a person is more likely to perform a *handling* activity class when retrieving articles in the warehouse (step 3 in Fig. 4) than when moving the cart (step 2 in Fig. 4). Furthermore, information about the currently executed process step is often available directly from external sources. For example, in a packaging scenario, each article and empty box are scanned before being packed, according to a shipment list. This scanning process, which is recognized in the warehouse management system, indicates a transition between process steps.

In this paper, we show how such additional *context information* can be integrated systematically into a deep neural network-based HAR system. Specifically, we propose a hybrid architecture that allows integrating context information without re-training the neural network from scratch: We use a deep neural network that extracts high-level movement descriptors (*attributes*, like posture or usage of left or right hand) from raw wearable-sensor data. On top, a shallow classifier estimates activity classes from the

attributes and optionally from the context information—like the currently executed process step, in our case. This way, we can use a deep neural network independently of available context information and domain specifics. Instead, we only need to adapt the final classifier to different domains, i.e., different process models or changes in the available context information.

We evaluate this architecture on a HAR task in intra-logistics, and empirically show that it achieves increased HAR performance, compared to the state of the art, even when no context information is available. Additionally, the HAR performance increases further when context information in the form of the currently executed process step is available—even when it is just partially correct.

2 Related Work

Methods of statistical pattern recognition are common for analyzing human movements from measurements from on-body devices. A standard pipeline involves pre-processing, segmentation, hand-crafted feature extraction, and classification. Recently, deep learning methods become the standard method for solving HAR problems in gesture recognition and activities of daily living [Ordóñez and Roggen, 2016; Grzeszick *et al.*, 2017; Hammerla *et al.*, 2016; Ronao and Cho, 2015; Yao *et al.*, 2018]. Deep architectures combine the feature extraction and classification in an end-to-end approach. These architectures learn the non-linear and temporal relations of basic, complex, and highly dynamic human movements. They learn non-linear transformations directly from raw-inertial data. Compared to the hand-crafted ones, these transformations are more discriminative to human actions and invariant to distortions and temporal translations [Hammerla *et al.*, 2016]. The authors in [Ronao and Cho, 2015] introduced temporal convolutional neural networks (tCNNs), which carry out convolution and pooling operations along the time axis. These architectures share rather small convolution filters among all the sensors, as local temporal measurements are correlated independently of the sensor type; this is also valid assuming a normalization per sensor. An architecture that combines temporal convolutions and recurrent networks is introduced in [Ordóñez and Roggen, 2016]. Specifically, the authors used Long Short-Term Memory units (LSTMs), recurrent units with memory cells, and a gating system [Hochreiter and Schmidhuber, 1997]. The authors in [Hammerla *et al.*, 2016] utilized shallow recurrent networks: a three-layered LSTM and a one-layered bidirectional LSTM. The authors in [Grzeszick *et al.*, 2017] created a tCNN that adapts to the wearables per human limb. This architecture contains parallel-convolutional branches per human limb. Each convolutional branch creates a deep representation of the measurements per human limb. The network deploys a late fusion for creating a final deep representation. The authors in [Qian *et al.*, 2019] proposed an end-to-end architecture that extract temporal, body-relations and statistical features.

Nonetheless, the performance of these deep learning methods did not show a significant increase compared to other areas such as image and video classification or segmentation [Hammerla *et al.*, 2016; Grzeszick *et al.*, 2017]. HAR remains a challenging task due to large intra- and inter-class variability of human move-

ments, i.e., humans carry out similar tasks differently. In addition, there is a broad range of human activities or movements, and there is not a standard definition nor structure for formulating a clear problem of HAR [Bulling *et al.*, 2014]. Likewise, datasets for HAR suffer from the class-unbalance problem, where the number of samples per action class differs strongly [Ordóñez and Roggen, 2016].

Attribute representations help solve zero-shot learning and transfer learning. High-level attribute representations are semantic descriptions that describe categories, e.g., in object and scene recognition problems [Cheng *et al.*, 2013; Lampert *et al.*, 2014; Zheng *et al.*, 2017] and words in document analysis. In object recognition, attributes can be the shape, color, texture, size of objects, or even geographic information. Different approaches have introduced semantic descriptions of activities as representations for solving HAR [Cheng *et al.*, 2013; Arif Ul Alam and Roy, 2017; Zheng *et al.*, 2017]. In [Cheng *et al.*, 2013], the authors proposed to use semantic attributes for recognizing unseen activities. In [Zheng *et al.*, 2017], human-annotated attributes and data-driven attributes are combined for solving HAR in sports videos. They selected a subset from both attribute groups maximizing the discrimination capability of attributes for distinguishing different sets of classes. The authors in [Arif Ul Alam and Roy, 2017] designed a hierarchical representation of human-activity taxonomy based on semantic descriptions in the context of smart-home applications.

Precisely, attribute-based representations have been deeply explored on HAR in the manual order picking process in [Reining *et al.*, 2018b]. Attribute representations are beneficial for dealing with the versatility of activities. The authors in [Reining *et al.*, 2018b] compared the performance of deep architectures trained using different attribute representations, evaluating their quantitative performance and quality from the perspective of practical application. The authors in [Reining *et al.*, 2018c] tested different attribute representations, expert-given and random, for solving zero-shot learning in HAR. Unseen activities were described using attributes that are shared with the seen activities. Expert-given attribute representations performed better than a random one. The latter is created following the conclusions in [Moya Rueda and Fink, 2018]. A semantic relation between attributes and activities enhances HAR not only quantitatively with regards to performance but also guarantees a transfer of the attributes between activities by domain experts. In this preliminary work, the mapping between activity classes and attribute representations was one-to-one. Fig. 2 presents an example of attribute-based representation for HAR.

Deep neural networks do not consider domain knowledge about the causal relation of activities that is often available in highly structured domains. To overcome this limitation, a combination of symbolic reasoning methods (Computational State-Space Models, CSSM [Krüger *et al.*, 2014]) and deep neural networks is introduced in [Moya Rueda *et al.*, 2019]. The CSSM models prior knowledge about the high-level, causal structure of the domain, and the deep neural network acts as observation model, relating the sensor data to CSSM states.

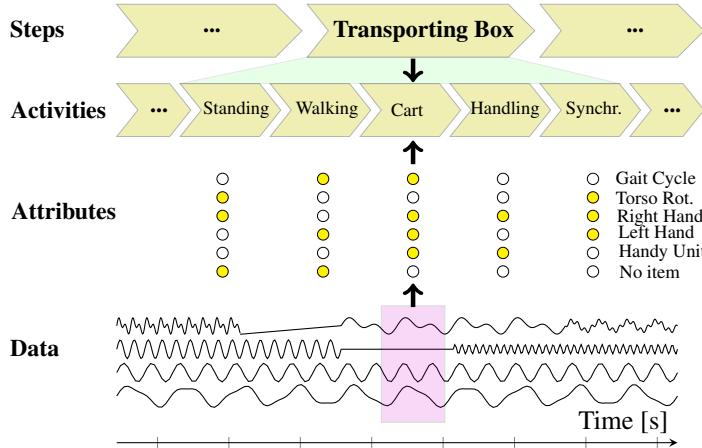


Figure 2: Relationship between attributes, process steps and activity classes. First, attributes are estimated from the raw sensor data. Activities are then classified based on these attributes and the current process step.

3 Methods

We propose a hybrid HAR method, consisting of a deep neural network that predicts movement descriptors (attributes) from the sensor data, and a model that predicts activity classes from the attribute estimates.

In the following, we first present the architecture of the neural network in more detail. Afterwards, we discuss different options for representing and learning the relationship between attributes and activity classes. Finally, we show how context information can be integrated into the classification system.

3.1 Attribute-based Deep Neural Network for HAR

We deploy the temporal convolutional neural network (tCNN) from [Yang *et al.*, 2015; Grzeszick *et al.*, 2017]. It has shown to perform relatively well despite its simplicity [Grzeszick *et al.*, 2017; Moya Rueda and Fink, 2018]. A tCNN is an end-to-end architecture composed of feature extractors and a classifier, either a softmax or a sigmoid. The architecture processes sequences of size $[T, W]$, with T the sequence length and W the number of sequence channels. The tCNN contains four convolutional layers, no downsampling, and two fully connected layers, and a classifier. The convolutional layers are composed of 64 filters of size $[5 \times 1]$, performing convolutions along the time axis. The first and second fully connected layers contain 128 units. Depending on the task, the tCNN will have a softmax layer for activity classification or a sigmoid layer for attribute classification [Moya Rueda and Fink, 2018]. Here, a sigmoid activation function replaces the usual softmax layer, as we use the tCNN_{attribute} to compute an attribute representation from an input sequence, rather than directly classifying it, following [Moya Rueda and Fink, 2018]. Using attribute representations have shown to be beneficial for HAR, following the conclusions in [Moya Rueda and Fink, 2018; Niemann *et al.*, 2020b]. The tCNN_{attribute} is trained using the binary-cross entropy loss. Fig. 2 presents the relation between

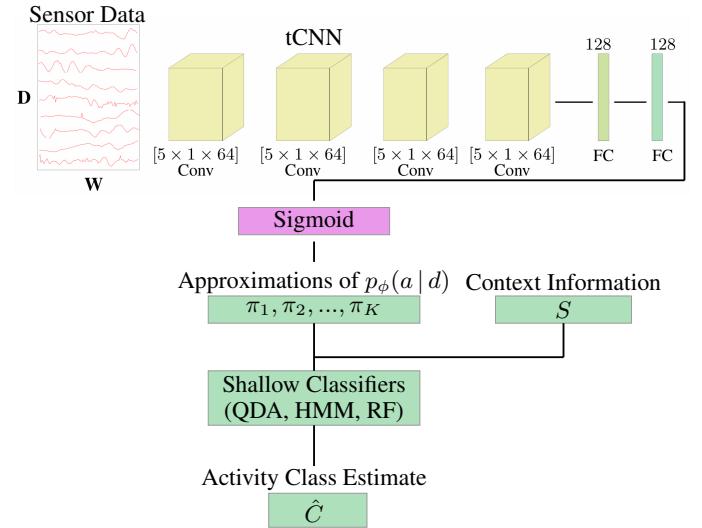


Figure 3: Proposed hybrid activity recognition architecture. The neural network provides probabilities π_i that each of the attributes a_i is present. The probability vector π and the context information (e.g. the current process step) is used as input of a shallow classifier that estimates an activity class \hat{c} .

activities, attributes, and sequence input, and Fig. 3 shows the tCNN_{attribute} architecture.

Due to the final sigmoid layer, the tCNN_{attribute} output can be interpreted as the probabilities for each attribute being present or not present in the input segment. More formally, the tCNN_{attribute} is a function $\phi : D \rightarrow \Pi$, where $d \in D$ is a data sample, i.e., segment, and $\pi \in \Pi$ is a parameter vector for the posterior distribution over attributes $p_\phi(a | d)$. Specifically, the probability of a binary attribute vector $a = (a_1, \dots, a_K)$ is given by a product of Bernoulli distributions:

$$p_\phi(a | d) = \prod_{k=1}^K p(a_k | d) = \prod_{k=1}^K \pi_k^{a_k} (1 - \pi_k)^{1-a_k} \quad (1)$$

3.2 Human Activity Recognition from Semantic Attributes

From an application viewpoint, we are interested in the activity class c instead of the attributes themselves. In general, the maximum-likelihood estimate \hat{c} of the activity class is given by

$$\hat{c} = \operatorname{argmax}_c p(c | d) \quad (2)$$

We assume that the activity class c and the sensor data d are conditionally independent, given the attribute vector a , so that the activity class posterior can be written as

$$p(c | d) = \sum_{a \in A} p(c | a) p_\phi(a | d). \quad (3)$$

Here, $p_\phi(a | d)$ is given directly by the tCNN ϕ , as shown in Section 3.1 (different options for modeling $p(c | a)$ are discussed below). Thus, in principle, an estimate of the activity class \hat{c} for a given data segment d is obtained in two steps:

(i) Compute π via the neural network ϕ : $\pi = \phi(d)$

(ii) Compute Eq. 2 with these values π

In the following, we present two options for modeling $p(c|a)$: The *Direct Attribute Prediction* (DAP), proposed by [Lampert *et al.*, 2014], and the *Generalized DAP*, that we formally introduce here for the first time, but that was implicitly used by [Moya Rueda and Fink, 2018; Niemann *et al.*, 2020b] before in an approximate form. Finally, we describe a novel idea that does not explicitly use the probabilistic model above, but uses a classifier to model the dependency between network outputs π and activity classes c .

Direct Attribute Prediction The *Direct Attribute Prediction* (DAP) model [Lampert *et al.*, 2014] assumes that each class c has an associated unique attribute representation $a^{(c)}$, i.e. there is a deterministic relationship $f(c) = a^{(c)}$ so that $p(a|c) = \mathbb{1}(a = a^{(c)})$. By making use of Bayes' theorem, one can rewrite $p(c|a) = \frac{p(c)}{p(a)} p(a|c)$. Inserting back into Eq. 3 gives

$$p(c|d) = \sum_{a \in A} \frac{p(c)}{p(a)} p(a|c) p_\phi(a|d) \quad (4)$$

The normalization factor $p(a)$ is constant w.r.t. the class. Furthermore, [Lampert *et al.*, 2014] propose to use a uniform class prior $p(c)$. Thus, the factor $\frac{p(c)}{p(a)}$ can be ignored for classification, so that the expression simplifies to

$$p(c|d) \propto \sum_{a \in A} \mathbb{1}(a = a^{(c)}) p_\phi(a|d) \quad (5)$$

$$= p_\phi(a^{(c)}|d), \quad (6)$$

where $a^{(c)} = f(c)$ is the unique attribute representation of class c .

Generalized Direct Attribute Prediction Unfortunately, in HAR, activity classes usually do not have a unique attribute representation. For example, the activity class *take* can involve the attribute *left hand* and/or *right hand*. Thus, the DAP model cannot be used directly.

However, a deterministic relationship in the opposite direction often exists in HAR, i.e., each attribute vector corresponds to exactly one activity class. This mapping $g(a) = c$ can either be defined in advance from prior domain knowledge, or estimated from training data.

In either way, the distribution becomes $p(c|a) = \mathbb{1}(c = g(a))$, which can be inserted into Eq. 2 and 3. In this case, the maximum-likelihood estimate is computed as

$$\hat{c} = \operatorname{argmax}_c \sum_{\{a \in A \mid g(a)=c\}} p_\phi(a|d) \quad (7)$$

We call this model *Generalized Direct Attribute Prediction* (GDAP).

However, computing the sum in Eq. 7 exactly is only feasible as long as the number of attribute vectors $|A|$ is small. To make classification feasible even when $|A|$ is large, [Moya Rueda and

Fink, 2018; Niemann *et al.*, 2020b] propose to compute the sum not exactly, but to compute the maximum-likelihood approximation¹ $\hat{a} = \operatorname{argmax}_a p_\phi(a|d)$, and return $\hat{c} = g(\hat{a})$. That is, their approximation assumes that the sum in Eq. 7 is dominated by its largest term, similar to the approximation done in [Ramírez and Geffner, 2010] for goal recognition.

Classification-based methods The former methods either require to sum over all attribute vectors (which quickly becomes infeasible, as the number of possible attribute vectors grows exponentially with the number K of attributes), or need to make strong assumptions that might induce a large error.

Therefore, we propose a novel approach to obtain activity classes, shown in Fig. 3. The key insight is to view the neural network outputs π_1, \dots, π_n not as parameters of $p_\phi(a|d)$, but as high-level *features* or *latent representations* that can be used as input of a classifier that learns the function $c = h(\pi)$.

In this case, an estimate of the activity class \hat{c} is for a given data segment d obtained in two steps: (i) Compute the neural network output $\pi = \phi(d)$; (ii) Compute the activity class estimate $\hat{c} = h(\pi)$. Specifically, we use Quadratic Discriminant Analysis (QDA), a Hidden Markov Model with Gaussian observation model (HMM) and Random Forests (RF) as classifiers.

In addition to the potentially higher computational efficiency during classification, this concept has two key advantages over the probabilistic models introduced before:

- It allows to account for bias in the neural network, i.e., the shallow classifier can produce a correct classification even when the network output π does not assign a high probability to attribute vectors associated with the true class.
- Integrating process step information as an additional feature of the classifier becomes straightforward, and does not require re-training of the complete neural network when additional process information becomes available, as discussed next.

3.3 Making use of Process Knowledge

In highly structured domains like manual-work processes, the overall activity sequence can often be separated into distinct *process steps*, see Fig. 2. As a concrete example, consider the Business Process Model (BPM) in Figure 4 that describes a simple order-picking process in a warehouse. At each point in time, the subject is performing exactly one of these process steps.

Here, we assume that an estimate of the current process step is available. For example, in the warehouse scenario, the worker needs to scan each article before putting it inside a box for further transportation or packing. These scanning events, thus, allow recognizing process-step transitions accurately. In other domains, information about process transitions might not be available directly, but can be estimated by causal models, like Computational State-Space Models [Krüger *et al.*, 2014].

In principle, such process step information could be used directly as additional input to the attribute network ϕ . However,

¹More precisely, [Moya Rueda and Fink, 2018; Niemann *et al.*, 2020b] do not compute $\hat{a} = \operatorname{argmax}_a p_\phi(a|d)$, but $\hat{a} = \operatorname{argmin}_a \|a - \pi\|_2$ after normalizing a and π , hence we call this model *nearest neighbor approximation*.

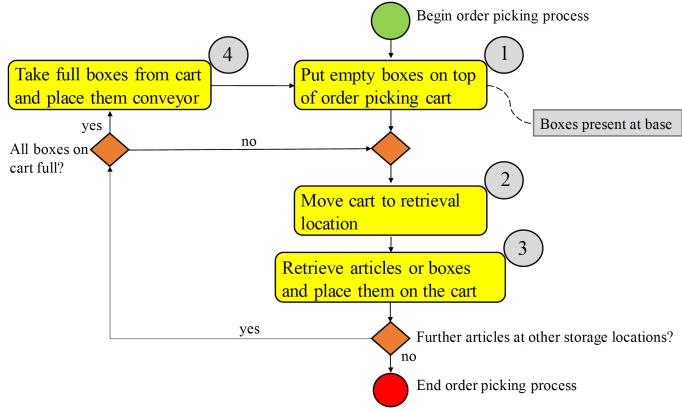


Figure 4: The Business Process Model (BPM) for the logistics scenario.

BPM state	Stand	Walk	Cart	Handle (up)	Handle (center)	Handle (down)	Synch.
S0	0.266	0.375	0.000	0.047	0.195	0.031	0.086
S1	0.000	0.053	0.000	0.342	0.474	0.132	0.000
S2	0.011	0.000	0.757	0.006	0.215	0.011	0.000
S3	0.061	0.000	0.000	0.239	0.534	0.166	0.000
S4	0.000	0.095	0.000	0.167	0.500	0.238	0.000

Table 1: Distribution of activity classes for each BPM state.

this would require re-training of the network, which can be computationally very costly. Additionally, the original training data still need to be available to allow re-training, which might not be the case when using a pre-trained network as a black-box model. Furthermore, we assume that the data-attribute dependency that is modeled by the neural network is independent of the process step—how the sensor observations relate to poses and movements of the subject, i.e., the attributes, does not change when the subject is executing different sub-tasks.

Instead, we assume that only the relationship between *attributes and activity classes* depends on the process step. Therefore, we propose to include the process step information into the shallow classifier that predicts activity classes c from network outputs π . More specifically, we make use of the process knowledge in the different classifiers in the following ways:

- In the Random Forest, we add the process step as an additional input feature.
- For the QDA, we estimate different priors $p_s(c)$ for each process step s , but use a single, shared Gaussian likelihood $p(\pi | c)$, because the class priors differ substantially in each process step (see Table 1), but the amount of data is insufficient to estimate the parameters of individual likelihoods for each process step.
- For the HMM, we train different transition models $p_s(c_t | c_{t-1})$ for each process step s , to account for the different system dynamics of the different process steps. Similar to the QDA above, the model uses a shared Gaussian observation likelihood $p(\pi | c)$.

Classifier	No Process Info		With Process Info	
	Acc.	F1	Acc.	F1
GDAP	0.636	0.628	—	—
NN	0.621	0.614	—	—
QDA	0.670	0.650	0.721	0.711
HMM	0.685	0.672	0.719	0.708
RF	0.688	0.674	0.710	0.715

Table 2: Experimental results, best results are printed in boldface. For each model, we compared a baseline case where no process information is available and the case where the correct BPM state at each time step. For GDAP and NN, information about process states cannot be integrated directly. GDAP: Generalized Direct Attribute Prediction; NN: Nearest neighbor-approximation of GDAP; QDA: Quadratic Discriminant Analysis; HMM: Hidden Markov Model; RF: Random Forest.

4 Experimental Evaluation

4.1 Dataset: LARA

The Logistic Activity Recognition Challenge (LARA) dataset [Niemann *et al.*, 2020b; Niemann *et al.*, 2020a] is used to evaluate our approach. The dataset recreates three warehousing scenarios in a constrained environment, ensuring natural motion and resemblance to reality [Reining *et al.*, 2018a]. This dataset contains measurements of a marker-based MoCap system, called LARA-MoCap, and on-body devices, called LARA-OB, from 14 humans performing activities in the intra-logistics. LARA provides recordings from the 3D joint-poses and the 3D linear and angular acceleration of subjects performing eight activities. Joint poses are recorded with a rate of 200 Hz, and linear and angular accelerations with 100 Hz. The activities are common activities in the intra-logistics: *Standing*, *Walking*, *Moving Cart*, *Handling (upwards)*, *Handling (centred)*, *Handling (downwards)*, *Synchronization* and *None*. Here, we experimented with LARA-MoCap. For LARA-MoCap, there are 22 joints. For each joint, LARA provides the 3D pose. These are all centered with respect to the lower back of a subject. In general, LARA provides 714 min of annotated recordings being a large annotated dataset for HAR.

The LARA dataset consists of non-overlapping training, validation, and testing sets. For LARA-MoCap, the validation and testing sets contain recordings from subjects [5, 11, 12]. The training set contains recordings from the other eight subjects of LARA-MoCap. Our experiments have used the test set, which contains annotations of process steps from one of the three warehouse scenarios.

4.2 Results

We compared the different methods, introduced in Section 3.2), for obtaining activity classes from the tCNN_{attribute} output. Specifically, we compared our GDAP model, the nearest-neighbor (NN) approximation of GDAP proposed by [Niemann *et al.*, 2020b], and three classification-based methods (QDA, HMM and RF). All experiments were performed in R [R Core Team, 2020], and we used the randomForest [Liaw and Wiener, 2002] with 500 trees for fitting RFs.

We used a pre-trained tCNN for attribute prediction for all experiments, with the architecture described in Section 3.1. The pre-

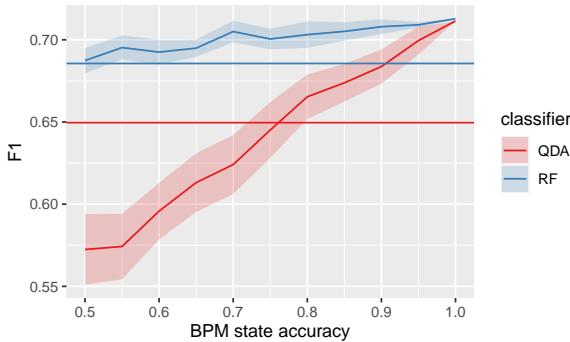


Figure 5: F1 score w.r.t. accuracy of BPM state prediction. Horizontal lines indicate the baseline performance without BPM state information. Each experiment has been repeated 10 times, the ribbons indicate standard deviation of the F1 score. The RF is more robust regarding inaccurate state information than the QDA.

trained tCNN can be found in [Niemann *et al.*, 2020a]. The network was trained on the full LARA-MoCap dataset, as described in [Niemann *et al.*, 2020b]. Furthermore, for the classification-based methods, we investigated the case where the correct BPM state (see Fig. 4) is provided to the models for each time step, see Section 3.3 for how this feature is utilized in the different classifiers.

Table 2 shows the HAR performance of the different methods. When the currently performed BPM state is not provided to the models, the classification-based methods (QDA, HMM, RF) all outperform the probabilistic models (GDAP, NN). Of the classification-based methods, the RF achieves highest accuracy (0.668) and F1 score (0.674) – higher than prior results by [Niemann *et al.*, 2020b] (NN, with 0.621 accuracy and 0.621 F1 score).

To assess the information contained in the BPM state, we first computed the mutual information $I(C, S) = H(C) - H(C|S)$ of activity classes c and BPM states s (where $H(C)$ the marginal entropy of activity classes and $H(C|S)$ is the conditional entropy): On the test data, we obtain $H(S) = 2.44$ bit and $H(C|S) = 1.55$ bit. Thus, knowledge of the BPM states provides $I(C, S) = 0.89$ bit of additional information for classification. This finding is reflected in the classification performance: The performance of all models increases when they are provided with the correct BPM state at each timestep (see Table 2). Again, all three classifiers show similar performance, with RF having the highest F1 score (0.715), and QDA having the highest accuracy (0.721).

Next, we investigated how inaccuracies in the provided BPM state influence HAR performance. Specifically, we introduced inaccuracies into the BPM state feature as follows: With probability p , each BPM state s_t is replaced by a different state, chosen uniformly among the remaining states $S \setminus s_t$. Thus, the noisy state sequence has an accuracy of $1 - p$. We varied the parameter p between 0, i.e., perfectly accurate state information, and 0.5, in steps of 0.05, and investigated how this affected the F1 score of the HAR models.

Fig. 5 shows the results for this experiment. For both the QDA and the RF, the F1 score decreases when the BPM state estimate

becomes less accurate. However, the RF is more robust in that regard: Even when the accuracy of the BPM state estimate is only 50%, the RF is not worse than the baseline RF that does not make use of the BPM state at all – in contrast to the QDA, which requires a state accuracy of at least 80% to outperform the QDA baseline.

In summary, the results indicate that providing the shallow classifiers with context information in the form of the currently performed BPM state increases the overall HAR performance, even when the BPM state is not always accurate.

5 Discussion and Conclusion

In this paper, we proposed a HAR system that combines a deep neural network and a shallow classifier. The system predicts activity classes from the neural network output, i.e., from the approximations of the posterior distribution over attributes. Our empirical evaluation on a HAR task in logistics shows that this combined method can increase HAR performance, compared to previous methods [Niemann *et al.*, 2020b]. Furthermore, this system allows integrating additional context information (like an estimate of the current process step, as used here) directly, without re-training the neural network. We showed that providing an estimate of the current process step as an additional feature can increase HAR performance, even when the process-step estimation is not always correct.

However, these results are preliminary: Our evaluation is based on only a subset of the complete LARA dataset, as process step annotations were only available for one of the three scenarios contained in the dataset. Furthermore, we just simulated the noise on the process step estimate. Future work needs to confirm our findings for a more realistic case where the process step is estimated from external sources, as well as on additional HAR datasets. On a technical level, future work will investigate using an additional neural network layer for predicting activity classes from attributes instead of using a shallow classifier, thus allowing end-to-end-training of the complete HAR system.

Depending on the sensor setup and the specific domain, additional context information, other than the current process step, might be available, e.g., currently handled objects or the location of the subjects. Our proposed system architecture allows using such features directly in a systematic way. Investigating such cases is another topic for future work.

Finally, the BPM can be seen as a form of prior domain knowledge that we aim to exploit. So far, we only made use of the fact that different BPM states have different prior probabilities of activities, but the BPM also provides information about the *dynamics* of states and action. In the future, we will explore how such information can serve to generate symbolic precondition-effects models of the system dynamics, e.g., Computational State-Space Models [Krüger *et al.*, 2014].

References

- [Arif Ul Alam and Roy, 2017] Mohammad Arif Ul Alam and Nirmalya Roy. Unseen Activity Recognitions: A Hierarchical Active Transfer Learning Approach. pages 436–446. IEEE, 2017.

- [Bulling *et al.*, 2014] Andreas Bulling, Ulf Blanke, and Bernt Schiele. A Tutorial on Human Activity Recognition Using Body-worn Inertial Sensors. *ACM Comput. Surv.*, 46(3):33:1–33:33, January 2014.
- [Cheng *et al.*, 2013] Heng-Tze Cheng, Martin Griss, Paul Davis, Jianguo Li, and Di You. Towards zero-shot learning for human activity recognition using semantic attribute sequence model. In *Proc. of the 2013 ACM Int. Joint Conf. on Pervasive and Ubiquitous Computing*, UbiComp ’13, page 355–358, New York, NY, USA, 2013. Association for Computing Machinery.
- [Grzeszick *et al.*, 2017] Rene Grzeszick, Jan Marius Lenk, Fernando Moya Rueda, Gernot A Fink, Sascha Feldhorst, and Michael ten Hompel. Deep neural network based human activity recognition for the order picking process. In *iWOAR*. ACM, 2017.
- [Hammerla *et al.*, 2016] Nils Y. Hammerla, Shane Halloran, and Thomas Ploetz. Deep, Convolutional, and Recurrent Models for Human Activity Recognition using Wearables. *arXiv:1604.08880 [cs, stat]*, April 2016. arXiv: 1604.08880.
- [Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-term Memory. *Neural computation*, 9:1735–80, December 1997.
- [Krüger *et al.*, 2014] Frank Krüger, Martin Nyolt, Kristina Yordanova, Albert Hein, and Thomas Kirste. Computational state space models for activity and intention recognition. a feasibility study. *PloS one*, 9(11):e109381, 2014.
- [Lampert *et al.*, 2014] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-Based Classification for Zero-Shot Visual Object Categorization. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, volume 36 of 3, pages 453–465. IEEE, 2014.
- [Liaw and Wiener, 2002] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.
- [Moya Rueda and Fink, 2018] Fernando Moya Rueda and Gernot A Fink. Learning attribute representation for human activity recognition. In *2018 24th Int. Conf. on Pattern Recognition (ICPR)*, pages 523–528. IEEE, 2018.
- [Moya Rueda *et al.*, 2019] Fernando Moya Rueda, Stefan Lüdtke, Max Schröder, Kristina Yordanova, Thomas Kirste, and Gernot A. Fink. Combining symbolic reasoning and deep learning for human activity recognition. In *15th Workshop CoMoRea*. in press, 2019.
- [Niemann *et al.*, 2020a] Friedrich Niemann, Christopher Reining, Fernando Moya Rueda, Erik Altermann, Nilah Ravi Nair, Janine Anika Steffens, Gernot A. Fink, and Michael ten Hompel. Logistic Activity Recognition Challenge (LARA) – A Motion Capture and Inertial Measurement Dataset, 2020.
- [Niemann *et al.*, 2020b] Friedrich Niemann, Christopher Reining, Fernando Moya Rueda, Nilah Ravi Nair, Janine Anika Steffens, Gernot A. Fink, and Michael ten Hompel. LARA: Creating a Dataset for Human Activity Recognition in Logistics Using Semantic Attributes. *MDPI Sensors*, 2020.
- [Ordóñez and Roggen, 2016] Francisco Ordóñez and Daniel Roggen. Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. *Sensors*, 16(1):115, January 2016.
- [Qian *et al.*, 2019] Hangwei Qian, Sinno Jialin Pan, Bingshui Da, and Chunyan Miao. A novel distribution-embedded neural network for sensor-based activity recognition. In *IJCAI*, pages 5614–5620, 2019.
- [R Core Team, 2020] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.
- [Ramírez and Geffner, 2010] Miguel Ramírez and Hector Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In *Proc. of the AAAI Conf. on Artificial Intelligence*, volume 24, 2010.
- [Reining *et al.*, 2018a] Christopher Reining, Fernando Moya Rueda, Michael Ten Hompel, and Gernot A. Fink. Towards a Framework for Semi-Automated Annotation of Human Order Picking Activities Using Motion Capturing. In *FedCSIS*, pages 817–821. IEEE, 2018.
- [Reining *et al.*, 2018b] Christopher Reining, Fernando Moya Rueda, Michael Ten Hampel, and Gernot A Fink. Towards a framework for semi-automated annotation of human order picking activities using motion capturing. In *2018 FedCSIS*, pages 817–821. IEEE, 2018.
- [Reining *et al.*, 2018c] Christopher Reining, Michelle Schlangen, Leon Hissmann, Michael ten Hompel, Fernando Moya, and Gernot A Fink. Attribute representation for human activity recognition of manual order picking activities. In *Proc. of the 5th iWOAR*, pages 1–10, 2018.
- [Ronao and Cho, 2015] Charissa Ann Ronao and Sung-Bae Cho. Deep Convolutional Neural Networks for Human Activity Recognition with Smartphone Sensors. In Sabri Arik, Tingwen Huang, Weng Kin Lai, and Qingshan Liu, editors, *Neural Information Processing*, Lecture Notes in Computer Science, pages 46–53. Springer Int. Publishing, 2015.
- [Yang *et al.*, 2015] Jianbo Yang, Minh Nhut Nguyen, Phyto Phyto San, Xiaoli Li, and Shonali Krishnaswamy. Deep convolutional neural networks on multichannel time series for human activity recognition. In *IJCAI*, pages 3995–4001, 2015.
- [Yao *et al.*, 2018] Rui Yao, Guosheng Lin, Qinfeng Shi, and Damith C. Ranasinghe. Efficient dense labelling of human activity sequences from wearables using fully convolutional networks. *Pattern Recognition*, 78:252–266, June 2018.
- [Zeng *et al.*, 2014] Ming Zeng, Le T. Nguyen, Bo Yu, Mengshoel Ole J., Jiang Zhu, Pang Wu, and Joy Zhang. Convolutional neural networks for human activity recognition using mobile sensors. In *MobiCASE*, pages 197–205. IEEE, 2014.
- [Zheng *et al.*, 2017] Jingjing Zheng, Zhuolin Jiang, and Rama Chellappa. Submodular attribute selection for visual recognition. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, volume 39 of 11, pages 2242–2255. IEEE, 2017.