PROCESS MODEL-GUIDED POST-FILTERING FOR WEARABLE ACTIVITY RECOGNITION IN ERGONOMIC WORK PROCESSES

DOI
https://doi.org/10.18690/um.fov.4.2026.48

ISBN
978-961-299-152-4

ANDREAS EMRICH,[1] ALESSA WEIN,[1]
JANAKI VISWANATHAN,[2] MICHAEL FREY,[3] PETER LOOS[1, 4]

[1] German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany
andreas.emrich@dfki.de, alessa.wein@dfki.de, peter.loos@dfki.de
[2] Department of Language Science & Technology, Saarland University, Germany
jviswanathan@lsv.uni-saarland.de
[3] TRON – Translationale Onkologie an der Universitätsmedizin der Johannes Gutenberg-Universität Mainz gemeinnützige GmbH, Mainz, Germany
michael.frey@tron-mainz.de
[4] Saarland University, Saarbrücken, Germany
peter.loos@dfki.de

Ergonomic assessment of manual work processes requires continuous recognition of workers' physical activities – a task complicated by privacy constraints, sensor complexity, and noisy movement data. This paper proposes a process model-guided post-filtering approach that leverages existing process knowledge to refine the output of wearable activity classifiers, regardless of the underlying recognition method. Evaluated across two industry-derived use cases using body area networks with 3, 5, and 7 sensors, the approach yields dramatic accuracy improvements – from below 50 % unfiltered to over 82 % with as few as three sensors. Results demonstrate that explicit process model knowledge can substantially compensate for reduced sensor setups, lowering hardware costs and privacy risks without sacrificing recognition quality. This proof-of-concept establishes process model filtering as a promising component in ergonomic monitoring pipelines for structured manual work.

Keywords:
process model filtering,
human activity recognition,
wearable sensing,
body area network,
ergonomic assessment,
work process monitoring

University of Maribor Press

1 Introduction

Ergonomic assessment in the workplace aims to improve worker health and sustainably enhance productivity and performance. In this context, activity detection for personalized recommendations can help play a key role in preventing injuries and reducing physical strain. While many industrial processes are already automated and optimized using machine data for process mining, a considerable share of work is still performed manually by human operators.

Process mining has emerged as a powerful technique for modeling, analyzing, and improving business processes. In automated environments, processes are typically IT-enabled and equipped with sensors, allowing for the generation of structured system logs. These logs provide a direct source of event data, which serves as a foundation for process mining and performance optimization. For manual human tasks it is not as straightforward.

1.1 Problem Statement

Wearable activity recognition supports health self-monitoring and prevention, providing early warnings or accident alerts (Assenza et al., 2020). However, applying process mining to human-centered work presents significant challenges. Unlike machines, human activities are complex and do not naturally generate the structured, semantic event logs central to classical process mining. Instead, human-generated data is often noisy and heterogeneous, requiring extensive preprocessing to approximate event structures.

Furthermore, designing reliable detection algorithms and selecting sensors are non-trivial tasks. While expressive sensors increase accuracy, they also raise costs, complexity, and training effort. Privacy is an additional concern, as movement data can identify workers and assess performance, which is a particular sensitive issue in healthcare (cf. Munoz-Gama et al., 2022). Consequently, following the EU-GDPR principle of data minimization is essential (cf. GDPR (Regulation (EU) 2016/679)). Ultimately, human-centered process mining deviates from "happy path" scenarios, necessitating novel approaches to data acquisition, preparation, and interpretation.

1.2 Goal

This paper proposes a rule-based post-processing approach that leverages existing process models to refine activity recognition results. The underlying Human Activity Recognition (HAR) classifier is treated as an exchangeable component – the contribution lies in the filtering stage and its integration with process model knowledge. To validate this, two industrial use cases are evaluated across three sensor configurations within a Body Area Network (BAN), demonstrating that filtering substantially compensates for reduced sensor setups.

1.3 Methodology & Structure

Following the Design Science Research (DSR) methodology, the goal is to develop an IT artifact that solves a real-world problem. Different sensor setups are compared using the "Design as a search process" guideline by Hevner et al. (2004). In accordance with Peffer's DSR process (cf. Peffers et al., 2007), the problem at hand is analyzed in this introduction and in Section 2. A motivational use case subsequently serves to refine the solution's objectives (cf. Section 3). Sections 4 and 5 explicate the artifact's design, detailing the system architecture and the specific activity detection methods. Finally, Section 6 demonstrates and evaluates the approach via a proof-of-concept implementation (cf. Nunamaker et al., 2015) comparing the detection accuracy across unfiltered and filtered results for the three sensor setups.

2 Related Work

This section identifies research gaps in (human) activity recognition, sensor-based process mining, and model-based approaches.

Human Activity Recognition (HAR) automatically classifies activities using data from video, accelerometer, or wearables (cf. Lara & Labrador, 2013). Applications range from healthcare to worker safety. For example, Faramondi et al. (2019) used waist-mounted IoT sensors to detect anomalous, potentially hazardous situations for workers. Recent advancements are driven by large datasets and deep learning (cf. Chen et al., 2021). While some focus on simple smartphone sensors (cf. Wan et al., 2020), others utilize graph convolutional networks (GCNs) on skeleton data (cf.

Peng et al., 2019). Context-aware HAR further incorporates human-object interactions via directed semantic graphs, often using graph attention networks or edge convolutions (cf. Corona et al., 2020). Recent ergonomics research also explores low-cost setups without strict process notions (cf. Demrozi et al., 2023; Raso et al., 2019). Crucially, none of these approaches utilize a dedicated process model to interlink activities in a temporal-logical order.

Sensor-based Process Mining: Predictive Process Monitoring aims to predict the current/next steps or outcomes of a process instance (cf. Francescomarino et al., 2018). While many methods rely on atomic process logs, IoT-based logs must be derived from complex, continuous sensor data. Although sensor fusion (cf. Rebmann et al., 2019) and specific aggregation steps (cf. De Leoni & Pellattiero, 2022) improve mining capabilities, uncertainty remains high. Knoch et al. (2018) demonstrated sensor-based monitoring for manual work, yet process model knowledge is rarely used to mitigate this uncertainty or enhance data quality.

Hybrid and Model-based Process Mining approaches: Some predictive monitoring approaches utilize a-priori instance knowledge (cf. Francescomarino et al., 2018), but the specific benefit of using process models for this purpose has not been systematically compared or explicitly stated. Notably, existing literature lacks approaches that leverage the process model to filter incorrect machine learning results from detection algorithms.

3 Motivating Use Case

The presented use case involves an ergonomics assessment at a large European OEM with highly standardized procedures and implicit process models. To address the physical strain of daily work, this paper identifies bottlenecks and problematic activities. This foundation enables future work to provide notifications and process recommendations to prevent ergonomically unfavorable postures.

3.1 Background: Upper-Extremity Workplace Ergonomics

Daily loads and physical stress on the upper extremities significantly impact the prevalence of work-related musculoskeletal disorders (WMSD) (cf. Schaefer et al., 2007). The Ergonomic Assessment Worksheet (EAWS) was developed as an expert screening procedure to assess physical stress by tracking static postures, movements, action forces, manual loads, and repetitive tasks (cf. Schaub et al., 2012).

Based on stress degree and duration, a score-based system determines if movements remain within recommended boundaries. Project partners developed a BAN-based automatic ergonomic assessment using EAWS. While not detailed here, validating these process mining methods helps locate ergonomic bottlenecks. This enables operative guidance to prevent harmful situations, long-term advice to mitigate unavoidable awkward postures (cf. Sundstrup et al., 2020), and the identification of workplace redesign opportunities.

3.2 Use Case 1 – Pre-control of Injection Rails

The first use case (cf. Figure 1) describes a process in injection rail production for the automotive OEMs. As Figure 1 indicates, the rails are being delivered to the left and right side of the user. In a time-fixed cycle the user performs his regular actions which include picking up the rail from either the left or right container, turning to the inspection table, and placing the rail into the inspection holder. There, the rail is being checked for defects. If the inspection is successful, the user lifts the rail above some obstacle into a basin of water, where the rail is being washed and transported to its next destination.

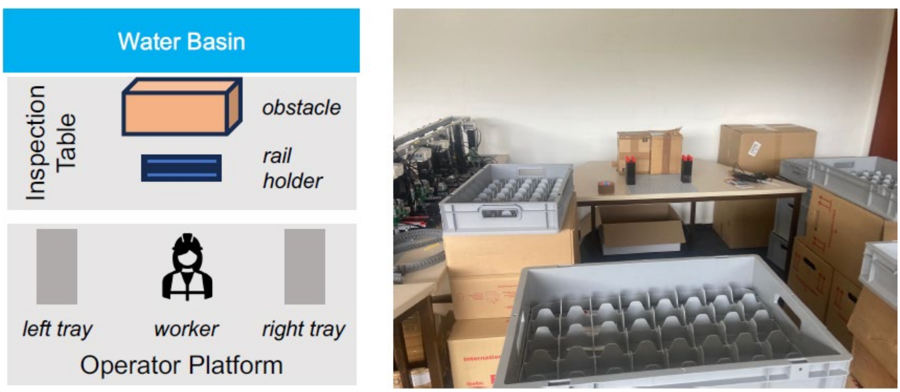

Figure 1: Use Case 1
Source: Own

3.3 Use Case 2 – Final Control and Packing of Rails

In Use Case 2, the rail is undergoing a final quality check before it is packed into boxes for delivery (cf. Figure 2). Once again, the delivery boxes are aligned left and right of the user. From a central conveyor belt, the user receives a rail that is being taken for inspection. To do so, the rail is put onto a holder. Now, the injection holes are closed with plastic caps, and the rail is being inspected with a magnifying glass. These two steps can be carried out in either order. Then the rail is taken again and deposited into one of the boxes.

Figure 2: Use Case 2
Source: Own

3.4 Relevance of the Use Cases for the Investigation

Although the use case processes are mostly linear, they expose some degrees of freedom which make classical HAR approaches hard to apply: In contrast to UC1, UC2 includes situations where both movement-based machine learning and process models may prove insufficient for accurate activity detection.

4 System Design

To capture the manual work processes described in Section 3, the system must balance detection accuracy against practical constraints such as sensor complexity, privacy, and worker comfort. The design therefore rests on two interdependent pillars: a body area network (BAN) that acquires upper-body movement data, and a process-aware software architecture that transforms raw sensor streams into activity predictions. The following sections detail both the hardware sensor setup and the software components, with particular attention to how process model knowledge is embedded into the detection pipeline.

4.1 Hardware & Sensor Setup

The sensor suit consists of seven sensors of the type IMU (inertial measurement units) that are attached to the upper body and extremities via a sensor suit. The sensors as enumerated in the figure are: (0) for the lower back (root sensor), (1) left shoulder, (2) right shoulder, (3) left upper arm, (4) left lower arm, (5) right upper arm, (6) right lower arm. Each sensor is a digital signal processor (DSP) that returns three-dimensional angle data. The sensors are connected via cables to a multiplexing unit which is controlled by an Arduino board, that compresses the single sensor data into timed data frames. The Arduino board is connected to a Raspberry Pi 4 which contains all required local software capabilities of the body area network (BAN) (cf. section 4.2). The Raspberry Pi is connected to a power pack and a WiFi antenna to maintain connectivity to the main server.

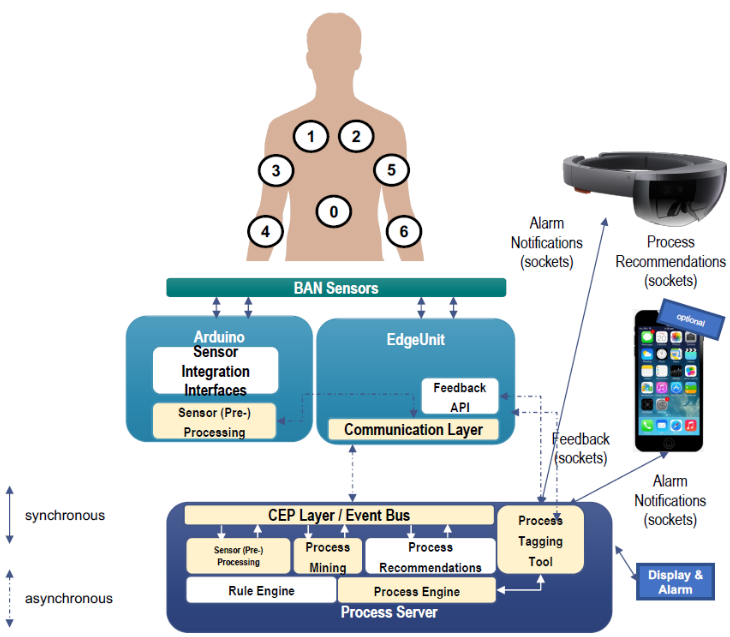

Figure 3: Hardware & Software Architecture (only yellow-highlighted components are relevant)

Source: Own

4.2 Software Architecture

As Figure 3 indicates, the overall system interacts with the following **components** in its infrastructure: the *Arduino* controller mainly has the functionality to multiplex and pre-process the sensor signals and collect them in common data frames and forwards them to the *Edge Unit*, which in our case is the Raspberry Pi. In this case, its main purpose is the data transport to a backend server via its communication layer, which is implemented with a socket connection.

The **Process Server** is the main backend server which is responsible for detecting, monitoring, analyzing and recommending the process and associated actions. In the presented task, the incoming data is received through the *Event Bus* via socket connections. The *Sensor Preprocessing* on this side prepares the data for the respective machine learning algorithms (cf. section 5), e.g. by aggregating individual data frames into windowed data frames, building stacker functions, etc. The *Process Mining component* performs the actual algorithms for determining the current and predicted process activities. This is the main focus of the evaluations in this work. The *Process Tagging Tool* is a Universal Web App that can be used both on PCs and tablets to tag the current activity. It is used to train the system in a supervised learning manner. Whenever a process model is chosen in this app, it shows the list of available activity types, and by selecting one the given activity type is started, while the previously running activity instance is stopped. The approach is sufficient for this purpose, as the use cases involve no parallel activities. Identifying these time segments allows filtering of relevant sensor data frames for training detection models. The *Process Engine* – an open-source jBPM implementation (cf. Cumberlidge, 2007) – is a standard business process management (BPM) engine used to enact and monitor instances. Triggered either by automated detection models or manual tagging via the Tagging App, it provides the ground truth for process data. The resulting process logs serve as output labels for detection models.

5 Activity Recognition and Model-based Post-filtering

The following describes the overall detection pipeline. The HAR component serves as an interchangeable baseline classifier; its specific architecture is not the focus of this work. The primary contribution is the rule-based filtering stage described in Section 5.4, which leverages process model knowledge to refine classifier output.

5.1 Input & Output Models

A DSP sensor captures 3-dimensional angles in space, with a set of sensors oriented in relation to a root sensor (sensor 0) located at the lower back. Consequently, a sensor data frame is defined as follows:

$$s_i = \langle x_i, y_i, z_i \rangle \; with \; x_i, y_i, z_i \in \mathbb{R}[-180, +180] \tag{1}$$

The input and output vectors for the learning models are defined as:

$$\langle s_1, \dots, s_i \rangle \rightarrow a \; with \; i \in \mathbb{N} = 3,5,7 \tag{2}$$

This setup utilizes either the full 7-sensor suit (as depicted in Figure 3) or specific subsets. The model predicts the label of the current activity.

5.2 Data Preparation & Stacking

To build a supervised predictive model, it is essential to build the data with proper labelling. The input vectors s_i and the output vectors a_i are obtained from two different components as described in Figure 3. The a_i vectors obtained from the process tagging tool are cleaned by identifying incomplete processes and wrong process tags due to human errors while tagging.

Once the data is cleaned, the input and output vectors are mapped to each other using the time stamp on which it was collected. Let a_j and a_{j+1} be two subsequent activities, any s_i collected between the corresponding timestamps of a_j and a_{j+1} will be labelled as a_j. Any s_i with no corresponding a_i is removed.

The cleaned and labelled data is then stacked with a windowing length of four, i.e. a 4x21 matrix is formed for every four sequential data points when we consider all the seven sensors with x_i, y_i and z_i for each. This enables the model to learn how an activity changes over a short period of time, making it more realistic and reliable.

The stacked data is then split into train, validation and test sets for model training. First, an 80:20 split is done on the stacked data to create a train set and a test set. A validation set is created by doing an 80:20 split on the train set created in the previous step. This results in an overall data split of 64 %|16 %|20 % for train, validation and test set respectively.

5.3 Network Architecture

A deep learning model is built to identify activities from sensor input vectors. To extract temporal features from sequential sensor data, a convolutional neural network (CNN) is integrated with a bidirectional long short-term memory (BiLSTM) layer. The CNN layers employ a 2x2 kernel and Rectified Linear Unit (ReLU) activation functions. For improved convergence speed, batch normalization is applied after each CNN layer, while dropout provides regularization to prevent overfitting. A BiLSTM layer with 64 units, followed by a linear layer and a softmax function, classifies the activities. Figure 4 illustrates this network architecture.

The Adam optimizer is employed with a learning rate of 10^{-4}. Its adaptive learning rate enables faster optimization compared to standard stochastic gradient descent. Cross-entropy serves as the loss function, as the softmax layer outputs a probability distribution across classes. Accuracy is chosen as model performance metric.

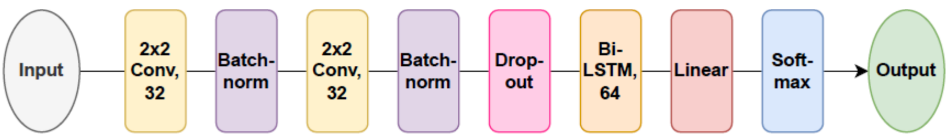

Figure 4: The network architecture used to do activity detection.
Source: Own

5.4 Rule-based Post-filtering of the Steps

Let M be the process model with activity types $\langle a_1, \dots, a_n \rangle$ with n as the number of activity types contained in M. Let furthermore a_i' be instantiations of the respective activity type a_i. A successor candidate $sc(a_i)$ contains the set of all successor candidates of activity type a_i. The objective is to ensure that the current running activity instance a_i' matches the type detected by the machine learning algorithm.

The algorithm outputs probabilities for all available activity types $a_1, \ldots, a_n$. Through continuous evaluation of the active instance, all predictions are filtered out that neither satisfy the successor function $sc(a_i)$ nor match a_i itself. Given the prediction probability p for a candidate activity a_k, the activity type with the highest probability score hs among these candidates is selected (cf. Figure 5).

```
Data:   n ∈ ℕ
Data:   a'i ← current activity instance
Result: a* ← activity type with highest score in C

C ← sc(a'i);
if C ≠ ∅ then
    k ← 0;
    m ← |C|;                    # elements in C
    hs ← 0;
    a* = NIL;
    while k < m do
        s ← p(ak);
        if s > hs then
            hs ← s;
            a* ← ak;
        end
    end
end
```

Figure 5: Algorithm for post-filtering

Source: Own

6 Evaluation

To provide a proof-of-concept implementation, the approach was tested in a laboratory environment covering the use cases from Section 3. The evaluation is designed to isolate the contribution of the filtering stage. Accordingly, the comparison between unfiltered and filtered predictions – across varying sensor configurations – constitutes the primary experiment. The absolute performance of the underlying classifier is deliberately secondary; what matters is the delta introduced by filtering. The two use cases were tested with 27 participants. However, experiments were conducted under explicit instructions and supervised to ensure correct process execution. Models were trained during the test runs using the Tagging App described in Subsection 4.2. As stated in Section 3, users were free to choose between left/right variants, or in which order to perform certain tasks (UC2).

6.1 Experimental Setup

Three distinct sensor setups are evaluated for each use case: (i) root and shoulder sensors (0, 1, 2); (ii) root, shoulders, and upper arm sensors (0, 1, 2, 3, 5); and (iii) the full 7-sensor setup. To analyze the impact of post-filtering on predicted process activities, unfiltered and filtered approaches are compared across all sensor setups.

6.2 Results

Table 1 allows the comparison of different sensor setups and pre- and post-filtering of predicted process activities. The sensor setup with all seven sensors consistently performs best for both use cases and for the non-filtered and filtered approaches. The difference in test accuracies for the different sensor setups without any filtering is as high as approximately 10 %, however, when the filtering approach is used, it reduces significantly to 2.6 % – 5.36 %. It is evident that post-filtering predictions consistently and significantly improves test accuracy.

Table 1: Accuracy comparison of Filtering approaches

Case	Sensor Setup	Model Filtering	
		No	Yes
UC1	3	47.00 %	82.09 %
	5	50.40 %	83.35 %
	7	56.94 %	84.69 %
UC2	3	38.70 %	81.72 %
	5	42.04 %	85.12 %
	7	46.42 %	87.08 %

Source: Own

Figure 6 helps to understand class-wise performance. When no filtering is applied, the activities “check rail”, “close openings” and “take rail” are often confused with “mover rail to the holder”. Similarly, “bend down to box” is sometimes confused with “deposit rail in box”. Applying the filtering approach, Figure 6 shows that almost all classes perform very well.

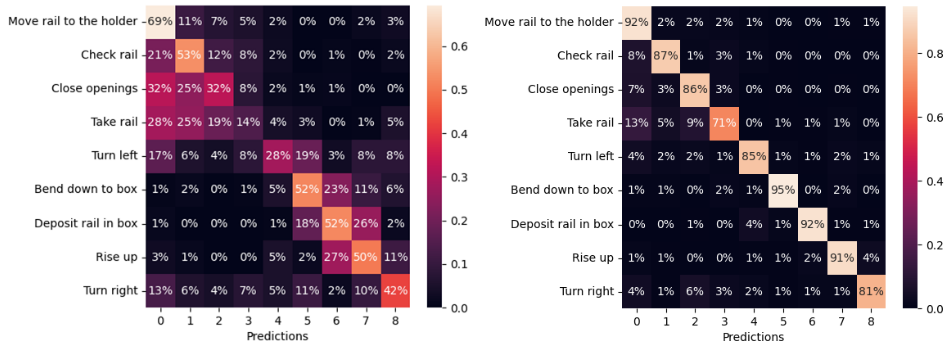

Figure 6: Confusion matrices for use case 2 without (left) and with (right) post-filtering of predicted activities. Rows represent actuals, columns represent predictions.
(Source: own)

6.3 Discussion

Regarding the **sensor setups**, Table 1 demonstrates that increasing the number of sensors improves detection accuracy. However, overall accuracy remains relatively low, with the 7-sensor configuration providing the greatest advantage. An analysis of the confusion matrices for UC2 reveals that the 3-sensor setup leads to high misclassification rates for manual activities such as “close openings” and “take rail”. While these rates remain elevated across all configurations, they consistently decrease as sensors are added, indicating that the chosen sensor design does not fully capture hand movements. This is acceptable, however, as the primary goal described in Section 3 is to analyze upper-back ergonomics.

Model filtering consistently outperforms non-filtered alternatives. Notably, filtering considerably reduces the accuracy gap between the different sensor setups in both use cases. While results reach a “good” range, the peak accuracy (87.08 % for the UC2 7-sensor setup) remains modest. For both use cases, the frequency of confused activities (discussed in Section 6.2 and seen in Figure 6) improves significantly after filtering. **Limitations** that negatively impact performance are:

1. *Activity transitions:* Despite the largely linear process model which might suggest near 100 % accuracy the system must recognize activity boundaries. These transitions are not fully captured and should be addressed in future work.

2. *Inexperienced test subjects:* The use of laboratory participants rather than trained workers occasionally caused delays and execution errors, impacting the consistency of the sensor data.
3. *Simultaneous non-determinism of sensor setup and process model:* In cases with process decision splits, the process model alone cannot determine the next activity. If the sensor setup also fails to distinguish between these candidates (e.g., "check rail" and "close openings" in UC2), filtering approach lacks the necessary data to resolve the ambiguity reliably.
4. *Low raw accuracy of the baseline method:* Due to the very restricted sensor setup that intentionally prioritizes upper-body ergonomics over fine-grained finger or hand tracking. Consequently, these baseline results do not represent a failure of the machine learning model, but rather highlight the 'information gap' created by data minimization – a gap that our rule-based post-filtering approach is specifically designed to bridge.

7 Conclusion

This paper presents a novel approach for predictive monitoring of human work processes, leveraging model data to minimize sensor setups and machine learning requirements for activity recognition. Results from configurations with 3, 5, and 7 sensors across two laboratory-modeled industrial workplaces demonstrate that explicit process models enable a rules-based post-processing approach to recognize activities more efficiently and with fewer sensors than purely sensor-based methods. This proof-of-concept suggests the approach should be considered when benchmarking new capturing environments for human work processes.

Limitations: The models follow a sequential pattern with few splits or forks. While more advanced sensor setups or baseline algorithms reduce the accuracy gap between unfiltered and filtered approaches, filtered accuracy is still expected to remain significantly higher.

Outlook: Furthermore, the demonstrated ability to achieve over 80 % accuracy with only a 3-sensor setup has significant practical and economic implications. By reducing hardware requirements, the system lowers initial procurement costs and minimizes the time and training effort required for workers. These factors are critical for the adoption of ergonomic monitoring in real-world settings, particularly within

Small and Medium Enterprises (SMEs) where resource constraints often hinder the implementation of high-fidelity capturing environments.

Acknowledgements

This Paper has been written in the context of the project "ErgoBest" funded under the ID "16SV8672" by the German Federal Ministry of Education and Research (BMBF). We want to thank our project partners and especially our student assistants Jonybul Islam, Mejbah Uddin Shameem and Anushka Kothari for their assistance in the laboratory experiments.

References

Assenza, G., Fioravanti, C., Guarino, S., & Petrassi, V. (2020). New Perspectives on Wearable Devices and Electronic Health Record Systems. *2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT*, 740–745. 2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT). https://doi.org/10.1109/MetroInd4.0IoT48571.2020.9138170

Chen, K., Zhang, D., Yao, L., Guo, B., Yu, Z., & Liu, Y. (2021). Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges, and Opportunities. *ACM Computing Surveys*, *54*, 1–40. https://doi.org/10.1145/3447744

Corona, E., Pumarola, A., Alenya, G., & Moreno-Noguer, F. (2020). *Context-Aware Human Motion Prediction*. 6992–7001. https://openaccess.thecvf.com/content_CVPR_2020/html/Corona_Context-Aware_Human_Motion_Prediction_CVPR_2020_paper.html

Cumberlidge, M. (2007). *Business Process Management with JBoss jBPM*. Packt Publishing. https://univ-senegal.scholarvox.com/catalog/book/88852238?_locale=en

De Leoni, M., & Pellattiero, L. (2022). The Benefits of Sensor-Measurement Aggregation in Discovering IoT Process Models: A Smart-House Case Study. In A. Marrella & B. Weber (Hrsg.), *Business Process Management Workshops* (Bd. 436, S. 403–415). Springer International Publishing. https://doi.org/10.1007/978-3-030-94343-1_31

Demrozi, F., Turetta, C., Kindt, P., Chiarani, F., Bacchin, R., Valè, N., Pascucci, F., Cesari, P., Smania, N., Tamburin, S., & Pravadelli, G. (2023). A Low-Cost Wireless Body Area Network for Human Activity Recognition in Healthy Life and Medical Applications. *IEEE Transactions on Emerging Topics in Computing*, *PP*, 1–12. https://doi.org/10.1109/TETC.2023.3274189

Faramondi, L., Bragatto, P., Fioravanti, C., Gnoni, M., Guarino, S., & Setola, R. (2019). *A Wearable Platform to Identify Workers Unsafety Situations* (S. 343). https://doi.org/10.1109/METROI4.2019.8792857

Francescomarino, C. D., Ghidini, C., Maggi, F. M., & Milani, F. (2018). *Predictive Process Monitoring Methods: Which One Suits Me Best?* (Bd. 11080, S. 462–479). https://doi.org/10.1007/978-3-319-98648-7_27

Hevner, A., R, A., March, S., T, S., Park, Park, J., Ram, & Sudha. (2004). Design Science in Information Systems Research. *Management Information Systems Quarterly*, *28*, 75.

Knoch, S., Ponpathirkoottam, S., Fettke, P., & Loos, P. (2018). Technology-Enhanced Process Elicitation of Worker Activities in Manufacturing. In E. Teniente & M. Weidlich (Hrsg.), *Business Process Management Workshops* (S. 273–284). Springer International Publishing. https://doi.org/10.1007/978-3-319-74030-0_20

Lara, O., & Labrador, M. (2013). A Survey on Human Activity Recognition Using Wearable Sensors. *Communications Surveys & Tutorials, IEEE*, *15*, 1192–1209. https://doi.org/10.1109/SURV.2012.110112.00192

Munoz-Gama, J., Martin, N., Fernandez-Llatas, C., Johnson, O. A., Sepúlveda, M., Helm, E., Galvez-Yanjari, V., Rojas, E., Martinez-Millana, A., Aloini, D., Amantea, I. A., Andrews, R., Arias, M., Beerepoot, I., Benevento, E., Burattin, A., Capurro, D., Carmona, J., Comuzzi, M., … Zerbato, F. (2022). Process mining for healthcare: Characteristics and challenges. *Journal of Biomedical Informatics*, *127*, 103994. https://doi.org/10.1016/j.jbi.2022.103994

Nunamaker, J., Briggs, R., Derrick, D., & Schwabe, G. (2015). The Last Research Mile: Achieving Both Rigor and Relevance in Information Systems Research. *Journal of Management Information Systems*, *32*, 10–47. https://doi.org/10.1080/07421222.2015.1094961

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Peng, W., Hong, X., Chen, H., & Zhao, G. (2019). *Learning Graph Convolutional Network for Skeleton-based Human Action Recognition by Neural Searching* (arXiv:1911.04131). arXiv. https://doi.org/10.48550/arXiv.1911.04131

Raso, R., Emrich, A., Burghardt, T., Sträter, O., Fettke, P., & Loos, P. (2019). *Einsatz eines textilbasierten Assistenzsystems zur Analyse von körperlich anstrengenden Arbeitsprozessen*. Springer. https://doi.org/10.22028/D291-33497

Rebmann, A., Emrich, A., & Fettke, P. (2019). Enabling the Discovery of Manual Processes Using a Multi-modal Activity Recognition Approach. In C. Di Francescomarino, R. Dijkman, & U. Zdun (Hrsg.), *Business Process Management Workshops* (S. 130–141). Springer International Publishing. https://doi.org/10.1007/978-3-030-37453-2_12

Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA Relevance), 119 OJ L (2016). http://data.europa.eu/eli/reg/2016/679/oj

Schaefer, P., Boocock, M., Rosenberg, S., Jäger, M., & Schaub, Kh. (2007). A target-based population approach for determining the risk of injury associated with manual pushing and pulling. *International Journal of Industrial Ergonomics, Musculoskeletal Load of Push–Pull Tasks*, *37*(11), 893–904. https://doi.org/10.1016/j.ergon.2007.07.008

Schaub, K., Mühlstedt, J., Illmann, B., Bauer, S., Fritzsche, L., Wagner, T., Bullinger-Hoffmann, A., & Bruder, R. (2012). Ergonomic assessment of automotive assembly tasks with digital human modelling and the ‘ Ergonomics Assessment Worksheet ’ (EAWS). *International Journal of Human Factors Modelling and Simulation*, *3*, 398. https://doi.org/10.1504/IJHFMS.2012.051581

Sundstrup, E., Seeberg, K., Bengtsen, E., & Andersen, L. (2020). A Systematic Review of Workplace Interventions to Rehabilitate Musculoskeletal Disorders Among Employees with Physical Demanding Work. *Journal of Occupational Rehabilitation*, *30*. https://doi.org/10.1007/s10926-020-09879-x

Wan, S., Qi, L., Xu, X., Tong, C., & Gu, Z. (2020). Deep Learning Models for Real-time Human Activity Recognition with Smartphones. *Mobile Networks and Applications*, *25*. https://doi.org/10.1007/s11036-019-01445-x