Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 106 (2022) 233–238

www.elsevier.com/locate/procedia

9th CIRP Conference on Assembly Technology and Systems

A Methods-Time-Measurement based Approach to enable Action Recognition for Multi-Variant Assembly in Human-Robot Collaboration

Julian Koch[a,*], Lukas Büsch[a], Martin Gomse[a], Thorsten Schüppstuhl[a]

[a]*Hamburg University of Technology, Institute of Aircraft Production Technology, Denickestraße 17, 21073 Hamburg, Germany*

* Corresponding author. Tel.: +49 40 428 78 - 4324; fax: +49 40 427 3 - 14551. *E-mail address:* julian.koch@tuhh.de

Abstract

Action Recognition (AR) has become a popular approach to ensure efficient and safe Human-Robot Collaboration. Current research approaches are mostly optimized for specific assembly processes and settings. This paper introduces a novel approach to extend the field of AR to multi-variant assembly processes. The approach is based on generalized action primitives derived from Methods-Time-Measurement (MTM) analysis that are detected by an AR system using skeletal data. Subsequently a search algorithm combines the information from AR and MTM to provide an estimate of the assembly progress. One possible implementation is shown in a proof of concept and results as well as future work are discussed.

Keywords: Assembly; Human Action Recognition; Assembly Step Recognition; Human-Robot Collaboration; Industry 4.0; Methods-Time-Measurement; Azure Kinect; Skeleton Based Action Recognition; Particle Swarm Optimization; Artificial Neural Network

1. Introduction

With the continuing trend of mass customization the need for flexible production systems with high output rates is increasing. One approach to achieve this is Human-Robot Collaboration (HRC), combining capabilities of humans and robots in a shared workspace. Based on their individual strengths tasks are allocated either to the human or the robot maximizing efficiency and shortening cycle times by parallel process execution [11]. When performing tasks together in a shared workspace, new challenges arise to ensure efficient, safe and natural collaboration. The implementation of appropriate human-machine interfaces to achieve a bidirectional exchange of information during process execution is a key factor for this. Often the information channel from the robot to the human is used to display the current status and intention of the robot increasing the acceptance and safety of the worker [9]. To react and adapt to actions of the human, the robot must be context aware. In HRC manual input devices like buttons are used to confirm completed processes, tracking the progress - interrupting the process flow and representing a non-value-adding action. In this context, fully automated Assembly Step Estimation (ASE) based on human AR is a promising approach for efficient HRC. To preserve the flexibility of HRC, the ASE must be small stepped and applicable to multi-variant processes and cross-process usage.

The paper first deals with the results of the current research. Then, a novel concept for an ASE is developed that is applicable to multi-variant processes and cross-process usage. Subsequently, the concept is implemented for an assembly process of a product from the aircraft production as an example. The results of the experiments are discussed and finally further improvement potentials for future work are identified. The entire code, as well as further information, is made available to the research community in the Git repository: https://github.com/LukasBuesch/MTMAssemblyStepEstimation.

2. Related Work

In this section the state of the art of ASE and AR relevant for this work is explained and discussed. ASE refers to the estimation of progress in an assembly plan. In this context, AR defines methods for recognizing human actions based on captured motion data.

10.1016/j.procir.2022.02.184

2.1. Assembly Step Estimation

Several publications have already dealt with determining the progress of an assembly process without user input. An assembly step recognition based on basic motions is proposed in [1]. A Hidden-Markov model (HMM) is trained for detecting an assembly task from position information of both hands of the worker. However, the experiment is limited to the detection of a certain trajectory in the assembly process. Consequently, the HMM needs to be trained on a specific process and cannot be reused for cross-process usage or multi-variant assembly without additional training of the HMM. Although the overall concept for the HMM is difficult to apply to varying processes, using the first layer of the HMM to classify and detect basic movements like ”move” and ”bring” is of interest for this work.

In [2] an object detection system is enabled to recognize the tools which are in use to estimate the current assembly progress. In addition to the object detection, a pose estimation of the worker based on a RGB video feed is used to determine operation times of the worker. This approach provides a generalized ASE for multi-variant processes and cross-process usage. However, an AR method or other techniques capable of determining assembly tasks without tool usage are not included. Information about the workers motion are captured and processed only for determining execution times after the worker grabs a tool. Hence, the approach is not capable of detecting manual processes without the usage of tools such as placing screws by hand, which limits it to processes with the use of many different tools, for a small stepped ASE.

An assembly task can also be modeled with finite states describing the task small stepped, which is done in [6] by adapting a Finite State Machine (FSM). Using video-based localization and object detection, the state transitions of the FSM are detected, which offers a well comprehensible method for determining the progress of the assembly process. Since the task models are handmade, the system is not applicable to varying processes without generating new task models. In addition, the complexity of the FSM scales linearly with the length of the assembly task, making the system incapable of handling long-lasting assembly processes.

Summarizing, the state of the art contains promising approaches for ASE but they are not small stepped applicable to multi-variant processes and cross-process usage.

2.2. Action Recognition

Methods for AR are subject of many current publications in the field of computer vision with a variety of different approaches available, almost all of them based on machine learning techniques. According to [10] two basic principles of AR are ascendant in current publications: AR methods based on depth features extracted from depth camera feeds and AR methods based on skeleton data obtained by motion sensors or camera systems. Since the concept is supposed to be applicable for varying processes, the AR method needs to be as independent of the environment as possible. Skeleton based AR methods are less dependent on the background [10] and therefore more robust against environmental changes than depth feature based AR methods. Among all compared AR Methods in [10], the Spatial Temporal Graph Convolutional Networks (ST GCNs) [12] currently outperforms all other methods.

The ST GCN is a graph convolutional neural network that takes spatial-temporal graphs as input. For AR applications these graphs are skeleton data where various points of the human body, hereafter referred to as body joints, are described by 6 coordinates (3 translational and 3 rotational degrees of freedom). The spatial aspect is the defined connection of certain body joints, the spatial edges, which represent the bones in the skeletal data. The temporal edges represent the connection of the single body joints with itself over the time.

The ST GCN was only applied to offline AR datasets in [12]. To enable the ST GCN for online AR, the use of a preprocessing Sliding Window (SW) was proposed in [4]. For this, a SW with a defined size was placed over a continuous skeleton data stream and was moved along with time. The skeleton data within the SW formed the spatial-temporal graph.

Recent publications on ASE focus on recognizing a specific process step in a fixed setting [1, 6]. However, those solutions are usually not transferable to other related or even novel processes. So far, there are only limited approaches to recognize the processes of different product variants [2] which contradicts the application of HRC with ASE as a flexible production system. Therefore, this paper presents a novel approach to enable ASE to be small stepped applicable to multi-variant products and their processes in HRC. For this purpose, a generalized description of processes using Methods-Time-Measurement (MTM) methods is proposed, which serve as action primitives for an AR and can be used independently of the specific process.

3. Concept for Assembly Step Estimation: A Methods-Time-Measurement based Approach

To enable the ASE method for multi-variant assembly processes or cross-process usage, it is not sufficient to train the AR system on a specific process. Those processes can be composed by assembly tasks of generalized descriptions, which again can be decomposed into very small stepped action primitives. Instead of defining new action primitives for the AR, the use of already available action primitives from a MTM analysis is proposed.

During the assembly the movements of the human worker are captured by a motion capturing system, preprocessed and serves as input for an AR system. The AR system is trained on determining the before mentioned MTM action primitives from human motions. To increase the systems accuracy, the usage of object detection and localization methods is proposed. This information combined with the time history of the detected action primitives and the existing assembly plan estimates the state of the assembly process. The concept is depicted in fig. 1 and will be explained in detail in the following subsections. First the inputs, then the processing of these via software modules and lastly the final output function are discussed.

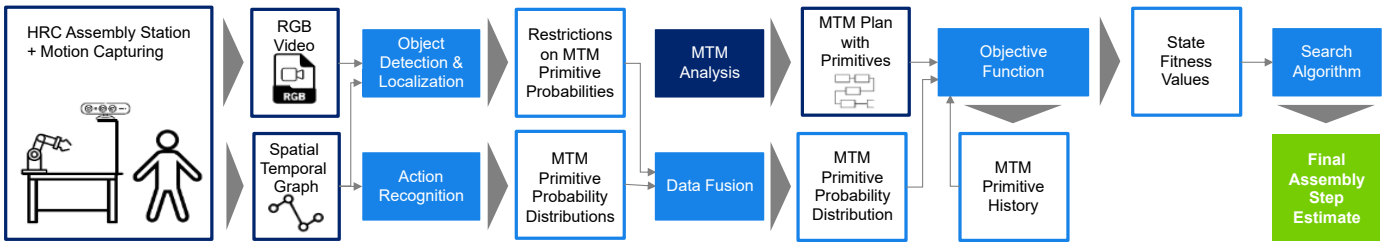

Fig. 1. Concept for Assembly Step Estimation. Legend: • Input, • Software Module, • Output

3.1. Methods-Time-Measurement

MTM analysis is an established tool for the decomposition of industrial processes [8]. MTM allows describing an assembly process in small steps, with the basic movements of a working person and their respective execution times, which are called MTM Action Primitives (MTM APs) in the following. Since MTM analysis has remained a popular technique in industry for planning and analyzing processes for many years, MTM assembly plans are often available for industrial processes. Obtained by a preceding MTM analysis of the assembly process, the MTM assembly plan describes the underlying process in detail with sequential MTM APs. Hence, these MTM assembly plans can be used for ASE saving the effort of creating new process analyses for the respective assembly process. It follows that it is beneficial to use the MTM APs instead of primitives especially defined for the corresponding use-case.

3.2. Motion Capture

In order to avoid disruptions and to prevent inserting non-value adding activities in the workflow, the information about the assembly process is collected using a motion capturing system instead of manual user inputs. Furthermore, a camera-based motion capturing system offers a simpler system integration and a greater flexibility compared to attaching sensors to the worker since workers can alternate during the process. As already mentioned in sec. 2, it is beneficial to convert the data from the motion capturing system into a skeleton data format. Since the skeletal data depicts the position relative to the respective camera, the use of multiple cameras is also possible, whereby the bodies detected in each unit can be joined by their absolute position. Thus, the observable process environment can be extended to, for example, the whole assembly line.

The stream of skeletal data must be preprocessed to serve as input for a subsequent AR system. For this purpose, a SW with a defined size, referring to [4], is used. The SW is placed over the data stream and moved along with the newly captured data according to the principle of a first-in-first-out queue. As input for the AR system, a spatial-temporal graph is created out of the captured data within the SW.

The use of the SW enables online AR but defines the observation period of the AR system. Thus, not all MTM APs can be detected equally well since they are of different lengths. This motivates the simultaneous use of multiple SWs with different sizes where each SW size is beneficial for detecting MTM APs of certain execution times.

3.3. Action Recognition

As discussed in sec. 2, state of the art AR systems are mostly based on neural networks. To decouple the AR system from the actual process it is designed to recognize MTM APs out of the captured motion data of the worker. Thus, training data is required that enable the training of such a network.

Processing the spatial-temporal graphs obtained by the motion capturing system, the output of the AR system are probabilities for each MTM AP the system is trained on. As mentioned before, it is beneficial to run the AR algorithm on different SW sizes, which, however, specifies the temporal dimension of the input graph. Since a neural network needs to be trained on a certain input dimension running as many neural networks as SWs in parallel is proposed. For this purpose, network structures are needed that can be executed as efficiently as possible, so that several neural networks can be computed simultaneously with appropriate hardware. Finally, a data fusion is performed to merge the several MTM action primitive probability distributions from the respective neural networks to form one probability distribution on the MTM APs.

3.4. Object Detection and Localization

When considering assembly processes solely from the perspective of human movements, there are sets of processes that are difficult to distinguish from one another. For example, screwing in a bolt is difficult to distinguish from tightening a nut based on observations of human movements. Therefore, it is reasonable to also collect information about the tools used to compensate for the previously described deficiency.

To detect a tool usage in the process, the integration of object detection methods is proposed, inspired by [2]. An object detection capable of recognizing tools used in the assembly process can be based on a RGB video feed, which is usually already available in optical motion detection systems. The integration of tool detection methods restricts the possible assembly steps to those with certain tool use. Hence, the obtained information has the ability to improve the accuracy of the recognized MTM APs. The position data of the human body parts, already available from the motion capturing, and the positions of any tools can be considered for recognizing whether a tool is being used by the worker.

3.5. Assembly Step Estimation

To finally estimate the progress in the assembly process, referred to as state in the following, all information needs to be taken into consideration. Therefore, an objective function is proposed which estimates the value of each state considering and weighting all available information. The value of a state corresponds to the probability that this state applies. An example of such an objective function can be found in sec. 4.2. Since the MTM analysis is small stepped and the AR should also work for long assembly processes, the solution space for the final step estimation must be assumed to be large. Therefore, a search algorithm is proposed to find the minimum of the objective function for the final state estimation.

4. Proof of Concept

To facilitate future applications of the developed concept, a proof of concept is conducted in the following. Particular attention is paid to the limitations of currently available methods in implementing the concept. The implementation of the concept is described concisely in the following. Therefore, the entire code, as well as further information like confusion matrices, is made available to the research community in the Git repository to enable reproducibility.

4.1. Assembly Process for Investigation

The proof of concept deals with one example process from multi-variant assembly. The multi-variant product selected originates from the aircraft interior production and is subject of current publications [7] in the research field of HRC. In the chosen sub-process, two angled aluminum sheets are mounted on a lightweight panel. This process is well suited for the proof of concept since all action primitives occur multiple times.

The workstation used for the proof of concept is depicted in fig. 2(a). Note that there is no demand for defined tool stock positions, allowing the worker to place unused tools anywhere in the workspace for fast reuse.

4.2. Implementation of the Concept

A first implementation of the developed concept is presented, which provides a basis for discussion on future work.

MTM analysis and assembly plan: Since the training data available for the AR system is limited, only a subset of the MTM APs can be used for process description. This subset will be elaborated on during the AR system implementation. The MTM assembly plan is based on a MTM analysis of the chosen assembly process. Therefore, the process and its tasks are decomposed using the subset of MTM APs which are defined as ”assemble”, ”take” and ”put down”.

To obtain values for the execution times for the respective MTM AP, the process was carried out by several subjects and the execution times of the decomposed assembly steps (MTM action primitives) were measured. Based on the measured values, an average execution time per MTM AP was determined.

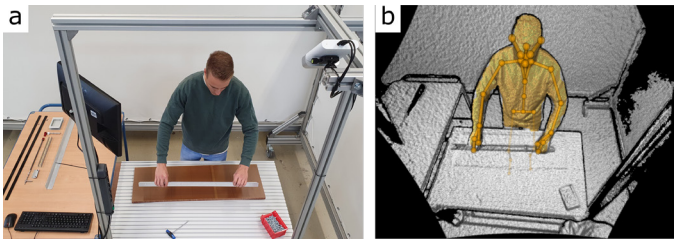

Fig. 2. (a) Workstation with Azure Kinect for multi-variant assembly process from aircraft interior production. (b) View of Azure Kinect.

Motion capture: For previously discussed reasons, a camera-based motion capturing system is used. Furthermore, the assembly process takes place at a well defined workstation where one camera unit is sufficient to cover the entire working area as it can be seen in fig. 2(b).

A camera system available on the market that can be used to develop applications for recognizing human actions using skeleton data is Microsoft’s Azure Kinect. The Azure Kinect is equipped with a RGB camera and a depth sensor and provides a Software Development Kit (SDK) for body tracking applications. The human body is modeled by the body tracking SDK as skeletal data using machine learning techniques as depicted in fig. 2(b). Thus, the workstation is equipped with an Azure Kinect device as a motion capturing unit. With the hardware used, the body tracking application runs on 24 Hertz which is mainly limited by the used GPU device (NVIDIA GeForce GTX 1650).

The captured data stream of skeletal data is preprocessed by three SWs in parallel to obtain the spatial temporal graphs serving as inputs for the AR system. The three SWs enable detection of the MTM APs used.

Action Recognition: With regard to the performance requirements, the ST GCN [12] was chosen and implemented as neural network for the AR system. The network implementation is based on [4], however the individual layers and dimensions were modified (see Git).

For teaching the ST GCN to detect different MTM APs, a training dataset representing those primitives is needed. Therefore, the InHARD dataset [3] is chosen since it provides motion data of humans in an industrial assembly context. However, the InHARD dataset does not provide pure MTM APs as action classes. Instead, fourteen basic assembly actions are provided where some are also dependent on gripping positions. Since the AR system needs to operate on multi-variant processes, features like stock positions of tools will vary and cannot be used for the AR implementation of the concept. After inspecting the action classes and the corresponding training data, three action classes are formed: ”assemble system”(0), ”put down”(1), ”take”(2). In the MTM context, these action classes exit the realm of basic movements and are to be classified as basic operations. Even though the concept suggests MTM basic movements as MTM APs, the availability of training data does not allow a smaller-

stepped analysis here. These three action classes are a subset of merged action classes from the InHARD and best depict a subset of MTM APs over all available skeleton-based AR datasets.

In the InHARD dataset all motions are captured by attaching motion sensors to the worker which collects data with an Inertial Measurement Unit (IMU) at a rate of 120 Hertz. Since the motion capturing method used runs on a lower frame rate, different approaches were tested to preprocess the training data by squeezing, thinning out and interpolating. The dataset was split for training and validation as proposed in [3]. Finally, the ST GCN was trained on the modified dataset with different SW sizes and could reach overall accuracies up to 85 percent on the validation data set. In general, an accuracy of 85 percent is considered as a valuable result in the AR domain. However, since the dataset has only recently become publicly available, there are currently no implementations with which the results can be compared. In addition, the number of action labels in the proof of concept were significantly reduced, making it difficult to compare the obtained result with the accuracies of future implementations using the dataset.

The, in the concept, proposed extension of the AR with object detection methods for the recognition of used tools is not implemented in the proof of concept. Lastly, the several MTM AP probability distributions of the neural networks are merged to one probability distribution by building the mean over each MTM AP probability.

Assembly Step Estimation: Receiving the MTM AP probability distribution from the AR in each step, the objective function value ($f(s)$) for each state (s) is antiproportional to the probability of its corresponding MTM action primitive (AP_{prob}). The objective function also includes the history of the MTM APs (h) received from the AR system. Assembly steps farther away from the most promising solution according to the MTM AP history also receive a higher objective function value. In a similar way, the execution times according to the MTM assembly plan are accumulated to an estimated assembly time ($t_{est}(s)$) and compared with the current assembly time (t_{ass}). States with an estimated assembly time near the current assembly time receive a lower objective function value. The weight (w_i) of each influence (i) can be controlled separately:

$$f(s) = w_{hist}||len(h) - s|| + w_{time}||len(t_{est}(s)) - t_{ass}|| + w_{AP}/AP_{prob}.$$

To obtain fast and accurate results, a Particle Swarm Optimization (PSO) is proposed as search algorithm. A PSO algorithm is a biological inspired search algorithm to find the minimum of an objective function. Moreover, the PSO is capable of following an in time moving optima of the objective function which is necessary for an online ASE. Due to its fast convergence properties a constriction factor PSO [5] is used.

5. Results

In the following, the results of the proof of concept are presented and a detailed discussion of the developed concept and its proof of concept is conducted.

The ASE was performed on the test process using different weights for the objective function. Setting a high weight on the accumulated MTM execution times (w_{time}), the ASE was able to predict the assembly progress with a mean of -2.9 seconds and a standard deviation of 5.6 seconds. This good performance was expected, since the execution times of the MTM APs were generated from the process itself and the process had a predefined assembly line, which was not deviated from. This variant is particularly suitable for linear processes in which a fixed sequence of tasks must be adhered to. However, the probability distributions of the MTM APs from the AR system did not achieve valuable accuracies. Therefore, experiments with a high weight on the MTM APs (w_{AP}) did not yield usable results. Hence, the proof of concept cannot yet meet the requirements set for the concept but serves as a basis for discussion on future work. The results of repetitions are similar. Thus, one result of each of the before mentioned weights of the objective function is depicted in fig. 3. More detailed results can be found in the Git.

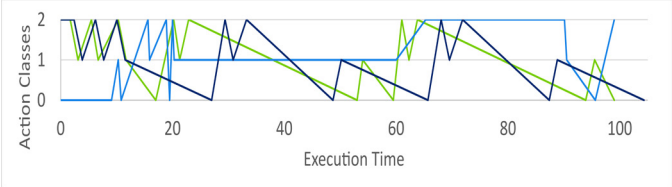

Fig. 3. ASE performance. Legend: • Ground Truth, • high weight on MTM action primitives (MTM APs) (w_{AP}), • high weight on execution times (w_{time}).

MTM methods can describe a wide variety of processes. The ASE concept trained on MTM APs has the potential to be applied to a wide variety of processes without additional training effort. This concept covers processes in which the sequence of assembly steps is not fixed over the entire process, since an action deviating from the assembly plan can be detected by the AR system. The objective function allows direct control on the various influences that finally lead to the ASE, so that the developed concept can be easily optimized for different process types.

The motion capturing technique of the training data and the technique used in the proof of concept exhibit different characteristics. The training data, captured by an IMU, has high sampling rates and less measurement noise, however, it is accompanied by a measurement drift over time. Whereas the captured skeletal data in the proof of concept, performed by an Azure Kinect device, is not subject to a measurement drift, but contains more noise and has a significantly lower sampling rate. Since deep learning features are extracted by the neural network during the training phase, the higher noise on the data in the proof of concept could attenuate the effect of the learned features. A considerably larger influence is certainly the lower sampling rate in the proof of concept. Several ways were attempted to compensate for this difference in preprocessing the training data as mentioned in sec. 4.

However, when implementing the whole ASE system, the used hardware was not sufficient to reach a sampling rate higher than 10 Hertz. The low sampling rate can be increased by more powerful hardware. For remedy, the use of a second GPU de-

vice, which can be used exclusively for motion tracking, is proposed. Actually, it is not crucial for the sampling rate whether only one neural network or three are executed parallel to the motion tracking on the GPU. This indicates that the GPU device encounters performance problems when it has to run other processes next to the bodytracking SDK of the Azure Kinect.

Furthermore, the training dataset used does not contain enough data to train a network for a generalized AR on all action classes available [4]. Since the number of action classes was decreased for the MTM approach the problem could be alleviated but not eliminated.

In addition, the ST GCN can still be improved. The original ST GCN [12] and its SW version [4] were developed and optimized for AR applications, but these networks were not used for AR in an industrial context. For the proof of concept, the AR network was optimized for the industrial training dataset, however, the development of a new neural network specifically for the industrial context was not in the focus of this work.

6. Conclusion and Outlook

This paper describes a novel concept for an ASE and makes an important contribution to overcome the lack of small stepped ASE methods for multi-variant processes and cross-process usage. This is achieved by extending AR methods to multi-variant processes. In a first proof of concept, the developed concept could be implemented with existing technologies and methods. Although the first proof of concept has not yet achieved a useful performance, the concept is still promising and this work provides guidance for future work. The goal of further work should be to accomplish a performant implementation of the concept. Based on the previously conducted discussion, suggestions are made hereafter concerning how this goal could be achieved.

The optimization of the neural network for the industrial context can be a starting point to reach faster learning with less training data. In addition the original ST GCN [12] features a Git repository which is continuously developed further for AR applications. Hence, this should be considered for future network optimizations.

Another approach for further work might be the implementation of the proposed tool detection. Recognition of tool usage can provide insightful information about the assembly progress and thus has the potential to highly increase the systems accuracy.

The lack of suitable training data for MTM based AR motivates the development of a new training dataset for human AR in industrial assembly processes. This dataset should contain several MTM APs as action classes. To teach a neural network the subtle differences of MTM APs, a large amount of training data is necessary. Therefore, the development of a collaborative dataset is proposed in which a certain standard for contribution is provided. A dataset allowing a supervised contribution from the research community could overcome the lack of training data. In addition, community contribution would diversify the data since a wide range of processes could be captured at various workstations. A subsequent work will investigate the requirements for such a collaborative dataset and provide a framework for implementation.

CRediT author statement

Büsch, Koch: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Data Curation, Visualization, Writing – Original & Draft & Review & Editing. **Koch:** Supervision, Project Administration. **Gomse, Schüppstuhl:** Resources, Funding Acquisition, Writing - Review & Editing.

References

[1] Berg, J., Reckordt, T., Richter, C., Reinhart, G., 2018. Action recognition in assembly for human-robot-cooperation using hidden markov models. Procedia CIRP 76, 205–210. doi:10.1016/j.procir.2018.02.029.

[2] Chen, C., Wang, T., Li, D., Hong, J., 2020. Repetitive assembly action recognition based on object detection and pose estimation. Journal of Manufacturing Systems 55, 325–333. doi:10.1016/j.jmsy.2020.04.018.

[3] Dallel, M., Havard, V., Baudry, D., Savatier, X., 2020. Inhard - industrial human action recognition dataset in the context of industrial collaborative robotics, in: Fortino, G. (Ed.), Proceedings of the 2020 IEEE International Conference on Human-Machine Systems (ICHMS), IEEE, [Piscataway, New Jersey]. pp. 1–6. doi:10.1109/ICHMS49158.2020.9209531.

[4] Delamare, M., Laville, C., Cabani, A., Chafouk, H., 2021. Graph convolutional networks skeleton-based action recognition for continuous data stream: A sliding window approach, in: Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, SCITEPRESS - Science and Technology Publications. doi:10.5220/0010234904270435.

[5] Eberhart, R., Shi, Y., 2000. Comparing inertia weights and constriction factors in particle swarm optimization, in: Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512), pp. 84–88 vol.1. doi:10.1109/CEC.2000.870279.

[6] Goto, H., Miura, J., Sugiyama, J., 2013. Human-robot collaborative assembly by on-line human action recognition based on an fsm task model. Human-Robot Interaction 2013 Workshop on Collaborative Manipulation URL: http://aisl.cs.tut.ac.jp/~jun/pdffiles/goto-hri13ws.pdf.

[7] Kalscheuer, F., Eschen, H., Schüppstuhl, T., 2021. Towards semi automated pre-assembly for aircraft interior production, in: Schüppstuhl, T., Tracht, K., Raatz, A. (Eds.), Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021, Springer International Publishing, Basel, Swiss. doi:10.1007/978-3-030-74032-0.

[8] Maynard, H.B., Stegemerten, G.J., Schwab, J.L., 1948. Methods-time measurement. McGraw-Hill.

[9] Müller, R., Hörauf, L., Vette-Steinkamp, M., Kanso, A., Koch, J., 2019. The assist-by-x system: Calibration and application of a modular production equipment for visual assistance. Procedia CIRP 86, 179–184. doi:10.1016/j.procir.2020.01.021.

[10] Wang, L., Du Huynh, Q., Koniusz, P., 2020. A comparative review of recent kinect-based action recognition algorithms. IEEE transactions on image processing : a publication of the IEEE Signal Processing Society 29, 15–28. doi:10.1109/TIP.2019.2925285.

[11] Weidner, R., Rodeck, R., Wulfsberg, J.P., Schüppstuhl, T., 2016. Supporting manual tasks - using the example of the quality-critical process of scarfing of cfrp structures. wt Werkstattstechnik online 106, 624–630. doi:10.37544/1436-4980-2016-09-50.

[12] Yan, S., Xiong, Y., Lin, D., 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. Proceedings of the AAAI Conference on Artificial Intelligence 32. URL: https://ojs.aaai.org/index.php/aaai/article/view/12328.