2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)

A Sense of Quality for Augmented Reality Assisted Process Guidance

Anes Redžepagić[1,2*†] Christoffer Löffler[1,2‡] Tobias Feigl[1,2§] Christopher Mutschler[2,3¶]

[1] Friedrich-Alexander University Erlangen-Nuremberg (FAU), Erlangen, Germany
[2] Fraunhofer Institute for Integrated Cirtuits IIS, Nuremberg, Germany
[3] Ludwig-Maximilians-University (LMU), Munich, Germany

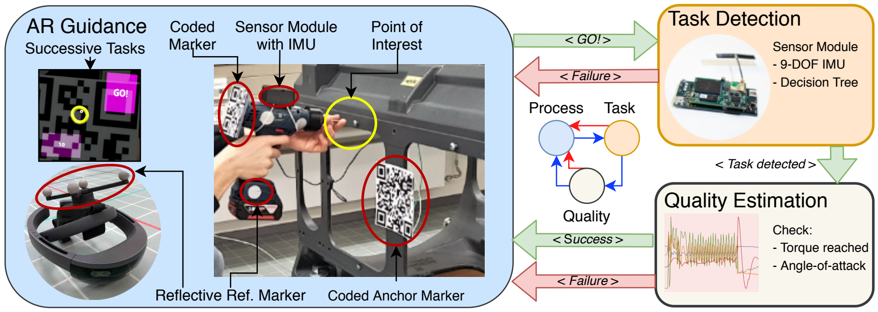

Figure 1: Users wear HMDs and use a sensor-augmented hand-held tool. They follow the task-by-task guidance to complete manufacturing processes. We continuously monitor the process, detect and classify actions, and extract quality metrics via ML.

Abstract

The ongoing automation of modern production processes requires novel human-computer interaction concepts that support employees in dealing with the unstoppable increase in time pressure, cognitive load, and the required fine-grained and process-specific knowledge. Augmented Reality (AR) systems support employees by guiding and teaching work processes. Such systems still lack a precise process quality analysis (monitoring), which is, however, crucial to close gaps in the quality assurance of industrial processes.

We combine inertial sensors, mounted on work tools, with AR headsets to enrich modern assistance systems with a sense of process quality. For this purpose, we develop a Machine Learning (ML) classifier that predicts quality metrics from a 9-degrees of freedom inertial measurement unit, while we simultaneously guide and track the work processes with a HoloLens AR system. In our user study, 6 test subjects perform typical assembly tasks with our system. We evaluate the tracking accuracy of the system based on a precise optical reference system and evaluate the classification of each work step quality based on the collected ground truth data. Our evaluation shows a tracking accuracy of fast dynamic movements of 4.92 mm and our classifier predicts the actions carried out with mean F1 value of 93.8% on average.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Mixed / augmented reality;

1 Introduction

In recent years we saw a shift towards automation of modern production towards Industry 4.0. While automation dominates many processes, still, manual labor remains important due to its flexibility and ease of deployment [6]. In this context, Augmented Reality (AR) assistance already adds many benefits for workers, such as better worker training and guidance [19, 20], monitoring [9, 16, 17], or process optimization [13]. However, these widespread AR approaches are unable to accurately measure the quality of the tasks performed and, unlike automated processes, lead to gaps in the quality assurance process. However, to identify malfunctions and defects in the products, they must be checked by a final inspection or a camera system upon their completion. There are some intelligent tools with built-in control units, but they are expensive and limited in their variety, e.g., [1] [2].

Besides these common usages, AR headsets have previously been deployed to monitor the environment for other means than visualizations. On one hand, the built-in cameras were used to automatically track and label real-world objects [5, 18], and depth sensors were used to compare the current work progress with a target/final 3D model [17]. These methods were also used to capture work processes, such as detecting the current stage in an assembly process. However, they are as limited as other optical quality assurance systems as they cannot extract important details of a work process, e.g., a screw's torque, from images. On the other hand, the positional tracking of AR systems is limited [7, 10] and impedes reliable tracking of ego-motion and arbitrary objects. While both external [15, 21] and internal systems such as simultaneous localization and mapping (SLAM) can assist and enhance the positioning performance [11], they have a high cost associated with them, reduce mobility and versatility, and suffer from computational complexity.

In contrast, sensor modules that function as add-ons for conventional work tools capture sensor characteristics automatically [14]. Such low-cost modules can be adapted to a large variety of tool types and seem to be natural partners for AR-assistance systems. In this paper we propose to co-opt existing AR-assisted processes and add quality predictions, that are orthogonal to what optical sensors can provide. To add this sense of quality we attach sensor modules to conventional work tools, and use a Machine Learning (ML) classifier to predict quality metrics for each work step of a process.

We use a modern AR head-mounted display (HMD) and a sensor module (for details see [14]), to simultaneously guide workers and to provide them with immediate feedback on the quality of their

*anes.redzepagic@fau.de
†A. Redžepagić and C. Löffler contributed equally to this work.
‡christoffer.loeffler@fau.de
§tobias.feigl@fau.de
¶christopher.mutschler@iis.fraunhofer.de

978-1-7281-7675-8/20/$31.00 ©2020 IEEE
DOI 10.1109/ISMAR-Adjunct51615.2020.00046

work. An overview of the components and pipeline is depicted in Fig. 1. First, we develop an exemplary but modular and expandable guidance in AR, rendered as 3D overlays to workpieces. Next, we extend the 3D tracking capabilities of the HoloLens to track the position and orientation of the HMD, tool, and workpiece with encoded markers via the Vuforia [12] SDK. Using this tracking, we monitor the work process in a finite state machine. Finally, we attach a sensor module to conventional tools, such as the electric screwdriver from Fig. 2 (left), to capture details of the work process. The key algorithm then uses ML to process the sensor data directly when the actions are executed. Our Decision Tree classifier (DTC) uses a single inertial measurement unit (IMU) (with an accelerometer, a gyroscope, and a magnetometer). We predict quality metrics on the broader process-level using the HMD tracking, e.g., to detect the correct order of tasks, and the deeper task-level using ML, e.g., by detecting actions and their duration, from that we can infer whether a predetermined torque was reached. The system detects errors early, and provides direct feedback to workers. Our add-on approach *learns on the job*, and aims at being as flexible and adaptive as the workers themselves.

Our approach compensates the optical tracking limitations of modern AR systems as it exploits external sensor information and ML to enhance AR-assisted worker guidance systems, that we validate in a case-study for a typical assembly process. We monitor the progress and predict quality metrics. The evaluation of the HMD's tracking accuracy and the sensor module's classification accuracy in a prototypical environment shows the system's performance.

The remainder of this paper is structured as follows. We discuss related work in Sec. 2. Next, we introduce our system architecture and introduce algorithms in Sec. 3. We show results from a study with several test subjects for tracking accuracy and the ML classifier's quality prediction in Sec. 4. Sec. 5 concludes the paper.

2 Related Work

We discuss related work that uses augmented reality for worker *guidance*, process *monitoring*, and tracking (*quality assurance*).

Guidance. The research field on using AR to train and guide workers is well represented by the meta-study by Werrlich et al. [20]. The authors present 17 different studies on this topic and evaluate their results. Most of them conclude that training with AR systems improves the quality (error susceptibility), the short- and long-term memory, skill-transfer, and user satisfaction, but the training can also take longer to complete. The structure of the application is important, e.g., step-by-step solutions seem to be less suitable for training, as they may make the trainees dependent on the instructions instead of learning the skill. To improve the understanding of how AR improves training efficiency and quality over paper-based training, Werrlich et al. [19] compare paper-based with a HMD-based training for manual assembly tasks, and come to the result that participants perform significantly faster but also significantly worse using paper-based instructions. Furthermore, all trainees preferred HMD-based learning for future assembly trainings. These encouraging findings show the clear benefits of AR for training and guidance, and we incorporate them into the design of our own guidance system, that additionally monitors the quality of work to provide feedback.

Evans et al. [8] also evaluate AR for assembly and present design guidelines. Their results confirm that it is possible to guide specifically assembly processes via AR, but show that tracking capabilities of HMD's are limiting this use-case. We also incorporate their design guidelines, confirm the HMD's limitations experimentally, and propose an improved tracking approach, that together with additional sensing capabilities enable AR for assembly monitoring.

Monitoring. Various methods for progress monitoring of manufacturing processes and objects were recently investigated. Omar et al. [16] found that progress monitoring, e.g., in construction processes, to detect malfunctions, defects, and errors as early as possible, reduces expensive rework. They also discuss AR-based sensing and visualization approaches that build on external RGB camera sensors that capture additional information about work processes and render feedback to the AR. However, since they use camera-based sensors, the authors also show several limitations such as accurate and reliable image registration and interference, that complicate the visualization, especially in outdoor construction places. Bekel et al. [5] and Shreve et al. [18] use AR HMDs to acquire ground truth image data and the labels of real-world objects. They either label image patches that are collected while free-roaming using self-organizing maps [5], or sample images from labelled 3D reconstructions of objects [18] that the AR HMD captured. Both works aim to collect visual data about objects, that then allow down-stream training of classification algorithms. These works are tangential to RGB-based process monitoring and introduce novel perception methods, however, they again only collect data on visual properties, and do not extend to dynamically changing objects, e.g., in assembly, easily. To cope with typical RGB limitations, Sawaga et al. [17] use an external depth sensor (RGBD), attached to an AR HMD, to track the hands of a worker and to compare a 3D reconstruction of the environment to a corresponding ground truth one in a timely, immediate manner. They show that their RGBD-based approach can monitor the progress of a manual labor task, based on the reconstruction error, more reliably than with a RGB sensor. While this approach is able to detect coarse changes of the processed objects, it is limited to visual properties. Hence, they cannot include non-visual cues about work processes or objects, e.g., the torque of a screwdriver, that are important for many assembly tasks. In contrast, our system provides this additional insight for every task of the process, and also gives accurate and reliable feedback to workers.

While our approach builds upon previous findings (as we exploit AR to render a guidance story that we can adapt to any work process), we think that for process monitoring and quality assurance the prior limitations of optical sensors require a solution with orthogonal sensor *views*. Hence, our framework (1) uses optical sensors (built-in HoloLens sensors) to sense coarse-grained location information of a worker, a work tool, and a workpiece along the whole work *process*, (2) senses additional fine-grained monitoring information about a *task* with an IMU (mounted at a work tool), (3) exploits ML to predict quality measures of a fine-grained *action* from these raw sensor data, e.g., to approximate non-visible properties like screw torque in an assembly application, and (4) renders enriched guidance feedback that we derive from these quality measures to AR to *guide* the worker and optimize the work process.

3 Method

Fig. 1 shows our processing pipeline. We use a commercial AR HMD system for worker *guidance* (see Sec. 3.2) to both visualize and instruct work process tasks and use its positional tracking (see Sec. 3.1) of the worker, a work tool, and a workpiece to derive process information to *monitor* it, and to predict its quality. We define specific quality metrics for assembly tasks (see Sec. 3.3) and train a Decision Tree classifier that allows to monitor the quality of work (see Sec. 3.4) for immediate feedback that feeds back into the worker guidance, e.g., alerts the users to mistakes. To evaluate our method, we perform a case study: we equip hand-held work tools with sensor modules [14] that classify sensor values, see Fig. 2 (left) and perform an exemplary manufacturing process with an engine block, see Fig. 2 (right).

3.1 Tracking

Our positional tracking approach registers and stabilizes the Microsoft HoloLens tracking system. Thus, we use Vuforia [12] and encoded optical markers (*targets*) to reliably identify the current work process and accurately re/calibrate our system. We attach encoded markers to both the hand-held tool and the workpiece, see

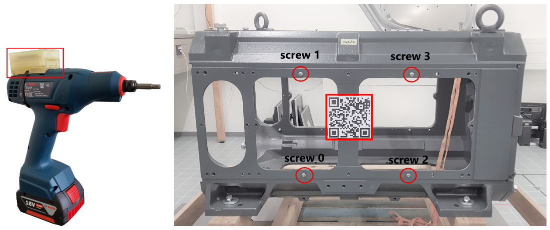

Figure 2: Electric screwdriver with attached sensor module (red circle) and motor-block with four screws (red circle) and encoded marker (red rectangle).

Fig. 2, and use HoloLens' built-in simultaneous localization and mapping (SLAM) to provide a precise mapping of the virtual instructions to the corresponding environment whenever the targets leave the field of view, by registering each target's position within the environment map.

We use Vuforia's proprietary tracker, that detects the encoded marker on the RGB camera image, and calculates its position and rotation. It then automatically transforms these properties into the HoloLens' internal spatial mapping coordinate space. The registration in the environment is then ensured by HoloLens, where its integrated SLAM tracks the environment via RGB-D sensors [10]. When a target re-appears after occlusion or loss of tracking, the system detects and registers the target, eliminating holographic drift of the point of interest that occurred in the meantime.

We use QR-codes of as encoded markers to easily and reliably identify the tool, the workpiece and its associated points of interest, i.e., the process with its tasks. The detection of markers is crucial for correctly visualizing the AR guidance. Hence, to ensure reliable detection, we use a 14 cm large marker for the workpiece, following Vuforia's recommendation for an assumed 65 cm distance between camera and object. However, for the tool we reduce this to 10 cm to compromise between accuracy and practical usability. The positions of points of interest, e.g., the screw threads in an assembly use-case, are inferred relative to the coded marker on the workpiece that encodes a position.

Thus, we continuously visualize each component's position to guide a worker through the steps in a process. Note that we still visualize overlays to the current screw even if the target of the current workpiece is no longer within the viewing port of the user.

3.2 Worker Guidance

Based on the findings of Werrlich et al. [20], we design our worker guidance system visualizations to be simple and consistent and enable an intuitive progression by giving users enough time to act and react. Fig. 3 depicts an exemplary construction process in 9 tasks: (1-8) un-/tightening of screws and (9) screwing in the air once, together with appropriate time windows for preparing actions and performing them. According to Evans et al. [8], we avoid distractions, as we display any visualization in a simple and readable manner, and hence, limit the color pallet to a fixed set of colors consistently encoding different types of tasks, e.g., only the colors cyan, magenta, and yellow, see Fig. 5 (top). We represent the current point of interest, e.g., a screw, by a yellow ring, and instruction panels in magenta.

Since we formulate our processing pipeline as a finite state machine, to complete a work process, the user must finish a sequence of tasks. Users progress through tasks sequentially either using a click gesture or by directing their gaze at a decision panel for a certain amount of time, see Fig. 5. Click is used for instruction panels, and gaze is used to start actions, i.e., when the hands hold a hand-held tool and gestures cannot be performed.

The HMD tracks the coarse-grained location of the work tool at the workpiece to identify, guide and monitor the current process stage. For instance, during a complete process, we use the coarsely tracked positions, to determine each task, such as `tightening` screw 1 to 8 in an assembly process, and also to synchronize the time frames of both the HoloLens tracking system and the external IMU data stream that we then use to control the quality of each individual task. Hence, we use countdowns of 3s before transitioning into each task, to give users enough time to position the tool correctly. At the end of the countdown a **GO!** visualization signalizes the user to start. Since we start recording the external IMU data stream 3 seconds before each task starts, we ensure that we capture all data that represents the full time range of a task for more precise ML-based analysis.

3.3 Predicting Quality Metrics

To solve the limitations of current monitoring systems reported by [8,17], we employ an additional external IMU on a sensor module to capture additional data on a task. We first acquire training data and corresponding ground truth labels to train a classifier that predicts a quality metric for each task of a work process by processing the raw time series data in a live-phase later on. Hence, in the live phase, the worker guidance system provides both the expected logical label, e.g., screwing clockwise, and coarse-grained time interval for each task. The IMU enables different quality metrics that we derive for each work process individually.

We implement three hierarchical levels of metrics, see Fig. 4: The first level is based on the HMD's *coarse-grained location tracking* through the complete process via the worker guidance system. Since every task (e.g., screw 1 to 8) in a process is performed exactly once in the state machine, its assigned location and order are deterministic and thus, we can use the coarse-grained location information to identify each task. The second level employs the external IMU for *fine-grained monitoring* of the execution of each task. Hence the external sensor module is used when the HMD cannot reliably detect the task. For instance, the HMD fails to register suitable characteristics to detect a tool's action, but the external sensor module reliably detects them via magnetic fields or acceleration forces. The third level monitors the quality of a task's execution. Thus, each task is associated with a set of constraints (i.e., *quality metrics*) that the external sensor module detects, such as the duration of the action or the tool's angle-of-attack. In our exemplary assembly study, we use a single quality metric that describes whether the correct torque was reached. This is predetermined by the duration of the configured rotation speed of the screw driver, the screw, and the screw thread at the workpiece.

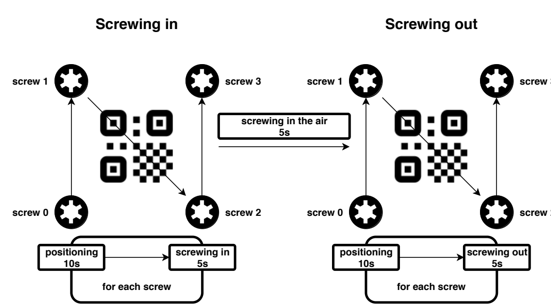

Figure 3: Worker guidance example with sequential work steps, e.g., positioning a tool and executing of tasks such as tightening screws. *Screwing in the air* is executed position-independent.

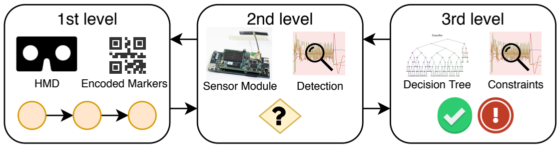

Figure 4: Our quality metric detection in three stages: Process level *coarse-grained monitoring*, *fine-grained monitoring* of the task, and *fine-grained monitoring* of each action.

3.4 ML-aided Analysis

In line with previous work [14], we use the same DTC approach together with a set of approx. 32 IMU-based features, e.g., norm of magnetic field or maximum of acceleration given a data window, to sufficiently and reliably detect each task in a process. By clearance, we can even distinguish between various sub-classes of actions of each task. However, for simplicity, in our case-study, we only train a DTC on a small (114 seconds, 96 tool actions), separately recorded dataset, that is publicly available[1], and that was labeled using the NOVA [4] tool, to predict the quality metrics, i.e., detection and measuring of duration of each task. We use this DTC to process IMU data on the external sensor module during the worker guidance.

The **training** is based on the AutoML pipeline presented in [14]. We use a sliding window on class-balanced data, extract and select features, and train a DTC using cross-validation and grid-search to optimize its hyperparameters. Given a dataset, we subdivide the time series data into equally distributed classes of sub-windows of $0.02s$ length with an overlap of 50%. In a preliminary study we found that these parameters achieve the highest accuracy over the variable-length actions of electric screwdrivers. For each window, we *extract* and *select* a set of features, inspired by [14]. Hence, from the 32 unique features we select those 16 features that represent the tasks best, using Random Forest feature importance and eliminating features that are correlated or add no information, using the Pearson correlation test. We use 5-fold cross-validation to train a robust classifier given the selected features, and a grid-search to tune the model's hyper-parameters, i.e., number of different features and tree depth, resulting in a DTC with 21 nodes.

4 Experimental Evaluation

We first describe the experimental setup, the study design, and present benchmarks for tracking (guidance) and monitoring.

4.1 Hardware Setup.

For our experiments, we use a Microsoft HoloLens (Unity 2017.4.30f1 with HoloToolKit 2017.4.3.0 and Vuforia 7.0.57), an electric screwdriver (Bosch Exact ION 12-700, programmable constant speed control set to 300 min^{-1} clockwise and 736 min^{-1} counter clockwise) and a workpiece, a large engine block, see Fig. 2. We use the millimeter-accurate ART [3] optical reference positioning system to validate the tracking capabilities of our system (MAE: Mean Absolute Error; SD: standard deviation; location: MAE=0.1 mm; min=0.001 mm; max=3.2 mm SD=0.54 mm, orientation: MAE=0.01°; min=0.001°; max=0.2°; SD=0.06°). ART employs 9 infra-red (IR) cameras that cover an area of $10\,\text{m}\times10\,\text{m}\times3\,\text{m}=300\,\text{m}^3$ using reflective markers that are attached to each of our system components.

We design the AR worker guidance system around a typical car assembly process and perform the tasks `tightening`, `untightening`, and `air screwing`. The HMD tracks the positions of tool and tasks (screw thread) relative to each other, i.e., point of interest on the workpiece, and the sensor module that we attach to the tool classifies actions. We visualize the order of the tasks and the workpiece in Fig. 2, and show an exemplary task in Fig. 5.

[1] https://github.com/mutschcr/tool-tracking

4.2 Study Design

We recruited 6 test subjects (male: 5; female: 1; average age: 24) with varying levels of previous experience with AR or professional powered tools (but all experienced amateurs).

We designed the study around a typical assembly process, see Fig. 3, with three different types of repeated tasks performed on an engine block: 4× tightening (clockwise), once screwing in the air and 4× untightening (counter-clockwise). By having the experiment repeated by a large set of participants that were not involved in the system's development and in the classifier training, we produce more reliable estimates for the tracking, classification accuracy and the system's overall applicability. We used artificial room lighting. All test subjects started under the same preconditions after a short period of familiarization with the HMD, and the tool, and worked without disturbance in a video-recorded laboratory setting. The average duration per experiment is approximately 469 s (min=315 s;max=572 s), and varies due to, we believe, higher cognitive load for less experienced users. From the study's results, i.e., precise reference location tracking and video recordings, we extract ground-truth position and task classification data, that we use in the following evaluation to validate our system. Note that we describe the study design for each benchmark again individually in Sec. 4.3.

4.3 Benchmarks

We describe the results of our method along its performance in coarse-grained guidance, i.e., its accuracy to locate each task correctly, and along its performance in the fine-grained monitoring, i.e., its accuracy to predict each task's (*action*) quality.

4.3.1 Coarse-grained Guidance Benchmark

To assess the first-level quality metrics (i.e., progress through the worker guidance) we have to evaluate the performance of our enhanced AR tracking system (fusion of HoloLens and Vuforia). Our study design is inspired by Feigl et al. [10], who describe an elaborate setup, where the HoloLens performs reliably in static scenarios as it keeps track of problematic ego-motion and its distance to other (tracked) objects. We record three different scenarios at 20 s each to evaluate the tracking accuracy: no device in motion, only tool in motion, and both HMD and tool in motion.

No device in motion. We statically place the HMD at a distance of 1 m from the workpiece. We also fix the hand-held tool in 30 cm, 60 cm, and 90 cm distance from the HoloLens, and report the worst-case results at a distance of 90 cm. The root mean squared error (RMSE) is between 0.016 mm and 0.058 mm and has a SD of between 0.249 mm to 0.671 mm. Both the error and the SD increase slightly with distance. The ranges of these results are overlapping the reference system's precision of 0.1 mm, but show the enhanced tracking's principle feasibility to locate coarse-grained tasks within a process.

Tool in motion. To investigate the effects of steady motion of the hand-held tool, we place the HoloLens as well as the tool at the same height. The HoloLens is at a distance of 80 cm from the workpiece, i.e., about an arm's length away. To evaluate tracking for simple movements we move the tool slowly and steadily from the left to the right in 20 cm, 40 cm, and 60 cm distance from the HMD, and additionally back to front, one time each for 20 s. The tool motion causes a significant increase in the RMSE (average of 3.948 mm) for back to front movements, and up to 5.87 mm for the 20 cm distance. We show the measured distances over time of our method and the reference system in Fig. 6 (top), together with the absolute error in the lower plot. While the error remains relatively low over the duration of the experiment, we can observe a loss of the tracking caused by the limited field-of-view of the camera sensor,

Figure 5: Visualization of one step from a larger process: First, the user is instructed on the next step. Then when ready, the user positions the tool within a given time frame, and finally performs the task. We use the direction of gaze and countdowns for progressing through the process guidance system.

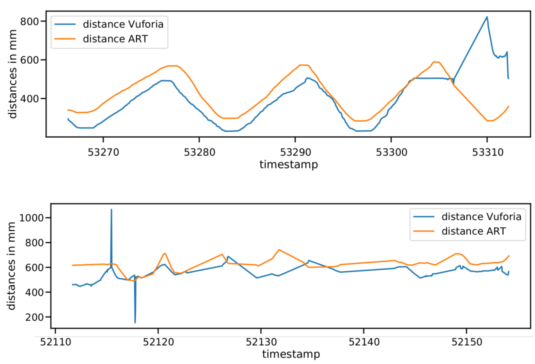

Figure 6: Top: Tool moves back to front in a range of 80*cm*. Bottom: Random slow movement of tool with SLAM enabled.

that is visible around 53.310 s. We think that this may be addressed internally by detecting the loss of tracking and handling it in the framework, or externally by asking users to keep their gaze on the tool when that becomes necessary during guidance.

HMD and tool in motion. We record a simplified work task that resembles a coarse-grained guidance task: The test subject wears the HMD and handles the tool at the extended arm, while standing frontally in a distance of 80 cm to a workpiece. We mostly fix the translational ego motion of the HMD but allow the head to move naturally, e.g., to follow the hand movement, and move the tool dynamically. We evaluate the enhanced AR tracking with and without SLAM enabled.

With an average error of 4.93 mm, the accuracy of our enhanced tracking (with SLAM) is comparable to the previous tests with slow, steady motion. However, the more dynamic motion and subsequently optical artifacts registering on the RGB sensor, Vuforia locates the coded markers falsely. We present this anomalous behavior in Fig. 6 (bottom) (between 52.115 s and 52.118 s). The error's SD in the best-case dynamic motion scenario is 67.37 mm. Without SLAM, however, the performance is predictably worse. SLAM helps to decrease the RMSE for dynamic motion from 5.397 mm to 4.93 mm (SD from 77.03 mm to 67.37 mm).

Label	Action	No Action
Precision	93.7 (±1.4)	99.9 (±0.1)
Recall	95.6 (±2.4)	99.9 (±0.1)
Accuracy	99.8 (±0.1)	99.8 (±0.1)
F1	94.6 (±1.7)	99.9 (±0.1)

Table 1: Confusion matrix with average values for 6 test subjects and actions detected by Decision Tree classifier.

Our studies show that our enhanced AR tracking system reliably locates the tool and workpiece even with dynamic motion, with an RMSE of 4.93 mm (SD=67.37 mm). Therefore, it provides information that is accurate enough to identify each task of a process, but only if each task's proximity to its closest neighbors is further away than one SD (e.g. 67.37 mm). The observed error is low enough to identify a task, but it cannot detect an action, i.e., its occurrence or start and stop time, that we need for quality metrics.

4.3.2 Fine-grained Controlling Benchmark

We use our external sensor module (with IMU) to derive a second- and third-level quality metrics, for the fine-grained part of our monitoring system. For this evaluation, we first train a DTC as described in Sec. 3.4. The classifier achieves an accuracy of 98% for action detection on its separate training dataset. Note that the HMD narrows down the sensor module's IMU data into coarse windows, one for each task, before ML inference takes place, and also hints logical labels that determine task specific quality metrics (constraints). Our classifier then predicts whether tasks occur and estimates their duration as a metric for process quality. In this evaluation, we compare predictions with ground truth labels that we annotated using video recordings and the NOVA [4] toolkit.

Second-level quality metric. We test the DTC's performance for detecting any type of task involving the tool during our assembly study with 6 test subjects, see Sec. 4.2 for details on the study design. The DTC predicts on short windows of 0.02 s length with 50% overlap, which adds 0.01 s uncertainty. A post-processing step collects consecutive predictions and filters false predictions that are shorter than 0.3 s. Table 1 lists the average values of precision, recall, accuracy, and F1-score together with their SDs across all test

Label	`tightening`	`untightening`	`airscrewing`
Precision	91.3 (±4.0)	88.6 (±4.3)	99.3 (±0.8)
Recall	92.3 (±3.1)	94.3 (±6.7)	97.8 (±2.0)
Accuracy	99.9 (±0.0)	99.9 (±0.1)	100 (±0.1)
F1	91.6 (±1.1)	91.2 (±4.6)	98.6 (±1.0)

Table 2: Confusion matrix with averages for 6 test subjects: Each logical action has different classification results, due to the diverse IMU data that characterize each action. Air screwing causes near to no distractions, whereas tool impacts on a workpiece water down the classifiers clear decision boundary for the other classes.

subjects. The average accuracy score for detecting tasks is about 99.8%, with an F1-score of 94.6%. The lower precision means that the classifier mixes up false positives, e.g., knocking tool against workpiece, with real tasks to a low degree. However, most false positives are still acceptable (as we know they are short we can easily filter them out) and no true tasks (that are usually hundreds of milliseconds long) were missed. In this regard, recall is slightly higher with 95.6%, meaning that tasks do not remain undetected.

Third-level quality metrics. We further separate the tasks into their known ground-truth action classes, i.e., `tightening`, `untightening` and `air screwing`, and focus on their duration as the core quality metric for our example application. The system achieves a very high accuracy for all classes, see Table 2. Due to the relatively low precision for `tightening` and `untightening`, their duration is slightly overestimated. On the flip side, the system generally does not underestimate action duration due to false negatives, with recall values of 92.3%, 94.3% and 97.9%. To go into further detail, the DTC performs better for `air screwing` because there are no physical impacts of the tool onto the workpiece that confuse the DTC. We conclude that we can reliably determine our action duration-based quality metric. However, for other tool types, e.g., pneumatic screwdrivers, that can have vastly different sensor characteristics with a long run-down phase after completing the action, another ML approach, e.g., with class specific training for each action, is required, as purely detecting activity is insufficient [14].

The overall results show that the system performs well for each of the three levels of our proposed quality metrics. The AR worker guidance can be enhanced with additional sensor modules to enable more advanced quality metrics that a DTC can predict from unknown sensor data from both the HMD itself and an external sensor.

5 Conclusion

We propose to enhance AR worker guidance with automatic quality assurance by using low-cost external inertial measurement units and machine learning. We monitor the process via the HMD's positional tracking, and use sensor modules to collect additional data that we analyze using ML, enhancing the sensing capabilities of AR worker guidance with new insights, for individual tasks of a process.

We implemented an exemplary assembly use-case and attach a sensor module to a hand-held tool to evaluate three levels of quality metrics. While the HMD's positional tracking is already good enough for task-by-task guidance, the combination with the sensor modules greatly enhances the system: we detect actions precisely with an accuracy of 99.9%, and predict fine-grained quality metrics reliably with a class specific recall between 92.3% and 97.8%.

Acknowledgements

This work was supported by the Bavarian Ministry of Economic Affairs, Infrastructure, Energy and Technology as part of the Bavarian project Leistungszentrum Elektroniksysteme (LZE) and through the Center for Analytics-Data-Applications (ADA-Center) within the framework of "BAYERN DIGITAL II".

References

[1] Atlas copco catalogue. `https://www.atlascopco.com/content/dam/atlas-copco/industrial-technique/mvi/documents/Atlas_Copco_smart_connected_assembly_roadshow_screen.pdf`. Accessed: 2020-07-22.

[2] Bosch catalogue. `https://www.boschrexroth.com/en/us/products/product-groups/tightening-technology/topics/nexo-cordless-wi-fi-nutrunner/index`. Accessed: 2020-07-22.

[3] W. Armin and U. Probost. Art, advanced realtime tracking. `https://ar-tracking.com/`. Accessed: 2020-01-25.

[4] T. Baur, A. Heimerl, F. Lingenfelser, J. Wagner, M. F. Valstar, B. Schuller, and E. André. explainable cooperative machine learning with nova. *KI - Künstliche Intelligenz*, Jan 2020.

[5] H. Bekel, G. Heidemann, and H. Ritter. Interactive image data labeling using self-organizing maps in an augmented reality scenario. *Neural Networks*, 18(5-6):566–574, 2005.

[6] G. Boothroyd. *Assembly automation and product design*. CRC Press, 2005.

[7] J. Y. Chen and G. Fragomeni. *Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation: 10th Intl. Conf., VAMR, Held as Part of HCI Intl. Las Vegas, NV, 2018*.

[8] G. Evans, J. Miller, M. I. Pena, A. MacAllister, and E. Winer. Evaluating the microsoft hololens through an augmented reality assembly application. In *Degraded Environments: Sensing, Processing, and Display*, vol. 10197, 2017.

[9] M. G. Fard and F. Pena-Mora. Application of visualization techniques for construction progress monitoring. In *Intl. Workshop on Computing in Civil Engineering*, pp. 216–223. Pittsburgh, PA, 2007.

[10] T. Feigl, A. Porada, S. Steiner, C. Löffler, C. Mutschler, and M. Philippsen. Localization limitations of arcore, arkit, and hololens in dynamic large-scale industry environments. In *Proc. Intl. Conf. Computer Graphics Theory and Applications*. Malta, Malta, 2020.

[11] T. Frantz, B. Jansen, J. Duerinck, and J. Vandemeulebroucke. Augmenting microsoft's hololens with vuforia tracking for neuronavigation. *Healthcare technology letters*, 5(5):221–225, 2018.

[12] H. JIM. Vuforia engine developer portal. `https://developer.vuforia.com/`. Accessed: 2020-01-25.

[13] C. Liu, S. Cao, W. Tse, and X. Xu. Augmented reality-assisted intelligent window for cyber-physical machine tools. *Journal of Manufacturing Systems*, 44:280–286, 2017.

[14] C. Löffler, C. Nickel, C. Sobel, D. Dzibela, J. Braat, B. Gruhler, P. Woller, N. Witt, and C. Mutschler. Automated quality assurance for hand-held tools via embedded classification and automl. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Ghent, Belgium, 2020.

[15] K. C. Mathi. *Augment HoloLens' Body Recognition and Tracking Capabilities Using Kinect*. PhD thesis, Wright State University, 2016.

[16] T. Omar and M. L. Nehdi. Data acquisition technologies for construction progress tracking. *Autom. in Construction*, 70:143–155, 2016.

[17] H. Sagawa, H. Nagayoshi, H. Kiyomizu, and T. Kurihara. [poster] hands-free ar work support system monitoring work progress with point-cloud data processing. In *2015 IEEE International Symposium on Mixed and Augmented Reality*, pp. 172–173, 2015.

[18] M. A. Shreve, S. K. P. Kumar, J. Sun, G. R. Gavai, R. R. Price, and H. M. Eldardiry. System and method using augmented reality for efficient collection of training data for machine learning, May 2 2019. US Patent App. 15/826,588.

[19] S. Werrlich, A. Daniel, A. Ginger, P.-A. Nguyen, and G. Notni. Comparing hmd-based and paper-based training. In *IEEE Intl. Symp. on Mixed and Augmented Reality (ISMAR)*, pp. 134–142, 2018.

[20] S. Werrlich, E. Eichstetter, K. Nitsche, and G. Notni. An overview of evaluations using augmented reality for assembly training tasks. *Intl. J. Computer and Information Engineering*, 11(10):1068–1074, 2017.

[21] P. Wozniak, O. Vauderwange, A. Mandal, N. Javahiraly, and D. Curticapean. Possible applications of the leap motion controller for more interactive simulated experiments in augmented or virtual reality. In *Optics Education and Outreach IV*, vol. 9946, p. 99460P, 2016.