



ELSEVIER

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

ScienceDirect



[www.elsevier.com/locate/procedia](http://www.elsevier.com/locate/procedia)

Procedia CIRP 00 (2024) 000–000

57th CIRP Conference on Manufacturing Systems 2024 (CMS 2024)

# Human-AI Interaction in Industrial Robotics: Design and Empirical Evaluation of a User Interface for Explainable AI-Based Robot Program Optimization

Benjamin Alt<sup>\*,a,b</sup>, Johannes Zahn<sup>a</sup>, Claudius Kienle<sup>a</sup>, Julia Dvorak<sup>c</sup>, Marvin May<sup>c</sup>, Darko Katic<sup>a</sup>, Rainer Jäkel<sup>a</sup>, Tobias Kopp<sup>d</sup>, Michael Beetz<sup>b</sup>, Gisela Lanza<sup>c</sup>

<sup>a</sup>*ArtiMinds Robotics, Albert-Nestler-Str. 11, 76131 Karlsruhe, Germany*

<sup>b</sup>*Institute for Artificial Intelligence, University of Bremen, Am Fallturm 1, 28359 Bremen, Germany*

<sup>c</sup>*wbk Institute of Production Science, Karlsruhe Institute of Technology, Gotthard-Franz-Straße 5, 76131 Karlsruhe, Germany*

<sup>d</sup>*Institute for Learning and Innovation in Networks, Karlsruhe University of Applied Sciences, Moltkestraße 30, 76133 Karlsruhe, Germany*

\* Corresponding author. Tel.: +49 721 509998-66. E-mail address: benjamin.alt@uni-bremen.de

arXiv:2404.19349v1 [cs.RO] 30 Apr 2024

## Abstract

While recent advances in deep learning have demonstrated its transformative potential, its adoption for real-world manufacturing applications remains limited. We present an Explanation User Interface (XUI) for a state-of-the-art deep learning-based robot program optimizer which provides both naive and expert users with different user experiences depending on their skill level, as well as Explainable AI (XAI) features to facilitate the application of deep learning methods in real-world applications. To evaluate the impact of the XUI on task performance, user satisfaction and cognitive load, we present the results of a preliminary user survey and propose a study design for a large-scale follow-up study.

© 2024 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of the scientific committee of the 57th CIRP Conference on Manufacturing Systems 2024 (CMS 2024).

**Keywords:** explainable artificial intelligence; explanation user interfaces; deep learning; industrial robotics; manufacturing; user study

## 1. Introduction

The past two years have seen breakthrough success of deep learning (DL) and related artificial intelligence (AI) approaches in a variety of domains such as natural language and image processing. In industrial robotics, AI promises to leverage large amounts of data to automatically optimize robot programs [1] or enable seamless human-robot collaboration [2], saving costs, addressing shortages of qualified labor and increasing the precision or robustness of robot-based production processes [3, 4]. Despite this potential, AI in the manufacturing industry remains far from widespread [5]. With the introduction of digital technologies in manufacturing such as Enterprise Resource Planning (ERP) and Manufacturing Execution Systems (MESs), a “skill gap” between the IT skills required to configure, operate and maintain the novel systems and the lack of such skills among the workforce slowed the digital transformation of the manufacturing industry [6]. A similar skill gap is evident for

AI technologies, with workers lacking the required skills and experience in mathematics, programming or data science to deploy, interact with or understand state-of-the-art AI methods [7]. Explainable artificial intelligence (XAI) has been identified as a key component of increasing industry adoption of AI, with interpretability tools considered a “main catalyst” of industrial AI [5]. To be useful in practical applications, XAI methods must be paired with Explanation User Interfaces (XUIs) to display explanations and facilitate user interaction [8]. Several principles for the systematic development of XUIs have been proposed [9]. However, comparatively few real-world XUI systems have been studied from a human-machine interaction (HMI) perspective. Studies of XUIs for AI agents in strategy games [10] and loan applications [11] are notable examples. While in the manufacturing domain, XAI methods have been proposed for a wide range of applications such as forecasting, decision making and job scheduling [12], to our knowledge, no study of XUIs for robot programming has yet been undertaken.

In this paper, we present an XUI for a state-of-the-art DL-based method for optimizing industrial robot programs and de-

2212-8271 © 2024 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of the scientific committee of the 57th CIRP Conference on Manufacturing Systems 2024 (CMS 2024).

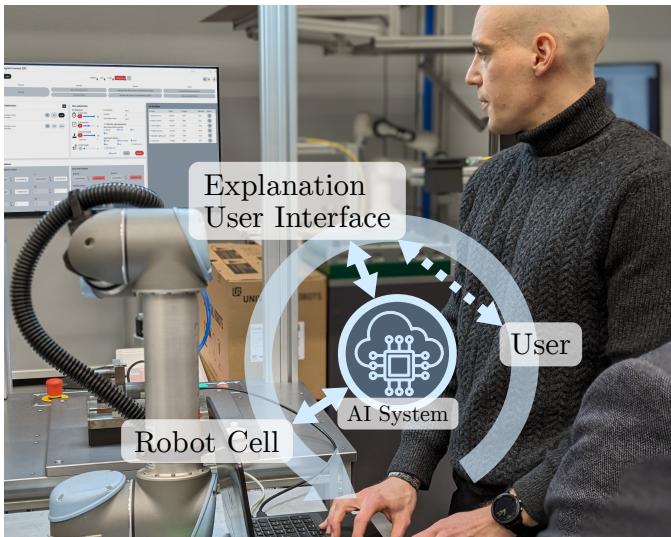


Fig. 1. Overview of the proposed system: A user interface enables intuitive interaction of a human user with an AI system for robot program optimization.

scribe its underlying design principles. We present a preliminary, small-scale user study on a realistic assembly use case, which evaluates to what extent our proposed XUI enables industry practitioners to use AI for robot program optimization. Lastly, we propose a study design for a planned large-scale user study to evaluate the potential of XUI methods in industrial manufacturing.

## 2. An Explanation User Interface for AI-based Robot Program Optimization

Industrial robots are currently programmed by experts using textual or graphical programming interfaces. Particularly the task-specific parametrization of robot programs during the ramp-up phase of robot workcells has been identified as a major cost factor in industrial automation [13]. We present an XUI for an AI-based robot program optimizer to enable industry practitioners the use of state-of-the-art AI methods (see Fig. 1).

### 2.1. AI-based Robot Program Optimization

Shadow Program Inversion (SPI) is a novel, deep-learning-based approach for optimizing robot program parameters [1, 3]. Given a parameterized robot program  $P$  with parameters  $x$ , the execution of this program will result in a robot trajectory  $\Theta$ . Initially, SPI learns a *shadow model*  $\hat{P}$  of the program: A differentiable, neural model, which is trained on data from executions of  $P$  to predict the expected execution of  $P$  given some parameters  $x$ . It then uses  $\hat{P}$  in a model-based iterative first-order optimization process to optimize the parameters  $x$  to maximize a task-specific objective function. We refer to the literature for a detailed description of the approach [1, 3]. From a user's perspective, SPI requires the following three-step workflow:

- 1. Dataset definition:** The training data for  $\hat{P}$  consists of input-label pairs  $(x, \Theta)$ . For AI novices, it is challenging

to determine whether a given dataset is suitable for training. Poor data quality due to e.g. outliers can severely impact optimization results. One aspect generally missed by AI novices is the requirement for the inputs  $x$  to cover the range of the parameter space over which optimization is to be performed, and to have sufficient variance.

- 2. Model training:**  $\hat{P}$  contains neural networks, which must be trained. Challenges involve the choice of appropriate hyperparameters such as learning rates or network sizes, depending on the task. After a model has finished training, it is challenging for non-experts to access the quality of the trained model.

- 3. Parameter optimization:** The program parameters are optimized with respect to a task-specific objective function. At the time of writing, SPI supports process metrics such as cycle time, path length and task success probability, as well as a threshold on the forces that are allowed to occur during program execution, and arbitrary weighted combinations of these objectives. Choosing the appropriate objective function(s) and weights is challenging for non-experts, as is the choice of appropriate hyperparameters for the optimizer. The assessment of the quality of the optimization results is likewise challenging for naive users, as the relationship between program parameters and robot behavior is not always immediately obvious, depending on the parameter.

### 2.2. Guiding Principles for UI Design

A UI is required to enable the practical use of a complex AI approach such as SPI. We propose that to enable AI novices to use SPI and similar systems, UI designers ought to emphasize *user adaptability* and *explainability*, in addition to general UI design principles such as informative feedback, consistency, minimal memory load and user control [14].

#### 2.2.1. User Adaptability

The skill gap between the level of algorithmic and data competence required to use complex AI systems and the lack of experience of practitioners in industry with such systems often prevents the practical use of AI [7]. *Adaptable user interfaces*, where the UI can be reconfigured to adapt to the needs of users, can help bridge this gap [15]. We propose an adaptation mechanism by which the user can switch between “Guided” and “Expert” modes of the UI. In Guided mode, several features which give the user direct control over the AI system, such as hyperparameter selection, are simplified or hidden and replaced by defaults. UI elements directly visualizing technical aspects of the AI system, such as loss curves, are simplified, removed or replaced by more intuitive, but less information-dense metrics, and additional textual explanations are added. In Expert mode, all control options, such as training and optimization hyperparameters, are exposed to the user; plots have a higher degree of information density, and all default settings can be manually overridden. Users can freely toggle between these modes.

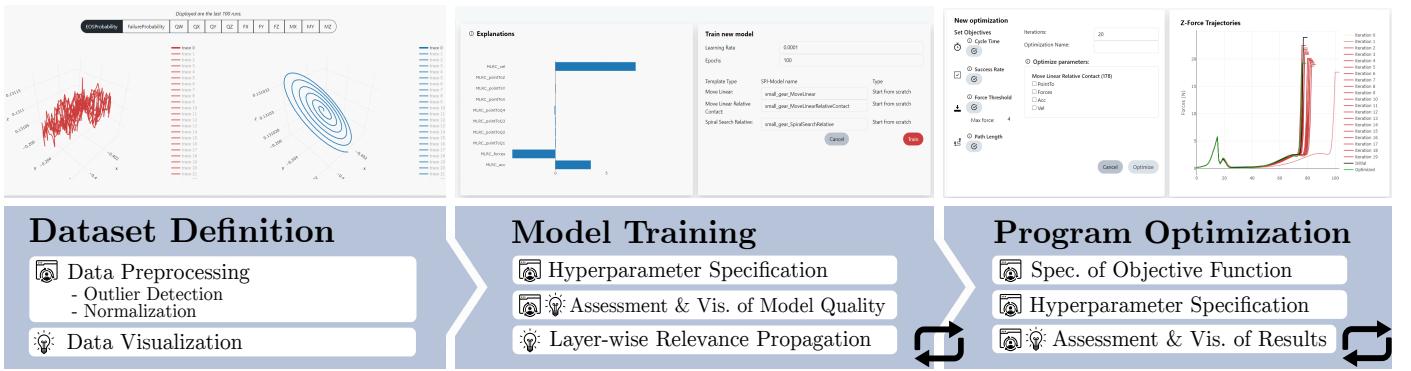


Fig. 2. Workflow and corresponding UI elements, with variation points for user adaptability (Ⓐ) and explainability features (Ⓑ). Screenshots for data visualization (left), LRP (center left, bars illustrate the relative impact of individual program parameters on model output), hyperparameter specification (center right), specification of the optimization objective (2nd from right) and visualization of optimization results (far right) are shown.

### 2.2.2. Explainability

Explainability has been identified as a crucial factor in human-AI interaction [8], significantly improving both trust in the system as well as task success [16]. We conceptualize the proposed interface as an XUI, integrating explainability features at every step of the workflow (see Fig. 2). We take a broad view of XAI that includes both explanations of the behavior of the deep neural networks (DNNs) at the heart of the system, but also the input-output relationships of the system as a whole. The following paragraphs outline the integrated XAI features at each step of the workflow.

### 2.3. XUI Workflow

The UI was realized as a component of ArtiMinds Learning & Analytics for Robots (LAR), a software platform for the collection and analysis of robot data. Robot programs are assumed to have been created through ArtiMinds Robot Programming Suite (RPS), an integrated development environment (IDE) for industrial robots based on a task-based programming paradigm. When an RPS program is executed on a robot, data such as end-effector poses, joint states or forces and torques are streamed to LAR and automatically annotated with semantic information about the currently executed robot skill, as well as user-defined tags such as serial numbers or part variants. The UI guides the user through the SPI workflow outlined in Section 2.1. Fig. 2 illustrates the workflow and the realized adaptability and explanation features at each step. At the beginning of the interaction, the user selects the robot program and one or more robot skills (the *target skills*) in the program to optimize (the *target program*). Once a workflow step has been completed, the user can press “Next” to continue to the next workflow step, and can return to completed workflow steps at any time to make changes.

#### 2.3.1. Dataset definition

To facilitate the definition of a suitable dataset for model training, the user is provided with UI elements for exploring and visualizing collected data. To prevent the user from selecting invalid data, only data collected from executions of the target program are displayed. The user can filter training data by

timestamp or any tags applied to the data at collection time. In Expert mode, the user can adjust further technical details of the dataset, such as the length with which trajectories are padded. Visualizations such as 3D plots of spatial data are displayed to provide the user with an understanding of the data. The challenges most frequently encountered when specifying datasets for SPI are a lack of variance in the inputs, the presence of outliers in the labels, as well as the definition of successful skill executions. To allow users to quickly gauge the variance in the inputs, the input distribution of the current data selection is displayed as box plots. Additionally, for each input parameter, a textual message is displayed informing the user whether the variance in that parameter is sufficient to allow later optimization of that parameter. The trajectory distribution is displayed in two plots, the first shows the end effector motion as a 3D line plot, the second displays the force-torque distribution over time. The trajectories are colored by their success, allowing easy identification of unsuccessful executions. When the user is satisfied with the data selection, they are prompted to provide a name for the dataset, which serves for later identification.

#### 2.3.2. Model training

To allow the user to train shadow models for the target skills given the training data, the user is presented with a list of available pretrained models (*base models*) which match the types of the target skills. For each target skill, they can choose to use a base model as-is, to finetune the base model on the previously configured dataset, or to train a new model from scratch. For each model to be trained, they are guided through a dialog to specify a name for the model for later identification, and hyperparameters for training, such as a learning rate, batch size, network size etc. Sensible defaults are provided for each setting. After training, loss curves and training metrics are displayed, as well as graphs comparing model predictions with ground-truth labels held out from the training data. In Guided mode, the UI for hyperparameter specification is simplified to hide all parameters but the learning rate and training duration. Additionally the loss curves are simplified significantly, giving a summarized overview over the training process. In both modes, training results are classified into “good performance”, “over-

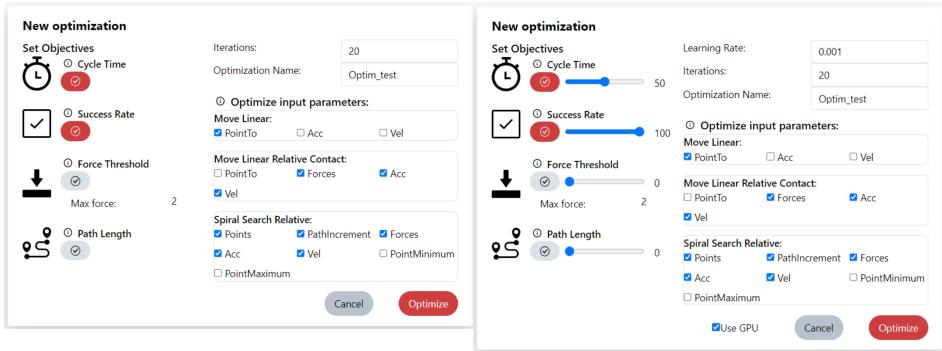


Fig. 3. UI for the optimization step, Guided (left) and Expert modes (right).

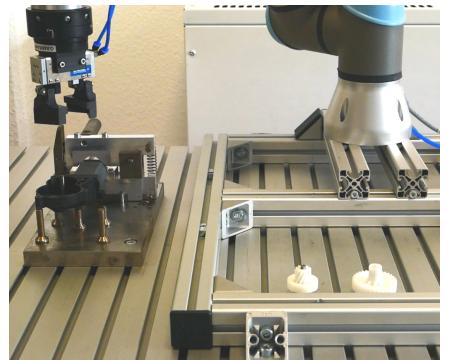


Fig. 4. Experiment set-up for robotic gearbox assembly.

fitting”, “underfitting”, “regularization” (dropout rate too high) and “erroneous training data”, along with explanatory descriptions. To explain the trained network’s behavior, LRP is computed to determine the relevance of each input parameter for the model’s predictions. This permits domain experts to check the model’s outputs for plausibility; predicted forces, for example, should typically increase with the velocity of the motion.

### 2.3.3. Program optimization

One typical user challenge when optimizing program parameters is the specification of the objective function. Process metrics such as task success, cycle time or path length are often contradictory, and simultaneous optimization for multiple metrics may result in parameterizations which are not intuitive to users, or the optimizer may fail to converge. In Expert mode, supported task objectives can be toggled, and their relative weights specified via sliders. In Guided mode, only toggling of objectives is allowed, and weights are assigned based on sensible defaults. Expert users have additional control over the hyperparameters of the optimizer, such as the step size and number of iterations (see Figure 3). The optimized parameterization is displayed alongside the initial parameterization, allowing a quick overview of how the program would change with optimization. To facilitate an intuitive understanding of optimization results and the behavior of the optimizer, we provide two explanation features. The model predictions for each optimization iteration are plotted together and the best iteration is highlighted, allowing the user to gauge the plausibility of the optimization results in terms of the robot’s motions. Additionally, the user can interactively change the optimized parameters and the resulting model predictions are plotted, providing an intuitive method for the user to sanity-check model predictions: If they increase the velocity of the robot, the predicted impact force should also increase and vice versa.

Model training and parameter optimization are typically performed more than once, as bad model performance such as insufficient generalization often becomes apparent once it is used for optimization. The UI permits users to navigate back and forth between training and optimization to repeat steps or fine-tune hyperparameters as needed.

## 3. Preliminary User Study

To quantify the influence of a targeted system design on the cooperation between human and machine for AI-based programming, a preliminary user study is conducted. Participants are tasked to solve a practical robot programming use case using the proposed system. The use case consists of optimizing a robot program for the assembly of the gearbox of a small electric motor, in which the robot grasps a gear, approaches the axle, performs force-controlled spiral search for the exact pose of the axle and inserts the gear onto the axle (see Figure 4). A preliminary study has been carried out and the results are discussed below (see Section 3.2). As the results are promising, a further investigation of the user interface is planned (see Section 4).

### 3.1. Study Design

The preliminary study comprised a total of 12 participants. While they were all familiar with robotics, 5 participants had a background in mechanical engineering, 6 in computer science, and 1 in product management. Based on their accounts of prior AI experience, we classified 4 of the participants as AI experts and 8 as AI novices. All participants were males over the age of 20 with at least one university degree. Each participant was presented a short task description. No additional help was provided. In a survey subsequent to the experiment, the topics (1) success, (2) usefulness, (3) usability, (4) user adaptation, (5) cognitive load and (6) transparency were evaluated on a unipolar Likert scale ranging from 1 (“not at all”) to 5 (“completely”). Questions (1), (5.1), (5.2), (5.3) and (5.4) are a subset of the NASA TLX questionnaire [17]. The questions on system explainability (6.1 - 6.9) are adapted from the “XAI question bank” [18].

### 3.2. Results

The results of the survey are shown in Figure 5. First, the overall perceived success of the users is evaluated (1). It shows a high variance, which might be due to different expectations of the system’s capabilities. Within the sub-tasks, perceived success is similarly high for both user groups with identical means

for both the context selection task (2.1) and the data generation and analysis task (2.2). This might be due to the simplicity of the tasks, where the extra expertise of AI experts is not required. Model training and evaluation (2.3) however requires a more comprehensive understanding of AI, which the AI experts leverage. This holds true for the final task of parameter optimization (2.4) as well. Participants understood each sub-task's objective and were able to generate a dataset, train a neural network and obtain predictions with that model. The provided textual explanations as well as default values were deemed useful by both participant groups (3).

The interface left the choice of the user mode (Guided or Expert) to the users. This decision is not adjusted to the specific task, which is unknown in advance. Thus users might find themselves facing unexpectedly difficult tasks or being limited by a simplified interface. During the experiment, it could be observed that only one participant changed the mode from Expert to Guided during the optimization task, motivated by the user's curiosity. Most users decided for one mode and never changed it (4.1). This indicates that users are able to self estimate their level of experience, and that the level of complexity was adequately adapted. (4.2) targets the user's perception of limited agency. The low scores ( $\mu_{\text{novice}} = \mu_{\text{expert}} = 1.75$ ) indicate that users of both groups seldom felt limited in their workflow. AI novices still had enough interaction possibilities while presented a reduced version of the user interface.

The pace of work (5.1) and negative emotions like stress or insecurity (5.2) were higher for AI novices which indicates a need for more guidance. Further, the experiment task was mentally demanding for most users of both user groups. In combination with the results on acquired guidance, it can be suggested that this might be due to the task being too complex.

To facilitate practical use of AI in real-world applications, AI-based decisions must be made transparent. The results showed that the source of the data was mostly understood by both user groups ( $\mu_{\text{novice}} = 4.125$ ,  $\mu_{\text{expert}} = 4.0$ ) (6.1). More complex tasks like the prediction of outputs was understood less well by AI novices (6.4). The implemented LRP plot was not fully understood (6.9). While the calculated feature importance score can serve as a starting point, explaining the neural network in depth remains a challenge.

### 3.3. Discussion

The preliminary study results indicate that the proposed system enables both AI novices and experts to use an AI-based robot program optimizer. The usefulness and intuitive usability of the system was ranked favorably, and users were able to perform a practical real-world program optimization task with acceptable cognitive load. The results showed high variance in perceived success, usefulness, and usability, which might be due to different expectations and understanding of the system's capabilities. The study emphasizes the need for additional guidance for AI novices, as well as more advanced techniques for explaining neural network behavior. The reliability of the results from the AI expert group is limited by the observed tendency of the expert participants to "play" with the system out

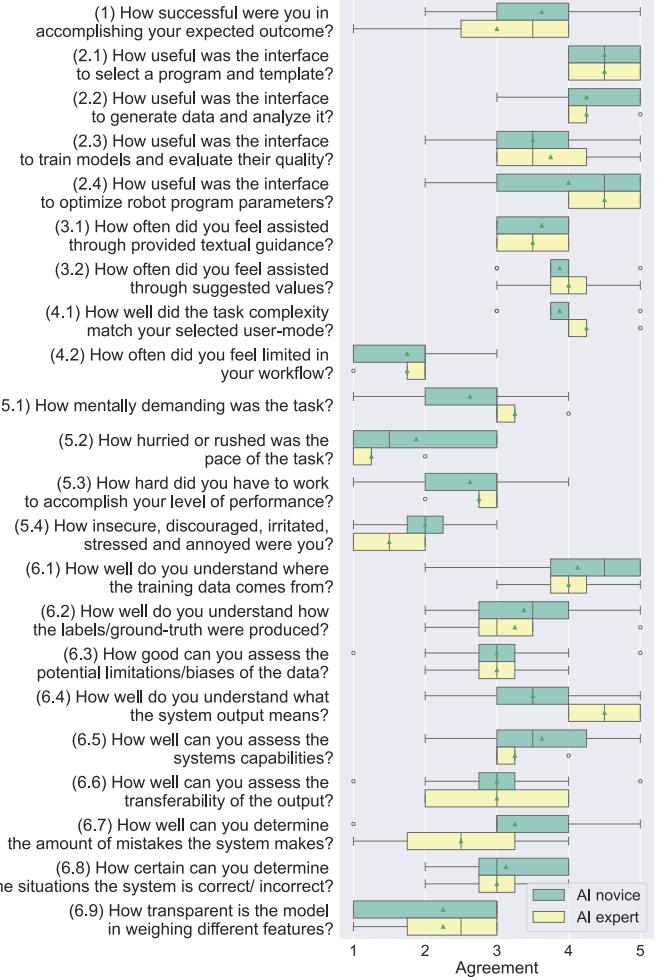


Fig. 5. Survey results of 12 participants, 8 of which are classified as AI novices and 4 as AI experts. ▲ indicates the median response.

of curiosity, rather than using the system with determination to achieve the task. Future studies will investigate methods to improve the comparability of results from different user groups.

## 4. Large-Scale User Study

In a large-scale study, the preliminary study will be systematized and the number of subjects increased to achieve statistical significance. The subjects will be students of engineering, computer science, and related disciplines, PhD candidates as well as engineers working in the field of automation or robotics.

### 4.1. Study Design

The two independent variables of the study are the level of explainability and the level of control of the system. For both, a distinction is made between low and high, resulting in a 2x2 matrix defining four variants of the user interface. We propose a double-blind between-subject design: To every participant, only one variant of the interface is presented, and neither the participants nor the experimenter know which of the four groups they are assigned to. Participants are asked to fill out a pre-task

questionnaire collecting demographic data (e.g. age, gender), self-reported level of experience with AI, industrial automation and robotics, attitude towards robots (GAToRs [19]), attitude towards AI (GAAIS [20]) and technology commitment [21].

For the explainability variable, several features have been defined and made available in the UI. These include statistics to explain the data, explanations of the model quality, and an evaluation of the relevance of individual elements. The level of control is expressed in the extent of possible hyperparameter tuning, both for model training and trajectory optimization.

During execution, the task completion time, success rate and UI interaction metrics such as the number of clicks are recorded. After the task, participants complete a questionnaire, including standardized measures for cognitive load (NASA TLX [17]), quality of explanations (Explanation Satisfaction Scale [22]) and trust in the system (Trust in Automation Questionnaire [23]). The exact design of the metrics is subject to further determination. A number of 120 subjects should be reached in order to achieve sufficient statistical power.

## 5. Conclusion

We present an XUI for an AI-based robot program optimizer, aiming to enable industry practitioners to use state-of-the-art AI methods for practical robot programming applications. The XUI is designed to address the skill gap between the AI competence required to use complex systems and the lack of experience of practitioners in industry, as well as to foster appropriate levels of trust in the correctness and performance of the AI system. The interface emphasizes user adaptability and explainability as guiding principles, and proposes explainability and intuitive user interaction mechanisms for each step in the workflow. We outline a preliminary user study on a realistic assembly use case and propose a study design for a planned large-scale user study. This study will focus on the level of explainability and control afforded by system, aiming to draw generalizable conclusions about the design of user interfaces for intuitive, trustworthy human-AI interaction in the manufacturing industry.

## Acknowledgements

This work was supported by the German Ministry of Education and Research grant 02L19C255, the DFG CRC EASE (CRC #1320) and the EU project euROBIN (grant 101070596).

## References

- [1] B. Alt, D. Katic, R. Jäkel, A. K. Bozcuoglu, M. Beetz, Robot Program Parameter Inference via Differentiable Shadow Program Inversion, in: 2021 IEEE Int. Conf. Robot. Autom. ICRA, 2021, pp. 4672–4678.
- [2] G. Evangelou, N. Dimitropoulos, G. Michalos, S. Makris, An Approach for Task and Action Planning in Human–Robot Collaborative Cells Using AI, Procedia CIRP 97 (2021) 476–481.
- [3] B. Alt, D. Katic, R. Jäkel, M. Beetz, Heuristic-Free Optimization of Force-Controlled Robot Search Strategies in Stochastic Environments, in: 2022 IEEE/RSJ Int. Conf. Intell. Robots Syst. IROS, 2022, pp. 8887–8893.
- [4] A. Muxfeldt, Automatic Error Recovery During Industrial Assembly Operations Based on Human Demonstrations (Dec. 2018).
- [5] R. S. Peres, X. Jia, J. Lee, K. Sun, A. W. Colombo, J. Barata, Industrial Artificial Intelligence in Industry 4.0 - Systematic Review, Challenges and Outlook, IEEE Access 8 (2020) 220121–220139.
- [6] F. Azmat, B. Ahmed, W. Colombo, R. Harrison, Closing the Skills Gap in the Era of Industrial Digitalisation, in: 2020 IEEE Conf. Ind. Cyberphysical Syst. ICPS, Vol. 1, 2020, pp. 365–370.
- [7] D. Gürdür Broo, O. Kaynak, S. M. Sait, Rethinking Engineering Education at the Age of Industry 5.0, Journal of Industrial Information Integration 25 (2022) 100311.
- [8] M. Chromik, A. Butz, Human-XAI Interaction: A Review and Design Principles for Explanation User Interfaces, in: C. Ardito, R. Lanzilotti, A. Malizia, H. Petrie, A. Piccinno, G. Desolda, K. Inkpen (Eds.), Hum.-Comput. Interact.–INTERACT 2021, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2021, pp. 619–640.
- [9] C. Haid, A. Lang, J. Fottner, Explaining Algorithmic Decisions: Design Guidelines for Explanations in User Interfaces, in: Hum. Factors Softw. Syst. Eng., Vol. 94, AHFE Open Acces, 2023.
- [10] J. Dodge, S. Penney, C. Hilderbrand, A. Anderson, M. Burnett, How the Experts Do It: Assessing and Explaining Agent Behaviors in Real-Time Strategy Games, in: Proc. 2018 CHI Conf. Hum. Factors Comput. Syst., CHI '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 1–12.
- [11] C. Bove, M.-J. Lesot, C. A. Tijus, M. Detyniecki, Investigating the Intelligibility of Plural Counterfactual Examples for Non-Expert Users: An Explanation User Interface Proposition and User Study, in: Proc. 28th Int. Conf. Intell. User Interfaces, IUI '23, Association for Computing Machinery, New York, NY, USA, 2023, pp. 188–203.
- [12] T.-C. T. Chen, Explainable Artificial Intelligence (XAI) in Manufacturing: Methodology, Tools, and Applications, Springer Nature, 2023.
- [13] R. Schmitt, I. Heine, R. Jiang, F. Giedziella, F. Basse, H. Voet, S. Lu, On the Future of Ramp-Up Management, CIRP Journal of Manufacturing Science and Technology 23 (2018) 217–225.
- [14] J. Ruiz, E. Serral, M. Snoeck, Unifying Functional User Interface Design Principles, Int. J. Human–Computer Interact. 37 (1) (2021) 47–67.
- [15] E. Schlungbaum, Individual User Interfaces and Model-Based User Interface Software Tools, in: Proc. 2nd Int. Conf. Intell. User Interfaces, IUI '97, Association for Computing Machinery, New York, NY, USA, 1997, pp. 229–232.
- [16] B. Leichtmann, C. Humer, A. Hinterreiter, M. Streit, M. Mara, Effects of Explainable Artificial Intelligence on Trust and Human Behavior in a High-Risk Decision Task, Computers in Human Behavior 139 (2023) 107539.
- [17] S. G. Hart, Nasa-Task Load Index (NASA-TLX); 20 Years Later, Proc. Hum. Factors Ergon. Soc. Annu. Meet. 50 (9) (2006) 904–908.
- [18] Q. V. Liao, D. Gruen, S. Miller, Questioning the AI: Informing Design Practices for Explainable AI User Experiences, in: Proc. 2020 CHI Conf. Hum. Factors Comput. Syst., CHI '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 1–15.
- [19] M. Koverola, A. Kunnari, J. Sundvall, M. Laakasuo, General Attitudes Towards Robots Scale (GAToRS): A New Instrument for Social Surveys, Int J of Soc Robotics 14 (7) (2022) 1559–1581.
- [20] A. Schepman, P. Rodway, The General Attitudes towards Artificial Intelligence Scale (GAAIS): Confirmatory Validation and Associations with Personality, Corporate Distrust, and General Trust, Int. J. Human–Computer Interact. 39 (13) (2023) 2724–2741.
- [21] F. J. Neyer, J. Felber, C. Gebhardt, Entwicklung und Validierung einer Kurzskala zur Erfassung von Technikbereitschaft, Diagnostica 58 (2) (2012) 87–99.
- [22] R. R. Hoffman, S. T. Mueller, G. Klein, J. Litman, Measures for Explainable AI: Explanation Goodness, User Satisfaction, Mental Models, Curiosity, Trust, and Human–AI Performance, Front. Comput. Sci. 5 (Feb. 2023).
- [23] M. Körber, Theoretical Considerations and Development of a Questionnaire to Measure Trust in Automation, in: S. Bagnara, R. Tartaglia, S. Albolino, T. Alexander, Y. Fujita (Eds.), Proc. 20th Congr. Int. Ergon. Assoc. IEA 2018, Springer International Publishing, Cham, 2019, pp. 13–30.