Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 00 (2019) 000–000

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing
August 9-14, 2019 | Chicago, Illinois (USA)

Intelligent authoring and management system for assembly instructions

Arno Claeys[a,b,]*, Steven Hoedt[a,b], Matthias Schamp[a,b], Lauren Van De Ginste[a,b], Georges Verpoorten[c], El-Houssaine Aghezzaf[a,b], Johannes Cottyn[a,b]

[a]*Department of Industrial System Engineering and Product Design, Ghent University, Technologiepark 903, 9052 Gent-Zwijnaarde, Belgium*
[b]*Industrial Systems Engineering (ISyE), Flanders Make, www.FlandersMake.be*
[c]*ProductionS Core Lab, Flanders Make, Oude Dietersebaan 133, 3920 Lommel, Belgium*

Abstract

Continuously increasing complexity and variance within high variety low volume assembly systems causes a vast amount of work instructions. As the amount of new models and variants increases, the need of efficient generation of unambiguous instructions rises. Continuous instruction modifications are unavoidable due to design, customer or process changes.
Case based research in cooperation with four manufacturing companies with manual assembly environments points out that assembly instructions authors currently are combining different authoring tools for creating and updating work instructions. Consequently, keeping the rising amount of work instructions up to date becomes less trivial. Furthermore, authors often create work instructions from scratch while instructions of product variants are mostly identical. This causes a large amount of similar work instructions stored as separate documents. As a result, the amount of inconsistent and outdated assembly instructions increases. Poor assembly instruction quality causes frustration and a lower performance of assembly operators.
An automatic authoring system and intelligent operator feedback must eliminate these problems. The automatic authoring system provides the author with an overview of preprocessed information and related historical assembly instructions that can serve as a basis for the newly created instructions. In this way, the creation of instructions can be significantly accelerated and work instructions will become more consistent. An experimental lab setup is built in order to test the presented framework. Based on the first tests, the authoring process was significantly accelerated. Further tests within production environments are required in order to validate the presented framework.

* Corresponding author. Tel.: +32 056 24 12 21
E-mail address: arnclaey.claeys@ugent.be

2351-9789
Selection and peer review under the responsibility of ICPR25 International Scientific & Advisory and Organizing committee members

Keywords: work instruction management; high variance low volume assembly; authoring effect suggestions

1. Introduction

Manually performed high variety low volume (HVLV) assembly operations become more complex due to the diversity of product variants [1]. In order to achieve the high quality standards, supporting the human operator with qualitative instructions is indispensable. However, HVLV warrant an inordinate amount of up-front methods engineering effort to create and maintain those instructions. As the amount of new models and variants increases, the need of efficient generation of unambiguous instructions rises. Nevertheless, research has shown that instruction quality problems (e.g. incorrect information) occur frequently [2]. Resolving these problems is often a laborious task while instruction authors are overwhelmed by the increasing demand for creating and updating assembly instructions.

Case based research in cooperation with four special machinery manufacturing companies points out that assembly instruction authors currently are combining various authoring tools for creating and updating assembly instructions. Consequently, manual authoring and keeping the rising amount of instructions up to date becomes less trivial. Also, there is a limited data stream between instruction authors and shop floor operators. Therefore, a large amount of valuable tacit operator knowledge remains unexploited. Based on tacit knowledge, best practices and potential assembly instructions for infrequent operations can be derived. Although several methods for tacit knowledge extraction have been developed [3], the classification and formalization of knowledge within HVLV environments requires a lot of effort. In current practice, operator feedback is only handled indirectly, via a long paper trail, which is slow and ineffective. Furthermore, authors often create instructions from scratch while instructions of product variants are mostly identical. This causes a large amount of similar assembly instructions stored as separate documents. As result, the amount of inconsistent and outdated assembly instruction increases.

It can be stated that there is a lack of tooling for the methods engineer to efficiently create and update instructions based on operator feedback. In order to define a methodology for an intelligent authoring system, current instruction quality deficiencies within HVLV assembly systems must be determined. Therefore, a case based study is set up and discussed in section 2. Based on the results of the case study, a framework for an intelligent authoring and management system for assembly instructions is presented in section 3. The framework describes a method for integrating operator feedback and for providing instruction authors an overview of preprocessed knowledge and historical assembly information. Section 4 describes the experimental approach for validating the framework. Finally, the conclusions of this work and further research are discussed in section 5.

2. Assembly instructions quality

It is shown that a high amount of quality deviations are originated from manual assembly [4]. Therefore, assembly instructions must have a satisfactory level of information quality. The quality of instructions has a significant impact on the operator's efficiency, learning rate [5], error rate and cognitive load [6,7]. Poor quality of instructions causes frustration and a lower performance of assembly operators. Haug [8] has developed a framework for evaluating informational quality for work instructions. The framework consists of 15 instructional information quality dimensions divided in five categories.

The first category, intrinsic problems, are closely related to the perception of the individual. These quality dimensions are derived from the relationship between needed and given instructions. The remaining categories are extrinsic problems and depend mostly on the individual's context. Therefore, extrinsic problems are harder to evaluate. Representational problems are related to the ease of understanding and interpretability of instructions. Poor represented information may tangle and delay the individual that receives the instruction. Unmatched information is information that is inappropriate according to the individual's situation. Both representational problems and unmatched information have a direct impact on the cognitive load of the individual. Questionable information dimensions are relevant in relation to how an individual follows the given instructions. If information is formulated in a questionable way that makes it implausible, the individual would be inclined to not believe that information. Lastly, inaccessible information refers to problems in relation to locating or accessing relevant instructions.

In the current situation, only a small part of the instruction quality problems are addressed as the set of assembly instructions at hand are typically believed to be correct and complete. Based on a number of use cases, the most relevant instruction quality dimensions for HVLV manufacturers and their impact on assembly performance are studied.

The first company is situated in the aviation sector. The use case is located in the flap beam mechanism assembly line. Assembly instructions are authored in MS Word and managed through a product lifecycle management (PLM) system. The instructions are available to the operator in an uncontrolled PDF hardcopy at the workstation or digitally through the PLM system. Each procedure contains a vast amount of instructions with images from the computer aided design (CAD) environment, textual instructions and pictures. Operator feedback towards the method engineer is either given indirectly through remarks in a logbook system, or orally.

The second company is part of the electronics industry. The use case is located at the assembly line of a new type of laser projector. Assembly instructions are authored based on CAD files. These files are used to display the different assembly steps in 3D, which are transformed to 2D-pages with additional information (e.g. bill of material (BOM)). The assembly procedure is exported as a PDF file and is digitally available at the workstation.

The third company is a manufacturer of agricultural machinery. The assembly instructions are concise and predominantly text based. When performing an assembly task, the operator has to rely on his knowledge. Only for complex procedures with a high error rate, standard operating procedures (SOPs) are available. The SOPs are authored in MS Excel and are distributed on paper to the shop floor.

The fourth company is specialized in machines for the textile industry. The use case is located in the niche zone where highly customized and unique machines are assembled. The assembly instructions are created and managed in a digital instruction platform. Operators receive digital step-by-step instructions on a wearable information device (tablet, smartphone, etc.). The operators give feedback with the mail application installed on the information device.

In order to determine the relevance of each dimension within HVLV assembly environments, interview sessions with assembly instruction authors (i.e. method engineer, quality engineer) and users (i.e. operator, team coordinator) are organized. The ratings are determined based on the problem frequency and the problem impact (e.g. assembly errors, cycle time, learning rate, etc.). The relevance scores for each dimension, on a scale from 0 to 5, are plotted in Table 1. It is clear that some information quality dimensions are less relevant in the context of manual assembly instructions. Unneeded, too repetitive and too large amount problems is reported by the companies to have a negligible impact on assembly performance. Furthermore, by applying context-aware content filtering [9], assembly instructions can be matched to the operator's context. Within the current situation of the use cases, inconsistent and inconcise problems are rated as irrelevant. Nevertheless, the heterogeneous author base and the increasing amount of assembly instructions may cause inconsistencies. Consequently, managing these instructions becomes more complicated. Within the use cases, operators are responsible for following the assembly instructions as they are formulated. If an instruction is found to be incorrect, this must be reported. Therefore, questionable information dimensions are rated as less applicable. Inaccessible information dimensions for digital assembly instructions are more related to connection and hardware issues. Thus, these problem dimensions are out of the scope of our work.

The highly relevant instructional quality dimensions have mutual characteristic similarities. Therefore, these dimensions can be categorized into assembly instruction quality categories. Deficient and ambiguous information both can have a noticeable impact on the operator's cycle time and learning rate. If the operator does not has the right knowledge, these problems may cause assembly errors. This type of problems can be solved by creating new instructions or by adding additional information. Incorrect and untimely information can have a direct impact on the assembly error rate. In order to solve these problems, assembly instructions need to be corrected by replacing the information. Assembly instructions that are too complex or difficult to understand, mainly cause confusion among novice operators. In this scenario, the operator usually calls for assistance, which results in delays and thus longer cycle times.

Table 1. Instruction information quality problems relevance

Category	Quality dimension	Case 1	Case 2	Case 3	Case 4	Average
Intrinsic problems	Deficient	5	5	5	5	5,0
	Ambiguous	3	4	4	5	4,0
	Unneeded	1	0	1	0	0,5
	Too repetitive	1	0	0	0	0,3
	Incorrect	5	5	5	5	5,0
Representational problems	Inconsistent	0	1	1	2	1,0
	Inconcise	1	2	1	2	1,5
	Difficult to understand	2	3	4	4	3,3
Unmatched information	Too complex	5	4	5	4	4,5
	Too large amount	1	2	3	2	2,0
	Untimely	2	4	3	5	3,5
Questionable information	Poor believability	0	1	2	4	1,8
	Poor reputation	0	1	2	0	0,8
Inaccessible information	Security barriers	0	0	0	0	0,0
	Other accessibility barriers	1	2	2	5	2,5

Based on the relevant instructional quality dimension characteristics within the context of assembly instructions, three types of assembly instruction quality problems are defined: incomplete, incorrect and complex assembly instructions. Table 2 summarizes the assembly instruction quality problem categories with the related instructional quality dimensions, the impact on assembly performance and the approach to eliminate the problem.

Table 2. Assembly instruction quality problems

	Incomplete	Incorrect	Complex
Instructional information quality dimensions	Deficient Ambiguous	Incorrect Untimely	Difficult to understand Too complex
Instructional information quality dimensions	Cycle-time (+) Error rate (++) Learning rate (--)	Error rate (+++)	Cycle-time (+) Error rate (+) Learning rate (-)
Approach to eliminate problem	Add information	Update information	Simplify and contextualize information

3. Framework for intelligent authoring and management of assembly instructions

The operator on the shop floor has the most expertise on the assembly activities and can indicate which instructions are clear and informative and which instructions need improvements. Extending the role of the operator to a knowledge worker will contribute to continuous instruction optimization and will minimize the assembly instruction quality problems, defined in the previous section. In order to allow an efficient feedback loop between the operator and the instruction author, a framework for an intelligent authoring and management system for assembly instructions is constructed.

Fig. 1 illustrates the middleware framework for automatic authoring effect suggestions. The middleware layer exists of three ontological data models. The assembly information model is an ISA-95 based domain ontology built upon a generic model for assembly instructions [10]. This model integrates all relevant manufacturing and assembly data from several data streams (e.g. enterprise resource system (ERP), manufacturing execution system (MES), PLM, CAD, etc.) through an assembly information configurator. The annotations ontology defines and maps the explicit feedback, given by the operators on the shop floor, to the assembly information ontology. The generic manual assembly context ontology [11] contextualizes both assembly information and annotations.

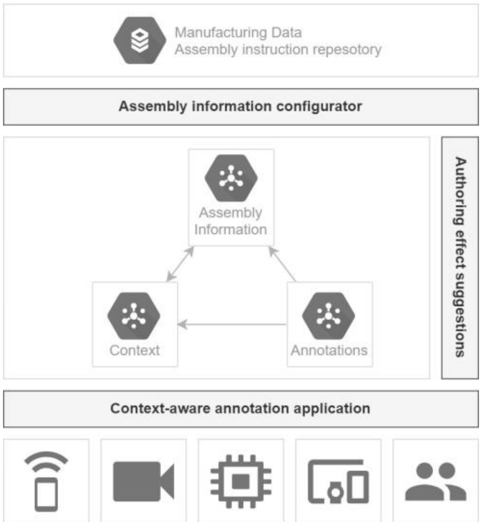

Fig. 1. Framework for intelligent authoring and management of assembly instructions

3.1. Context-aware annotations

In order to preprocess operator feedback, a method for contextualizing the operator's annotations is required. Through the usage of intelligent IoT sensors (e.g. smart equipment, real time location systems (RTLS), etc.), shop-floor context can be derived [12]. The W3C web annotation data model [13] is used as starting point for the design of the assembly instruction annotation model. The model provides an extensible, interoperable framework for expressing annotations such that they can easily be shared between platforms. Therefore, this model is of interest to integrate annotations within multiplatform manufacturing software solutions. An annotation is considered to be a set of connected resources, typically including a body and target, and conveys that the body is related to the target. The nature of this relationship changes according to the intention of the annotation. In most cases the body is content linked to the target (e.g. comment on an instruction). Annotation entities can be contextualized by linking them to manual assembly context entities, derived from shop-floor sensor data. By making context-aware annotations, less input is required from the operator. It is of importance to understand the reason why the annotation was created. Based on the assembly instruction quality problems, three motivations for annotation creation are distinguished: add new or missing information, update poor quality instructions with feedback and indicate unclear or irrelevant information.

3.1.1. Add new or missing information

When assembling a new product variant typically only a limited set of instructions is available. In this case, an operator must rely on raw information (i.e. drawings, short descriptions, bill of material, etc.) and his own knowledge. Therefore, these types of assembly operations are only eligible for experienced operators. Most relevant and complete information can be recorded by the operator on the shop floor. Facilitating operators to capture information (e.g. textual description, picture, video, speech, 3D view, etc.) during their task at hand results in documented assembly process knowledge. This information is helpful to find best practices and becomes available to create assembly instructions. In order to obstruct the operator as less as possible during his assembly task, the method for capturing information must be feasible and efficient. When the operator captures information, an annotation with the information content as body is created. Assembly context (i.e. operator ID, used equipment and material, location, posture, skill level, etc.) stored into the context model, is used to tag and thus contextualize the annotation. By analyzing the context tagged to the annotation, the annotation can be classified to the corresponding assembly activities. Thus the annotation target may be identified after context analysis.

3.1.2. Update poor quality assembly instructions with feedback

Currently, incorrect and deprecated assembly instructions are often only noticed after customer complaints. Detecting incorrect or outdated information during an early stage will reduce the amount of assembly errors significantly. Operator knowledge and expertise can be used to detect and update incorrect or inappropriate assembly instructions. By facilitating the process of giving feedback, experienced operators can give suggestions for corrections when incorrect information is noticed through an annotation. The annotation target is the concerning instruction, or part of the instruction (i.e. selection of text). The suggested correction is stored into the annotation body. By analyzing the context linked to the annotation, similar instruction problems can be detected and corrected. The feedback loop between the operator and the methods engineer must prevent future incorrect assembly instructions.

3.1.3. Indicate unclear or irrelevant information

Unexperienced operators often do not have the knowledge to correctly interpret complex assembly instructions. This causes confusion and a higher cognitive load. Complex and unclear procedures need to be split into smaller and easily understandable parts, used to guide the operator step-by-step through the assembly [14]. Nevertheless, one operator may need more information while another may be bored with unnecessary statements [15]. Also, the information format may differ as result of the operator's personal preferences. The introduction of a context-aware recommender system can solve this problem by offering personalized and context sensitive information towards the operator [16]. Still, operator feedback is required to refine the content filtering rules. By enabling the operator to rate information, existing filtering rules can be updated or new rules can be generated. Examples of ratings are instruction clearness and instruction relevance. By contextualizing the rating annotation, the operator's situation when giving the rating can be derived. By applying recommender system principles [9], predictions of operator ratings for that situation can be made and thus, filtering rules can be refined.

3.2. Information retrieval based on task similarity

In order to minimize the effort to process the explicit operator feedback, annotations must be classified and linked to already available assembly information. Furthermore, already available information could be suggested based on task similarity [17,18]. In order to make full advantage of available assembly instructions, a structured method of information comparison is required. By analyzing the data through different approaches, additional knowledge and meaning can be extracted.

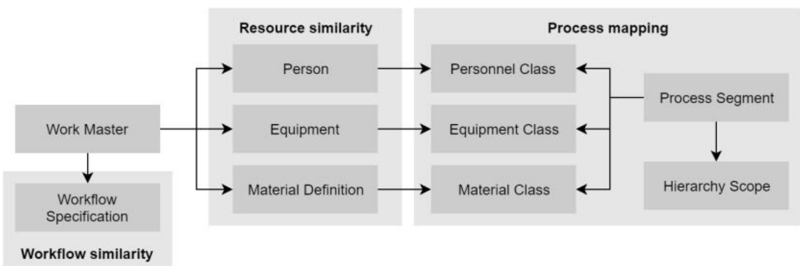

Fig. 2. Similarity approaches on the assembly information model

By using the structure of the ISA-95 based generic assembly information model, data coming from several sources can be analyzed and compared. Fig. 2 gives an overview of the similarity approaches performed on the assembly information model. A work master defines the resources and instructions required to perform a unit of work [19]. There are three types of resources: personnel, equipment and material. Resource classes define and group resources with similar characteristics. A process segment represents a manufacturing activity and defines the resource classes needed for execution [20]. The location where an activity fits is described through a hierarchy scope, referring to the equipment hierarchy model [21]. A workflow specification is a collection of nodes and connections that contains all information (i.e. assembly instructions) required to execute the related work master.

Through process mapping, work masters are classified with the use of process segments. Therefore, several process attributes, such as resource classes and hierarchy scope, are taken into consideration. If the resource classes derived from the work master requirements completely map to a process segment, then the work master can be linked to that process segment. A work master can be categorized by multiple processes. The assembly information ontology, built upon the generic model for assembly instructions, further formalizes assembly information through taxonomies. If a work master does not map to an existing process segment, a new process segment can be determined through the hierarchical links by using ontology distance and link prediction algorithms. A process describes the type of activity and is an important factor to determine the most appropriate information format and level for detail. For example, when assembling a certain unique component, it can be useful to show a detailed CAD drawing. In the case of a very common and simple activity, for example screwing a bolt, a description could be more relevant information than a detailed drawing.

A lot of information may be shared by a number of similar work masters. Renu et al. [17] compare several text similarity algorithms for computing the similarity of textual assembly instructions. In the case of authoring new assembly instructions, only a limited set of instructions is available. Liu et al. [18] use the technique of context matching for obtaining workflow similarity. Nevertheless, the comparison of context is text based. In order to calculate the similarity score between work masters, the data structure of the standardized ISA-95 model is used. Resource similarity compare both resources and resource classes in order to determine the similarity between two work masters. By comparing the ISA-95 attributes using set similarity algorithms (i.e. Jaccard, Pearson, Cosine, etc.) a relevance score can be calculated.

4. Experimental tests

In order to test the presented framework, experimental tests within a realistic assembly environment are required. Therefore an assembly product which requires several assembly operations (i.e. mechanical assembly, electrical assembly and pneumatic assembly) is introduced. The product is designed to be modular, thus several variants can be constructed. For one variant, all detailed assembly instructions were created from scratch, and integrated into the assembly information ontology. In order to test the similarity approaches, an application is built based on the framework presented in this work. An intelligent operator tracking system, integrated into the experimental setup, is used to capture context during the assembly process. When an annotation is created, the recorded context is mapped through the timestamp. Based on the linked context, annotations are classified into process segments and linked to the most appropriate work masters. Based on the annotations and already available information, assembly instructions for the remaining product variants where created. Based on the first tests, the authoring process was significantly accelerated compared to authoring from scratch.

In order to validate the framework, a larger set of assembly information and annotations is required. Therefore, further experimental tests and tests on production environments with experienced operators are required.

5. Conclusions and further research

Based on a case study, instructional information quality problems within HVLV assembly systems are studied. As result, three types of assembly instruction quality problems are defined: incomplete, incorrect and complex assembly instructions. This work presents a framework for intelligent authoring and management of assembly instructions. The framework describes a method for supporting the instruction author in obtaining qualitative assembly instructions.

The framework must facilitate explicit operator feedback through annotations. By enriching the annotation with context, less operator input is required and automatic task classification can be performed. The framework describes different approaches to retrieve relevant information based on task similarity. Based on process mapping, template information is suggested based on the process type. Similarity between assembly tasks are calculated through resource and workflow similarity. This allows efficient information reuse for similar assembly tasks.

An experimental lab setup is built in order to test the presented framework. Based on the first tests, the authoring process was significantly accelerated. During further research, more experiments will be performed on the lab setup. After refining the framework based on the experiment results, further tests with experienced operators in production environments will be performed in order to validate the framework.

Acknowledgement

This research is supported by Flanders Make, the strategic research center for the manufacturing industry in Flanders.

References

[1] Claeys A, Hoedt S, Soete N, Van Landeghem H, Cottyn J. Framework for evaluating cognitive support in mixed model assembly systems. IFAC-PapersOnLine 2015;28:924–9.
[2] Eriksson G, Malmsköld L, Moestam L, Johansson PE., Johansson P, Fast-Berglund Å. Assessment Based Information Needs in Manual Assembly. DEStech Trans Eng Technol Res 2018.
[3] Liu L, Jiang Z, Song B, Zhu H, Li X. A Novel Method for Acquiring Engineering-Oriented Operational Empirical Knowledge. Math Probl Eng 2016;2016.
[4] Johansson PEC, Enofe MO, Schwarzkopf M, Malmsköld L, Fast-Berglund Å, Moestam L. Data and Information Handling in Assembly Information Systems – A Current State Analysis. Procedia Manuf 2017;11:2099–106.
[5] Watson G, Butterfield J, Curran R, Craig C. Do dynamic work instructions provide an advantage over static instructions in a small scale assembly task? Learn Instr 2010;20:84–93.
[6] Broberg J, Skoogh A, Fast-Berglund Å, Li D, Mattsson S, Salunkhe O. Effects of Information Content in Work Instructions for Operator Performance. Procedia Manuf 2018;25:628–35.
[7] Mattsson S, Fast-Berglund, Li D. Evaluation of Guidelines for Assembly Instructions. IFAC-PapersOnLine 2016;49:209–14.
[8] Haug A. Work instruction quality in industrial management. Int J Ind Ergon 2015;50:170–7.
[9] Ricci F, Rokach L, Shapira B. Introduction to recommender systems handbook. 2011.
[10] Claeys A, Hoedt S, Van Landeghem H, Cottyn J. Generic Model for Managing Context-Aware Assembly Instructions. IFAC-PapersOnLine 2016;49:1181–6.
[11] Claeys A, Hoedt S, Schamp M, Landeghem H Van, Cottyn J. Ontological Model for Managing Context-aware Assembly Instructions. IFAC-PapersOnLine 2018;51:176–81.
[12] Perera C, Zaslavsky A, Christen P, Georgakopoulos D. Context Aware Computing for The Internet of Things: A Survey 2013;X:1–41.
[13] W3C. Web Annotation Data Model 2017. https://www.w3.org/TR/annotation-model/.
[14] Aehnelt M, Bader S. Information Assistance for Smart Assembly Stations 2015:143–50.
[15] Aykin NM, Aykin T. Individual differences in human-computer interaction. Comput Ind Eng 1991;20:373–9.
[16] Loeb S, Panagos E. Information filtering and personalization: Context, serendipity and group profile effects. 2011 IEEE Consum Commun Netw Conf CCNC'2011 2011:393–8.
[17] Renu RS, Mocko G. Computing similarity of text-based assembly processes for knowledge retrieval and reuse. J Manuf Syst 2016;39:101–10.
[18] Liu T, Wang H, He Y. Intelligent knowledge recommending approach for new product development based on workflow context matching. Concurr Eng Res Appl 2016;24:318–29.
[19] ISA. ANSI/ISA-95.00.04-2012 Enterprise-Control System Integration - Part 4: Objects and attributes for manufacturing operations management integration. 2010.
[20] ISA. ANSI/ISA-95.00.02-2010 (IEC 62264-2 Mod) Enterprise-Control System Integration - Part 2: Object Model Attributes. 2010.
[21] ISA. Ansi/isa-95.00.01-2010 (iec 62264-1 mod) - enterprise-control system integration - part 1: Models and terminology. 2010.