Intuitive Task Programming of Stud Welding Robots for Ship Construction

Rasmus S. Andersen[1], Simon Bøgh[1], Thomas B. Moeslund[2] and Ole Madsen[1]

***Abstract*— Ship construction is a major industry worldwide, and many tasks have been automated. One task that is still solely carried out manually is welding of studs. This paper presents a semi-autonomous approach to robotic stud the welding with focus on the HRI (Human-Robot Interaction). The welding itself is carried out autonomously by an autonomous industrial mobile manipulator (AIMM). An intuitive interface is proposed for the AIMM to ensure safe and correct operation. The interface allows non-expert operators to program, verify, and reprogram the robot's task on the manufacturing site. Task specific information is projected directly into object space as augmented reality using a projector mounted on the robot end-effector. Specifically, stud positions are shown on the ship wall before welding is initiated, and positions can be added, deleted, and moved using an IMU as pointing device.**

The contribution of this paper is an intuitive interface for on-site programming of stud welding robots; implemented in a skill-based task programming architecture. The system is designed and implemented, and proof-of-concept tests are presented.

I. INTRODUCTION

Ship construction is a major industry worldwide. In Europe alone, ship construction and related activities have an annual turnover of around €30bn and employ more than 500,000 people [1]. The industry is very competitive, and this has sparked much interest in increasing the productivity of ship building. One of the methods to increase productivity is to increase the degree of automation. Therefore, a significant amount of research has been directed towards automating various subtasks in ship construction, including hull blasting [2], [3] and welding [4], [5], [6]. In this paper, we look at automating stud welding, which has previously received less attention. The focus is particularly on developing intuitive human-robot interfaces that can enable non-expert workers to program and verify a stud welding task as fast and safely as possible.

On modern container ships, several millions studs are welded on the inside of the ship hull and compartments to hold insulation, cables and other equipment. An example of this is shown in Figure 1, where studs have been welded in place on two vertical wall segments and subsequently been bend.

Fig. 1. Two wall segments with studs welded in place and bend.

There can be several reasons why the automation of stud welding has not previously been attempted. Firstly, most studs have to be placed inside compartments, and a compartment is one of the less hazardous working areas in a ship under construction. It has been a higher priority to automate tasks in areas such as inside double hulls. Secondly, humans will typically also have to work in the compartments in order to perform other tasks, and this enforces additional safety requirements on the robot. However, automating stud welding tasks do have a large potential for reducing costs of ship construction; simply because of the large number of studs required.

When a robot is introduced for a new task in a shipyard, it is necessary for the shipyard workers on the site to be able to control and verify the task that it carries out. The HRI system proposed in this paper is designed to do exactly this: Enable non-experts to instruct an Autonomous Industrial Mobile Manipulator (AIMM) to carry out a welding task safely and sufficiently precise in accordance with the relevant standards. The instruction interface is shown in Figure 2.

Task relevant information is projected onto the ship wall; including the welding position for each stud. This is denoted the *object space*, because this is where the robot is to manipulate objects. The operator can add, remove, or move stud locations using an IMU device (in our experiments a Wii remote).

The paper is organized as follows: In Section II, related work is presented. Specifically, Section II-A deals with suitable architectures, while Sections II-B to II-D deals with related human-robot interfaces. In Section III, the methods applied in the system and interface is described, and our test setup and preliminary experiments are presented in Section IV. Finally conclusions and discussions are drawn in Section V.

II. RELATED WORK

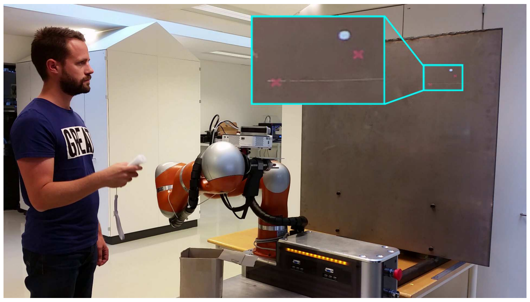
A. Robot Programming Architectures

A user interface intended for non-experts needs to provide information on as high a level as possible, while hiding com-

---

[1] Department of Mechanical and Manufacturing Engineering, Aalborg University, Denmark {rsa,sb,om}@m-tech.aau.dk

[2] Department of Architecture, Design and Media Technology, Aalborg University, Denmark tbm@create.aau.dk



(a) The operator adjusts stud positions.

(b) Autonomous stud welding.

Fig. 2. Teaching of stud positions in object space using a projector mounted on the robot end-effector using a Wii remote, and subsequent autonomous stud welding.

plicated configurations that require specialized knowledge. This is typically done by presenting functionality provided by a robot as *function blocks*, or *robot skills*. One of the first to draw attention to this idea was the STRIPS planner [7]. Their focus is to make it possible to perform automatic planning of how to get from an initial state to a goal state by combining a set of simple actions or skills. In [8], the idea of skills is further generalized in a framework known as SKORP. Here, the focus is both on what a skill should consist of, and also on developing a (at the time) modern and user friendly interface to enable workers to program robots faster.

A three-layered architecture is proposed in [9]. It is argued that a traditional sense-plan-act (SPA) architecture is insufficient for robots solving problems in a complex scenario. In the lowest layer, real-time control is placed. In the highest layer, time independent processes are run that are related to the overall task of the robot. The middle layer, the *Sequencer*, continuously changes the behavior of the lowest level, in order to complete a goal. The functionality in this level can be interpreted as a type of skills, which encapsulates the functionality of the lower levels, and works towards a goal by sequencing hardware-near actions.

More recently, in [10] the three layered structure is augmented by yet another layer on top, which contains a high level scenario description presented in a user friendly interface. This is aimed at solving manipulation tasks for service robots. The focus here is also on the architecture, where skills constitute the third layer; just above the hardware control layer. Another approach to skills is taken in the *Knowledge Integration Framework* [11]. Here skills are seen as a means to reuse functionality across multiple platforms.

The purpose of a skill in the current context is to present robot functionality as generic, understandable function blocks, which can easily be parameterized by non-experts. The skills must be very easy and fast to reconfigure, and the focus is therefore necessarily especially on the human-robot interaction, as well as on safe and reliable execution. The skill framework applied here is has previously been presented on a general level in [12].

B. *GUI Based HRIs*

Traditionally, HRI has started as human-computer interaction systems specifically designed for robots through GUIs. For mobile robots, a movable screen such as a PDA or a tablet is a straight forward choice. Examples of this is Lundberg et. al.'s field robot interface, where a simple interface including various logistic functions are included [13], or Perzanowski's multi-modal PDA interface supporting easy task instruction[14]. In [15], Muszynski et. al. presents a task level programming interface for tablets able to handle different control levels and more generic tasks.

For a welding robot working on a construction site, however, it will not be practical for workers to handle a tablet, each time they need to interact with the robot. Also, a tablet drives the operators attention away from the object space by nature, and thus decreases the intuitiveness of the task.

C. *Vision Based Gesture Recognition*

An alternative to provide input on a screen is to recognize natural human gestures. Many different approaches have been taken to this, including magnetic field trackers, data gloves, body suits, an cameras [16]. Vision based gesture estimation has the advantage that the user is not required to wear or handle external devices. This has been extensively researched for robots; especially since the launch of the MS Kinect in 2010. Pedersen et. al. developed a Kinect-based interface for instructing an AIMM to follow an operator and to pick up and place boxes [17]. A similar system was developed by Quintero et. al., where any surface can be selected [18]. The result is shown to the operator as augmented reality on a screen. The idea of using augmented reality for feedback is further extended by Alvarez-Santos et. al. in a interface for a tour guide robot [19]. They introduces augmented reality-buttons that the operator can interact with, thereby giving more complex instructions.

A limitation for vision based gesture recognition interfaces is that the operator always have to be located within the view of the detection device (e.g. Kinect). Also, gesture interfaces tend to be tedious to use for tasks that are not very simple.

D. *Smart Device Interfaces*

Smart devices have the potential to combine the precise and comprehensive information provided by a GUI with the direct interaction that can be achieved using gesture recognition. Some of the earliest interfaces to take advantage of smart devices were proposed by Microsoft in 2003 with the *XWand* [20] and *WorldCursor* [21]. The XWand is a pointing device based on IMU sensors, and the WorldCursor is a ceiling-mounted projector which can project a cursor anywhere (with line-of-sight) inside a room. In 2009, the XWand/WorldCursor idea was used Ishii et. al. to specifically provide simple instructions to an autonomously moving robot by [22]. Kemp et. al. developed a similar system in 2008, but replaced the projector with a laser pointer, thereby eliminating the line-of-sight restriction [23]. The laser dot is detected with an omni-directional camera on the robot, and a pan-tilt stereo camera on the robot is rotated to accurately determine the location of the laser dot.

The interface presented in this work attempts to take advantage of the rich information that can be shown by a projector, while enabling the operator to focus his attention on the working area.

III. METHODS

A. *The CARLoS Scenario*

CARLoS is an EU project aimed at increasing the competitiveness of European shipyards by automating stud welding tasks which are currently being carried out manually. In the CARLoS project, a robot is being developed that normal shipyard workers must be able to operate without any or with only minor experience in robot control. This includes setting up automatic stud-welding tasks, which the current paper focuses on. The different methods applied are described in the following subsections.

B. *Skill Based Architecture*

We take advantage of a skill-based architecture to provide intuitive and safe programming. In this architecture, each skill is interpreted as an object-centered ability which can easily be parameterized by a non-expert. A skill consists of a teaching phase and an execution phase, both of which transform an initial state to a goal state. This is illustrated in Figure 3 (slightly modified from [12]).

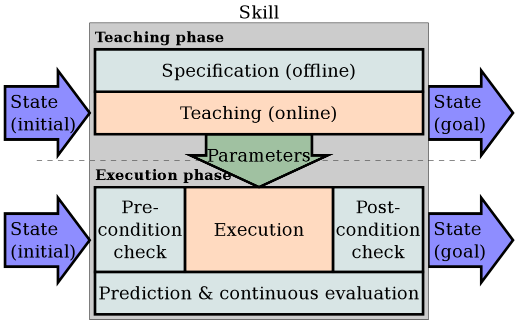


Fig. 3. Skill model that supports intuitive human-robot interaction. A skill consists of a teaching phase (top) and an execution phase (bottom), which both transforms an initial to a goal state. In the teaching phase, all necessary parameters are specified either online (on the production site) offline (on-site or in an office).

The basic abilities of the robot are contained in the execution block in the figure. The remaining blocks extends this functionality into a skill. The teaching phase makes the skill re-programmable. In this phase, the operator specifies all the parameters that transform the skill from a generic template into finished function which performs a useful operation. Teaching is divided into an offline specification part and an online teaching part. The online teaching must be done on the production site, and the parameters specified here are typically directly related to the movement of the robot. The offline specification, on the other hand, deals with parameters that can be chosen beforehand. Offline specification can therefore be carried out either on site or at another location.

The execution phase consists of the execution block combined with pre- and postcondition checks as well as prediction and continuous evaluation. The precondition check determines whether the world state lives up to the requirements of the skill. If this check is passed, execution can begin. During ongoing execution, continuous evaluation ensures that the skill is executed as expected. When the execution has finished, a postcondition check determines whether the current world state is as predicted. Together, the pre- and postcondition checks of a skill ensure that the world state is changed in a known, safe, and controlled manner and thus allows the skill to be combined with other skills in order to solve larger tasks.

To sum up, the purpose of a skills-base architecture is to increase the programming speed and to ease the way humans interact with industrial robots. In this paper, a welding skill is developed within the skill-based architecture. The paper focuses on an intuitive interface for online teaching which supports a non-expert in programming the robot safely and efficiently on the production site.

C. *The Welding Skill*

The autonomous welding functionality is implemented as a robotic skill as presented in Figure 3. The purpose of this specific skill is both to make it easy and safe to use robotic welding, and to allow the operator to easily combine welding with other independently developed skills. The skill realization is illustrated in Figure 4. The parameters necessary for execution are displayed in the middle, the teaching phase is shown the the left, and the execution phase is shown to the right. The first three parameters can be efficiently and safely specified offline in a graphical interface. The last parameter, the actual welding positions, might on the other hand not be safely and correctly specified offline. This is due to the fact that the environment inside the hull of a ship under construction is only semi-structured and is often not known precisely in advance. Therefore, the operator must have the possibility to adjust stud positions manually on-site.

Pre- and postcondition checks are included in Figure 4 as well. In these, status of all components of the system is checked to ensure predictable operation. In case of unexpected behavior, all operation is seized and the operator is warned.

D. *User Interface for Online Teaching*

The online teaching is in Figure 4 constituted by the block “Adjust studs”. In the online teaching, the operator should be

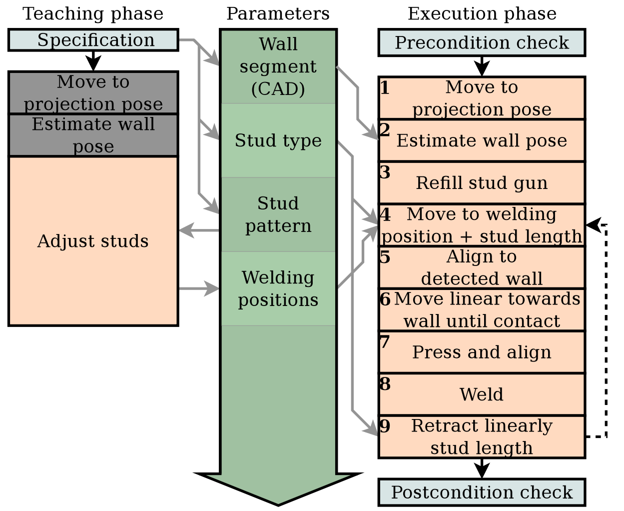


Fig. 4. Structure of the welding skill, corresponding to the general skill concept from Figure 3. The required parameters (middle) are specified during the teaching phase (left) and used during the execution (right). The human is interacting directly with the robot during online teaching which is constituted by the “Adjust studs” step. During execution, the robot should be fully autonomous.

assisted in adjusting stud positions manually on-site as easy and efficient as possible. The stud positions can initially be calculated from the wall model and the stud pattern. The operator must then both be able to add new stud positions and move and delete existing studs positions.

As in most HRI systems, two-way communication is required. The robot must be able to inform the operator about its current state, and the operator must be able to provide new commands to the robot.

1) Robot-to-Human: A traditional approach to interact with the operator would be to provide a graphical interface on a monitor or a tablet, where the stud positions could be adjusted. In this paper, we propose to instead project the stud positions directly to the ship wall segment. If the calibration between the robot and the wall is sufficiently precise, this will allow the operator to focus his attention towards the area of interest instead of towards a screen. We denote this space the *object space*.

2) Human-to-Robot: When the robot projects information directly into object space, the operator should also be able to work in object space. This could either be with human gestures that the robot can detect, or it could with a pointing device that can function as a cursor. For this system, a cursor based solution is chosen the cursor based solution, both because gesture recognition tend to be slow and unreliable, but also because a cursor based solution does not require the operator to be located at a particular location relative to the robot.

The relationship between the pointing device and the projected cursor can either be absolute or relative, as illustrated in Figure 5. With relative cursor control, the cursor must be displayed by the projector and it will therefore only be possible to point in the projectors’ field of projection. With absolute cursor control, a laser pointer or similar can be used as cursor, thereby making it possible to point also outside the projectors’ field of projection. For the current task, the cursor is only required to interact with projected information, however. Therefore the simpler solution is chosen with a projected, relative cursor control. This will also minimize calibration issues.

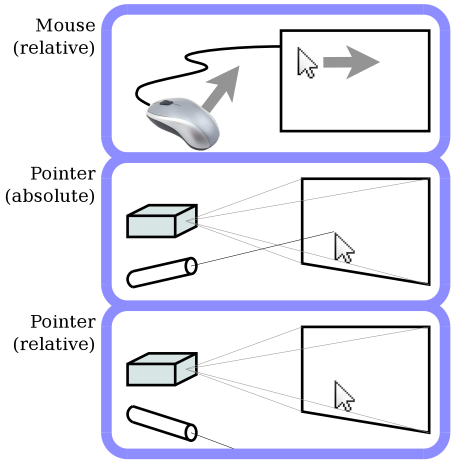
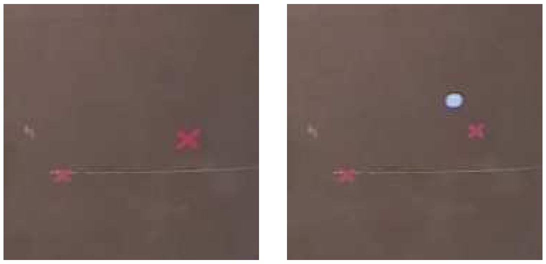


Fig. 5. The relationship between a pointing device and a cursor can be either relative or absolute.

The laboratory setup is shown in Figure 2. To control the cursor, any available IMU device with an accurate accelerometer could in principle be used, and in the setup presented here, a Wii remote has been chosen. In 2(a), the operator is holding the Wii remote and is adjusting stud welding positions on the metal plate. In 2(b), the robot is autonomously welding at the second position. The interface (including the cursor) is projected from the projector mounted on the end effector of the robot arm. Figure 6 shows the user interface close up. In 6(a), the operator is holding and moving the right-most stud position, and in 6(b), the final stud positions have been chosen, and the white cursor is not holding any stud positions.



(a) The cursor us “holding” and adjusting the right-most stud position.

(b) The final positions with the “free” cursor above.

Fig. 6. Teaching of stud positions in object space using the projector. The operator can add, move and/or delete stud positions using a cursor that is controlled using an IMU device. The cursor is projected into object space using the projector mounted on the robot’s end-effector.

IV. EXPERIMENTAL SETUP

The setup used for the results and experiments presented in this paper is based on commercial off-the-shelf components shown in Figure 7. For welding, a capacitor charged KÖCO ESP 1K stud welding gun is used. A Structure IO depth sensor is used for detecting the wall segments, and an AAXA P450 projector is used for providing feedback to the operator. All of these devices are mounted and calibrated on the end-effector of a Kuka LWR 4+ robot arm. The robot features impedance control, and this is used during teaching of a new stud welding task.

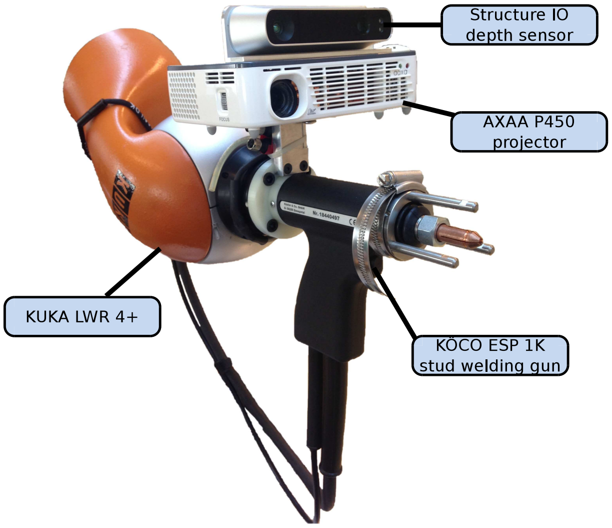
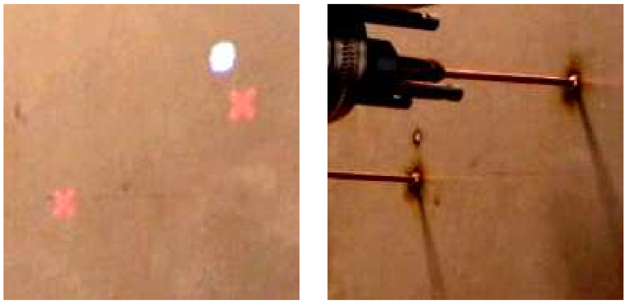


Fig. 7. Robot tool includes stud-welding gun, projector, and 3D sensor.

In the developed interface, stud positions are projected onto the wall itself where studs will eventually be welded. The operator is able to add, remove, and move stud positions using a cursor that is projected onto the wall and controlled by a Wii remote. The cursor is always initially in the middle of the projection area, independently of the (unknown) orientation of the remote. This means that the correspondence between the pointing direction and the cursor may and may not be good. If it is not, the operator is able to fixate the cursor by pressing a button, while moving the remote to a better position. Also, the cursor can never move outside the projection area.

Preliminary tests of the system indicate that a new task can be taught fast with a relatively high precision. For testing, the setup in Figure 2 has been used. An operator repeatedly instructed new welding tasks using the developed interface and each instructed task was then executed. A test was classified as successful if the robot managed to complete welding of all instructed pins.

During testing, one problem became evident. The interface is currently unable to indicate whether taught stud positions are inside or outside the working space of the robot. Whenever a stud was outside the working space, the robot simply stopped during execution. For all stud positions inside the robot's working space, the robot completed the welding successfully. The precision of one stud welding is defined as the distance between the specified stud position and the actual welded stud position. This distance error was in all successful tests below 1cm. An example is shown in Figure 8.



(a) Taught welding positions.

(b) Welded studs.

Fig. 8. Example of stud position measurements.

A video has been made available that shows the performance of the test setup[1]

V. DISCUSSION

This paper presents a system that allows an operator to program and reprogram an AIMM to perform stud welding tasks inside a ship hull during ship construction. The system consists of an intuitive human-robot interface; implemented in a skill-based architecture. Interaction with the robot does not require devices such as computer screens or tablets, which would be impractical in a shipyard. Instead, information is projected from the robot directly into object space, where the actual task is to be carried out. The operator can program/reprogram the task using an IMU device to control a projected cursor. An experimental setup has been implemented to test the potential of the concept.

To further investigate the usefulness of the concept, the HRI should be tested extensively and compared to other types of interaction, including a traditional GUI-based interface. This should make it more clear exactly in which situations a projection based interface will provide an advantage. Both the time it takes to teach a new welding task to the robot as well as the time it takes to learn how to use the interfaces should be evaluated against each other. Also, the precision of the welding positions projected onto the wall compared to the final, welded studs should be measured under varying circumstances.

In the near future, the presented interface will be combined with an interface for controlling a mobile platform and then tested extensively with shipyard workers. Finally, the system will be tested at an actual shipyard in a ship under construction.

ACKNOWLEDGMENT

This research was partially funded by the European Union under the Seventh Framework Programme project 606363 CARLoS - Cooperative Robot for Large Spaces Manufacturing.

REFERENCES

[1] CESA, "Towards a new dawn. europe's response to the global shipbuilding crisis," Brochure, 2010, available: http://www.cesa.eu/download/a2I8OjI1ZzgwZjUxaTIyMz41ZyE==.

---

[1] http://youtu.be/lkL9imXERlE

[2] C. Fernandez-Andres, A. Iborra, B. Alvarez, J. A. Pastor, P. Sanchez, J. M. Fernandez-Merono, and N. Ortega, "Ship shape in europe: cooperative robots in the ship repair industry," *Robotics & Automation Magazine, IEEE*, vol. 12, no. 3, pp. 65–77, 2005.

[3] A. Faina, D. Souto, A. Deibe, F. Lopez-Pea, R. J. Duro, and X. Fernández, "Development of a climbing robot for grit blasting operations in shipyards," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 200–205.

[4] P.-C. Tung, M.-C. Wu, and Y.-R. Hwang, "An image-guided mobile robotic welding system for smaw repair processes," *International Journal of Machine Tools and Manufacture*, vol. 44, no. 11, pp. 1223–1233, 2004.

[5] D. Lee, S. Lee, N. Ku, C. Lim, K.-Y. Lee, T. Kim, and J. Kim, "Development and application of a novel rail runner mechanism for double hull structures of ships," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 3985–3991.

[6] D. Lee, N. Ku, T.-W. Kim, J. Kim, K.-Y. Lee, and Y.-S. Son, "Development and application of an intelligent welding robot system for shipbuilding," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 2, pp. 377–388, 2011.

[7] R. E. Fikes and N. J. Nilsson, "Strips: A new approach to the application of theorem proving to problem solving," *Artificial intelligence*, vol. 2, no. 3, pp. 189–208, 1972.

[8] C. Archibald and E. Petriu, "Model for skills-oriented robot programming (skorp)," in *Optical Engineering and Photonics in Aerospace Sensing*. International Society for Optics and Photonics, 1993, pp. 392–402.

[9] E. Gat, "On three-layer architectures," in *Artificial Intelligence and Mobile Robots*, D. Kortenkamp, P. R. Bonnasso, and R. Murphy, Eds., 1998, pp. 195–210.

[10] Y. Mae, H. Takahashi, K. Ohara, T. Takubo, and T. Arai, "Component-based robot system design for grasping tasks," *Intelligent Service Robotics*, vol. 4, no. 1, pp. 91–98, 2011.

[11] A. Bjorkelund, L. Edstrom, M. Haage, J. Malec, K. Nilsson, P. Nugues, S. G. Robertz, D. Storkle, A. Blomdell, R. Johansson, M. Linderoth, A. Nilsson, A. Robertsson, A. Stolt, and H. Bruyninckx, "On the integration of skilled robot motions for productivity in manufacturing," in *Assembly and Manufacturing, IEEE International Symposium on*, 2011, pp. 1–9.

[12] S. Bøgh, O. Nielsen, M. Pedersen, V. Krüger, and O. Madsen, "Does your robot have skills?" in *Proceedings of the 43rd International Symposium on Robotics*, 2012.

[13] C. Lundberg, C. Barck-Holst, J. Folkeson, and H. I. Christensen, "Pda interface for a field robot," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 3. IEEE, 2003, pp. 2882–2888.

[14] D. Perzanowski, A. C. Schultz, W. Adams, E. Marsh, and M. Bugajska, "Building a multimodal human-robot interface," *Intelligent Systems, IEEE*, vol. 16, no. 1, pp. 16–21, 2001.

[15] S. Muszynski, J. Stückler, and S. Behnke, "Adjustable autonomy for mobile teleoperation of personal service robots," in *RO-MAN, 2012 IEEE*, Sept 2012, pp. 933–940.

[16] S. Mitra and T. Acharya, "Gesture recognition: A survey," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 3, pp. 311–324, 2007.

[17] M. R. Pedersen, C. Hoilund, and V. Kruger, "Using human gestures and generic skills to instruct a mobile robot arm in a feeder filling scenario," in *Mechatronics and Automation (ICMA), 2012 International Conference on*. IEEE, 2012, pp. 243–248.

[18] C. P. Quintero, R. T. Fomena, A. Shademan, N. Wolleb, T. Dick, and M. Jagersand, "Sepo: Selecting by pointing as an intuitive human-robot command interface," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1166–1171.

[19] V. Alvarez-Santos, R. Iglesias, X. M. Pardo, C. V. Regueiro, and A. Canedo-Rodriguez, "Gesture-based interaction with voice feedback for a tour-guide robot," *Journal of Visual Communication and Image Representation*, vol. 25, no. 2, pp. 499–509, 2014.

[20] A. Wilson and S. Shafer, "Xwand: Ui for intelligent spaces," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2003, pp. 545–552.

[21] A. Wilson and H. Pham, "Pointing in intelligent environments with the worldcursor." in *INTERACT*, 2003.

[22] K. Ishii, S. Zhao, M. Inami, T. Igarashi, and M. Imai, "Designing laser gesture interface for robot control," in *Human-Computer Interaction–INTERACT 2009*. Springer, 2009, pp. 479–492.

[23] C. C. Kemp, C. D. Anderson, H. Nguyen, A. J. Trevor, and Z. Xu, "A point-and-click interface for the real world: laser designation of objects for mobile manipulation," in *Human-Robot Interaction (HRI), 2008 3rd ACM/IEEE International Conference on*. IEEE, 2008, pp. 241–248.