Proceedings of the OAGM&ARW Joint Workshop 2017 DOI: 10.3217/978-3-85125-524-9-15

Toward Safe Perception in Human-Robot Interaction

Inka Brijacak, Saeed Yahyanejad, Bernhard Reiterer and Michael Hofbaur[1]

Abstract— **Perception is a major component of a system when it comes to the concept of safety in human-robot interaction. Although designing a mechanically safe robot may reduce lots of potential hazards, it is still beneficiary or even required to have detailed knowledge of the current status of the robot, human, and other environmental entities. We refer to this knowledge as perceptional awareness, or simply perception, that subsumes: (i) what our system perceives from robot state and its environment, (ii) what our system perceives from human state, and (iii) what a human perceives from the robot state. In this paper we provide requirements for a holistic architecture to construct safe perception using multiple heterogeneous and independent sensors and processing units in any environment that includes both robots and humans. We also illustrate our concepts on the basis of particular instances of this scheme realized in the robotic lab.**

I. INTRODUCTION

Nowadays, robots are being used widely in different fields due to their precision, accuracy, reliability, and easy deployment. In many initial applications of robots, they are functioning separated from humans in isolated areas. With advances of technology and the necessity for coexistence of robots and humans (e.g., medical application, service robots, collaborative production lines), the new era of human-robot interaction (HRI) has emerged. HRI studies and describes the types and characteristics of the possible interactions that can exist between a robot and a human.

When a human is working in a close distance with robots, the safety of the human becomes an important issue. Initially, safety requirements for many industrial robotic applications were achieved just by a physical separation of humans from any robot (e.g., using barriers or fences). This simple and effective way to impose safety, however, prevents direct interaction between humans and robots to work collaboratively. The relevant international standard for safety in industrial robots [10], [11], which specifies accepted means to impose safety, however, allows also human-robot collaboration in four clearly defined scenarios. The new technical specification ISO TS 15066 "Robots and robotic devices – Collaborative robots" [12] provides even more details on these operational settings and specifies comprehensive force, pressure, and speed limits for unintended human-robot interactions (collisions).

Risk reduction during human-robot interaction has three main approaches: (i) redesigning the system and the task realization, (ii) using functional or physical safeguards, and (iii) raising the awareness of the operator/user, either using active warnings during operation and/or by specific training. Taken into account the industrial experience, redesigning the system is the most effective risk reduction strategy and should always be applied first. However, when operating adaptively in less structured environments, redesign alone is often insufficient, and additional functional safety measures are mandatory [13].

It is possible to combine these three approaches to achieve higher levels of safety. In spite of that, no matter how accurate a system is designed, the continuous monitoring (the second approach mentioned above) is an important factor for a safe system. To be able to understand the status of the environment or a system, the concept of perception plays an important role. Similar to human perception, the concept of the perception for a system can be twofold:

- External: What a system sees, perceives, or understands from the environment, i.e., what types of object are around me? What are their positions, speed, shape, size? What are the states of other systems around me?
- Internal: What a system sees, perceives, or understands about itself, i.e., where should I be? Where am I? What is my current state?

For both of these perception types, we need dedicated sensors to obtain relevant data upon for perception. This demanding task requires to deal with the following issues:

- sensory data acquisition and storing the data
- data mining, enhancement, and filtering
- sensor fusion
- time synchronization
- dependable, safety-enabled operation.

The complexity highly increases with the larger number of heterogeneous sensors such as, safety-enabled laser scanners (LIDARs), RGB cameras, thermal cameras, time-of-flight (ToF) cameras, haptic sensors, proximity sensors, ultrasonic sensors, robot internal sensors (e.g., torque sensors), pressure sensors, etc. Redundancy achieved by using diverse sensor types highly improves the reliability of the overall perception unit. Dealing with diverse sensors requires one to carefully consider the different interfaces, data types, sampling rates and, of course, a potentially large amount of data. In order to deploy such an inclusive perception scheme in real-world robotic systems, however, it is also important to consider the requirements set by the relevant standards that include the entire life cycle of the system starting with the development process itself, hard- and software-requirements and functional issues for all forms of the system's operation. This goes far beyond the requirements necessary to realize a laboratory demonstrator. As a consequence, it is helpful to

[1] All authors are with JOANNEUM RESEARCH ROBOTICS - Institute for Robotics and Mechatronics - Cognitive Robotics Group, Austria <firstname.lastname>@joanneum.at

consider these aspects early in the R&D efforts in order to qualify for real-world applications.

II. Related Work

With the growing applications of robotics in human life and considering the high importance of human safety in HRI, more and more research is being dedicated to assure a safe perception in collaborative environments. Kulić [13] was one of the first who has provided an extended and detailed overview regarding the safety in HRI. She managed to define many important terms in this scope and formulate a metric for danger measurement. [20] also provides a quick overview of safety issues in HRI. The work done in [16] categorizes the safety strategies in three categories: I) crash safety (e.g., limiting the power/force), II) active safety (e.g., by using proximity sensors, vision systems and force/contact sensors), and III) adaptive safety for intervening in the robot operation and applying corrective actions that lead to collision avoidance without stopping the robot operation. In their work and also the work from [6] the focus is more on the design principles. The latter also considers robustness, fast reaction time, and context awareness as the main parameters of a safe design. One interesting and genuine idea mentioned there and originally in [7] is the recommendation to design the robots such that they are predictable for a human. For instance, by using special sounds or human-like movements for a robot, the human can expect and foresee the robot's moves and accordingly avoid unintended collisions with the robot.

Some other researchers just focus on detecting and localizing the human and accordingly prevent the robot from colliding with it. Depending on the type of the detecting sensor, their performance is evaluated. Active or marker-based sensors may be more challenging to implement and less convenient to apply in real scenarios, but they can provide a quite accurate and reliable collision avoidance. Their proximity distances can reach up to a few centimeters between human and robot [14]. On the other hand, other types of sensors such as cameras and laser scanners may have higher error ranges, but by combining multiple sensors together we can minimize the risk. In this direction, [18] fuses data from multiple heterogeneous 3D sensors to detect any moving object approaching the robot. Similar work has been done by [19], which constructs point clouds and 3D models of the moving objects and the robots in order to avoid collisions.

Safety of a human is not always achieved by immediate protection from danger or collision. Sometimes hazards can be results of long-term inappropriate actions in HRI. For instance, [4] looks at the human safety from an ergonomic aspect, which is a complementary point of view. They consider a work environment which ensures the occupational safety and describe the requirements for a workplace where human and robot can jointly perform an assembly process without separation between their workspaces. They also consider some human factors such as the age of the working person.

In our work we are going to look at the safe perception in HRI with a holistic approach. We are going to explain what kinds of criteria are necessary to be obeyed in order to have a safe perception architecture and why a single safety precaution will fail.

III. Risk analysis in safe perception

In the design of a robot system, risk assessment is a main measure for achieving standards-compliant safety. The general process, consisting of risk analysis and risk evaluation, is described in [9], with extensions specific to robots given in [11] and [12]. First, the potential hazards of the robot system during all phases of its life cycle are to be identified. The hazards given in the annex of [11] may be seen as a list of examples, which must always be considered and carefully examined with the specific robot system and its application/task in mind. All identified hazards are then to be evaluated in terms of their risks. From the obtained results it may become clear that the risks have to be reduced by certain measures, leading also to updates in the results of risk analysis and evaluation, and thus a new iteration of the process steps. Eventually the residual risks of any remaining hazards are sufficiently low to allow for the designed robot system to be realized and considered safe.

Risk assessment and safe perception influence each other in several ways. Already during the risk analysis, the capabilities gained from the perception infrastructure can serve as measures that counter certain hazards and reduce risks. But the risk analysis must also consider any potential hazards arising from system components, including also those that are specifically employed for safety reasons. If the used components do not provide a sufficiently high integrity / performance level or they are placed or configured in suboptimal ways, their total effect may be deteriorating. However, such choices will typically be identified and mitigated in the further course of the iterative risk assessment and risk reduction, so that the final solution is able to achieve the required safety properties. When modifying the system design to achieve risk reduction, the integration of the safety perception infrastructure or the modification of its integration can be a central measure. Thus, as one of its results, the iterative process of risk assessment and risk reduction gives constraints on effective sensor placement that enables comprehensive sensor fusion later on. An example of such an improvement can be seen in the step from the arrangement depicted in Fig. 3 to the one in Fig. 5. In the running system, any residual risks are permanently relevant. The setup of the safe perception must be designed in a way that potential hazards that could not be eliminated by system design can be prevented or dealt with accordingly based upon perception.

Finally, the operation of the system continuously gives opportunities to gather new knowledge that can be used in a refined risk assessment to further improve the system's safety. This could be done any time, but is necessary in particular when the system is going to be modified. Possible inputs may come from user feedback, other persons observing the operation, or also the system itself. For the latter, we envision a component that is able to identify events that may need further offline analysis later using different kinds of potentially related data available to the system.

IV. Safe perception architecture requirements

An architecture for a safe perception system typically includes components of the types machine, sensor, human, and processing unit. To construct a suitable architecture, we need a good understanding of these components in terms of their functionality and reliability as well as their relations and interfaces. Here we are going to propose a generic architecture by pointing out the requirements which enable the realization of a safe perception system for a typical collaborative robot system. This architecture should be independent from the robot type, size of the workspace, and environmental factors as much as possible and also easy to deploy. In order to achieve such a goal, we have to consider the possibilities of failure of individual components in a system as discussed in Section III. Accordingly, an ideal safe architecture considers/includes the following requirements:

- Embed safety inside different building blocks: consider safety not just as an add-on but embed it in each system component, robot, planning, and programming decisions. However, always keep the distinction between operational functionality and safety functionality in mind.
- A modular architecture: makes it easy to add/remove various hardware and software components. For instance, Robot Operating System (ROS) [17] has been used for our modular software architecture to provide a simple message passing and hardware abstraction.
- Adding parallel redundancy: use multiple sensors in parallel over independent platforms to make sure that the failure of one is not causing the whole system to fail.
- Heterogeneous system: using different types of sensors (e.g., laser scanner, time-of-flight camera, thermal camera, speech recognition, light curtain, etc.) to make sure that the system is robust against changing environmental variables. For example, if there are poor visibility conditions at the workplace, conventional cameras may fail to obtain a picture but a thermal or time-of-flight (ToF) camera can help and even provide images through fog or smoke.
- Reproducibility: which makes it easy to re-implement in different scenarios and setup the perception system in other new workspaces.
- Mapping the Environment: modeling the 3D environment in order to further simulate, localize and position the sensors and objects in the environment. This helps to decide how and where to mount the sensors to achieve the maximum coverage (high spatial distribution helps the robustness in case of local failures).
- Context aware: takes the context of the ongoing scenario into account either by receiving it from operator or by analyzing the scene. Accordingly the system adapts the parameters and decision-making to that specific scenario.
- Intelligent: learn from the previous situations (from both false-positives and false-negatives) and hence provide feedback data and parameter correction for future improvement. Using machine learning in robot perception is an example to achieve this goal.
- Exploiting human perception: warn the human about the potential hazards. Unlike the conventional sensory perception, we do not only inform the human in close-to-danger scenarios. Instead, we additionally count on human perception by constantly giving a feedback regarding the state of the robot to the human, for example by producing a sound according to the movements of the robot. This way the human herself/himself can make a decision if she/he feels something is out of the order.

As mentioned above, redundancy is a major design paradigm to realize safety through perception. Relevant standards such as the previously mentioned ISO 10218 and ISO 13849 enforce redundancy throughout the system for achieving a required performance level for a safety function, i.e. redundancy in sensors, computational units and actuators as indicated in Figure 1.

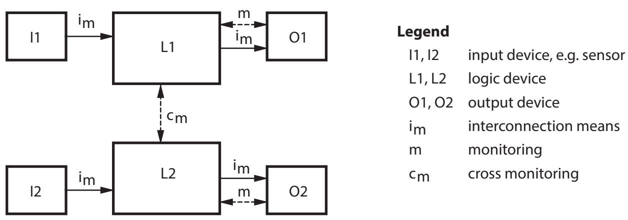

Fig. 1. Redundant Safety Architecture (cat. 3, ISO 13849-1 cl. 6.2.6)

This classic layout for achieving a high integrity / performance level has to be incorporated carefully as not to tamper with the safety of the overall system. This is important in particular as our complex robot system will involve both safety functionality at high integrity level as well as functional components with lower integrity level that should also contribute valuable information to improve the overall safety. In industry, one typically talks about *yellow* and *gray* components, referring to high integrity safety and general functional components, respectively. A clear structure, both in terms of hardware and software, is required in order to obtain the safety functionality at the desired performance levels.

V. Architecture realization

In our lab we have various types of serial robotic manipulators in workspaces where safe human-robot interaction or collaboration is compulsory. Therefore, we utilize sensors for highly dependable perception using safety LIDARS (yellow hardware – OMRON OS32c) at performance level D (PLd) [8]. On the other hand, we intend to use functionally powerful time-of-flight (ToF) cameras (gray hardware - PMD Pico Flexx) for environmental perception. Similarly, the control of the robots involve the low-level safety-enabled robot controllers (yellow hardware/software) in combination with a high-level control system that is implemented in ROS (gray hardware/software). The overall system should not just act as a ROS system with add-on safety, but integrate safety inclusively.

We propose a safety-enabled system architecture that solves the safe robot perception and control task through 3 levels of hardware abstraction. Basis for this architecture that is given in Figure 2 is a safety-rated robot controller (in our case the KUKA Sunrise controller for the sensitive iiwa robot). High level control is implemented in ROS running on separate (Linux-based) controllers. In between those two control layers, we introduce a safety-rated controller (e.g. a safety PLC) that connects to both, safety-rated sensors (safety LIDARS in our case) and the safety-rated input of the low-level controller. This allows us to implement dependable safety functionality that goes beyond the simple safety-logic of the low-level controller. However, it might also be implemented directly on the low level controller if the device offers to implement high integrity safety functions. This layered model clearly defines a priority structure where the safety-enabled control system takes control whenever a critical safety issue is detected. Thus, there is no direct connection that allows the ROS System to issue control actions for the low level controller except the authorized connection through this safety control layer.

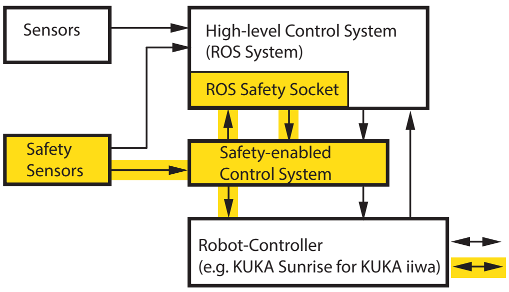

Fig. 2. Safety-enabled Architecture

Up to now, this structure resembles the classic add-on safety architecture. However, we intend to go beyond this architecture that will enable more inclusive perception and control schemes. As a consequence, we propose to provide a highly dependable ROS safety socket that connects the safety-controller to the ROS environment. Furthermore safety sensors could be connected to the ROS environment as well. For example our safety LIDARs provide safety-enabled outputs that define region interceptions through (safe) binary signals, whereas the more informative LIDAR scan is provided through standard interfaces to the ROS system. With our safety socket, we intend to enable ROS functionality not just at different levels of priority, but also at different levels of dependability. This safety-socket is only one pre-requisite. We also have to provide dependable and in particular trustworthy ROS nodes and communication between them and the socket. The standard ROS system does not address IT security adequately [15]. To compensate for this security flaw, our institute colleagues recently proposed a scheme for application-level security and safe communication [3], [1] for ROS that is now under consideration by the Open Source Robotics Foundation (OSRF) to be included in the SROS project for future public release.

Alongside of this implementation effort that will provide the necessary building blocks for a safety-rated perception and control functionality, we evaluated possibilities for functionally rich and safe multi-sensory perception using the standard ROS environment as an experimental testbed. We have set up a heterogeneous perception system comprising of two safety-rated OMRON OS32C laser scanners with data fusion running on two different computers and one or two ToF cameras for acquiring 3D data from the environment (the aforementioned PMD Pico Flexx camera and the single-beam ToF sensors Terraranger). We consider the proper combination of different technologies of parallel and independent sensors and the resulting high redundancy as a prerequisite for fulfilling safety requirements. Additionally, to achieve robustness in case of local failures, it is necessary to mount the sensors in a distributed way. As a basis for making safety-related decisions in the running system, we are going to define a distinction of three danger zones that are reported by our sensor fusion: *Danger*, *Warning*, and *Safe*. Their origin is in the origin of a robot, and they are surrounding the robot in a circular way. The border between *danger* and *warning* zones is defined using safety separation distance defined in ISO/TS 15066 [12]. Using distance of a moving object from a depth sensor, it will be decided in which danger zone the movement is detected.

The example setup of sensors which is shown in Figure 3 results the sensor fusion shown in Figure 4. Sensors are mounted close to each other, which leads to a higher chance for all sensors to fail together when a local hazard happens (e.g. physical damages). Knowing that, and also for a specific collaborative use case, sensors are mounted as shown in Figure 5. Regarding modular architecture and reproducibility, it is also very easy to change the mounting for other use-cases and workspaces. However, more automatized setup of sensors for maximum coverage of the workspace and their calibration is planned for the future work.

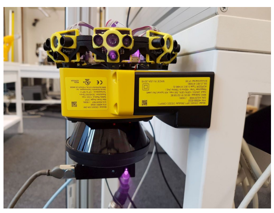

Fig. 3. Example of a problematic setup where 3 different types of sensors are mounted just next to each other. This setup increases the chance of perception failure due shadowing effects and local hazards such as physical damages.

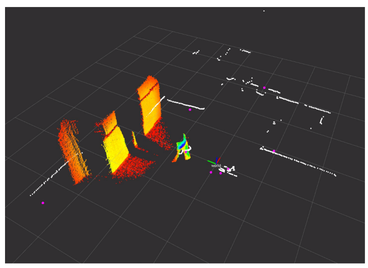

Fig. 4. Visualization of the 3D position data in RViz obtained from Teraranger Tower (8 pink points), Pico Flexx Camboard ToF camera (colored points), and laser scanner (white points).

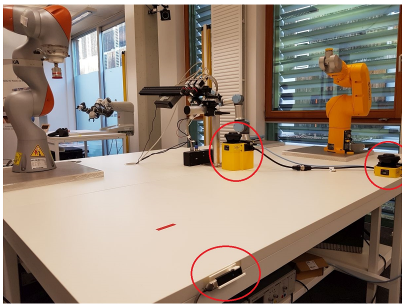

Fig. 5. Distributed setup of sensors including 2 laser scanners and one Pico Flexx ToF camera around the workplace.

The sensors we have chosen are not supposed to be used for object/human detection and localization per se, but mainly for distance measurement. Therefore, with both types of sensors, we need to perform some post-processing in order to be able to detect and perceive an approaching object. In order to have a safety-eligible sensory data analysis and decision making, we need to reduce the chance of false positive and false negative in our perception system. Safety-wise, perception scenarios with false negative (i.e., a human approaching the robot is not detected) are far more dangerous compared to scenarios with false positive. In case of false positive, on the other hand, we may observe instances of unwanted robot speed reduction or even a complete stop, which is affecting the system performance but not the safety property. For instance, in case of ToF camera the following steps are being performed to robustly detect a moving object:

- **Filtering** the depth image: it is performed by using various filtering method (e.g., median filter in both spatial and temporal domain) which mitigates the false detection. Filtering steps are shown in Figures 6 and 7, and resulting filtered depth image is shown in Figure 8.
- **Background image**: recording filtered depth image at startup. The background image is refreshed if there is no movement detected for a specific period of time (Figure 8).
- **Difference image**: subtraction of background and current depth image (Figure 8 – Figure 10 = Figure 12).
- **Blob Detection** in binary difference image (Figure 13). To avoid detecting changes produced by noise and also using the prior-knowledge of the size of an approaching object (e.g., human) we adjust the parameters of our blob detector (such as expected shape and size) in a way to detect only the intended moving targets. When at least one blob is detected, it means that there is a movement in the workspace, and therefore we can proceed with the next two steps.
- **Masking** the original depth image: binary difference image is used as a mask in order to have real depth data of each pixel of the blob that is assumed to be a moving object (Figure 11).
- **Final depth information** of detected moving object: is a result of using median value of depth info from the masked image. Higher importance is given to closer distances that still have a smaller covering area in the depth image, such as an intruding arm of a human.

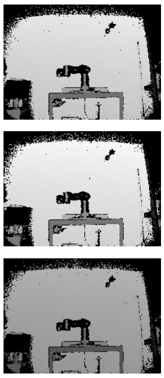

Fig. 6. Original depth images

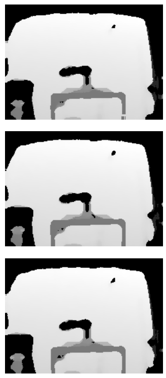

Fig. 7. Filtered depth images

Fig. 8. Final background image

Fig. 9. Original images with human

Fig. 10. Final filtered depth image

Fig. 12. Difference image

Fig. 11. Masked original depth image

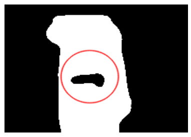

Fig. 13. Blob detection in BW diff. image

For the laser scanners, which provide 2D data (scan a plane or a cross-section out of the 3D space), the process of extracting distance of a detected object and its coordinates is aligned with the one of the ToF camera:

- **Background** data: resulting background data is median filter applied on temporal domain of data collected in initialization step.
- **Difference** data is calculated as subtraction of background data and current data every time stamp.
- **Movement** in the workspace is detected if the percentage of not moving points is less than 98.5%.
- **Transformation** of depth data from the laser coordinate system to the robot's coordinate system is done using Euclidean distance, taken into account the fixed position of the laser scanner relative to the robot.

Every time stamp we have the result of our sensor fusion as the final danger zone. From each sensor, regarding distance of a human, or any other moving object in robot's workspace, it is decided in which danger zone the detection happened, and the final danger zone is the worst case of all three. Measuring the separation distance between the object/human and the robot, in constant speed setting situations with worst-case value taken into account, it is ensured that the robot system never gets closer to the operator than the protective separation distance [12].

While it is essential to have a direct link from the safety-oriented sensor fusion to the robot control for adapting speed limitations or triggering an emergency stop, the combined information from the sensors also serves as a valuable input for generating task-level plans for the robot system. We use ROSPlan [2] as our infrastructure for task planning, which allows us to formulate the planning domain in the quasi-standard Planning Domain Definition Language (PDDL) [5]. The planner, given abstract logical models of the system and relevant entities in its surroundings on the one hand and goals to be achieved on the other, would typically generate sequences of actions such as picking up a certain object, placing it in a certain pose into the product that is being built, and fixing it there in a certain manner, using a certain path of motion trajectories from a set of possible ones. There could also be actions representing interaction with humans via user interface components or invoking arbitrary meaningful functionalities of connected devices.

The currently obtained safety zone information and other results of sensor fusion can be mapped to logical facts in the planning domain, and they in turn can be used in the conditions of PDDL actions in order to tie their applicability to the current safety situation. Examples for such conditional safety limitations include forbidding certain actions as a whole, forbidding trajectories in which parts of the robot would intrude certain zones or exceed a certain speed limit, forbidding interacting with potentially hazardous objects, or forcing the robot to assume a predefined home pose between any two other poses. The planning system takes care that such restrictions are not only considered when a new plan is generated but also that the current plan's execution is halted when an assumed precondition, safety-related or other, for a robot action is found to be not actually fulfilled, or when an action's execution was not successful. Then, starting from the updated current state, a new plan is generated and goes into effect.

VI. Conclusion

In this paper, we have emphasized the importance of a safe perception system in HRI scenarios where both human and robot coexist in a shared environment and collaborate toward their goals. We have taken into account a holistic approach toward safe perception and managed to introduce the requirements for a general architecture that integrates safety in any robotic environment independent of scenario, scale, shape, and the number of robots and humans. This architecture is modular, reproducible, context aware, intelligent and also has parallel redundancy, heterogeneous sensors, and embedded safety.

Furthermore we have presented how our safe perception is set up for a collaboration scenario in our lab to demonstrate the simplicity and reusability of our approach in real-world applications. In this demonstration multiple safety standards have been considered and included in order to have a correct risk analysis and safety-zone calculation.

References

[1] B. Breiling, B. Dieber, and P. Schartner, "Secure communication for the robot operating systems," in *Proceedings of the 11th Annual IEEE International Systems Conference*, 2017, to appear.

[2] M. Cashmore, M. Fox, D. Long, D. Magazzeni, B. Ridder, A. Carrera, N. Palomeras, N. Hurtós, and M. Carreras, "Rosplan: Planning in the robot operating system," in *Proceedings of the Twenty-Fifth International Conference on Automated Planning and Scheduling, ICAPS 2015, Jerusalem, Israel, June 7-11, 2015.*, 2015, pp. 333–341. [Online]. Available: http://www.aaai.org/ocs/index.php/ICAPS/ICAPS15/paper/view/10619

[3] B. Dieber, S. Kacianka, S. Rass, and P. Schartner, "Application-level security for ros-based applications," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 4477–4482.

[4] M. Faber, J. Bützler, and C. M. Schlick, "Human-robot cooperation in future production systems: Analysis of requirements for designing an ergonomic work system," *Procedia Manufacturing*, vol. 3, pp. 510 – 517, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2351978915002164

[5] M. Fox and D. Long, "PDDL2.1: an extension to PDDL for expressing temporal planning domains," *J. Artif. Intell. Res. (JAIR)*, vol. 20, pp. 61–124, 2003. [Online]. Available: http://dx.doi.org/10.1613/jair.1129

[6] M. Giuliani, C. Lenz, T. Müller, M. Rickert, and A. Knoll, "Design principles for safety in human-robot interaction," *International Journal of Social Robotics*, vol. 2, no. 3, pp. 253–274, 2010. [Online]. Available: http://dx.doi.org/10.1007/s12369-010-0052-0

[7] M. Huber, H. Radrich, C. Wendt, M. Rickert, A. Knoll, T. Brandt, and S. Glasauer, "Evaluation of a novel biologically inspired trajectory generator in human-robot interaction," in *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication*, Toyama, Japan, 2009, pp. 639–644.

[8] ISO, *ISO 13849-1:2006: Safety of machinery – Safety-related parts of control systems – Part 1: General principles for design.* Geneva, Switzerland: International Organization for Standardization, Nov. 2006.

[9] ISO, *ISO 12100:2010: Safety of machinery – General principles for design – Risk assessment and risk reduction.* Geneva, Switzerland: International Organization for Standardization, 2010.

[10] ISO, *ISO 10218-1:2011: Robots and robotic devices - Safety requirements for industrial robots - Part 1: Robots.* Geneva, Switzerland: International Organization for Standardization, July 2011.

[11] ISO, *ISO 10218-2:2011: Robots and robotic devices - Safety requirements for industrial robots - Part 2: Robot systems and integration.* Geneva, Switzerland: International Organization for Standardization, July 2011.

[12] ISO, *ISO/TS 15066:2016: Robots and robotic devices – Collaborative robots.* Geneva, Switzerland: International Organization for Standardization, Feb. 2016.

[13] D. Kulić, "Safety for human-robot interaction," Ph.D. dissertation, University of British Columbia, 2006.

[14] P. A. Lasota, G. F. Rossano, and J. A. Shah, "Toward safe close-proximity human-robot interaction with standard industrial robots." in *CASE*. IEEE, 2014, pp. 339–344.

[15] J. McClean, C. Stull, C. Farrar, and D. Mascareas, "A preliminary cyber-physical security assessment of the robot operating system (ros)," in *Proc. SPIE*, vol. 8741, 2013, pp. 874 110–874 110–8. [Online]. Available: http://dx.doi.org/10.1117/12.2016189

[16] G. Michalos, S. Makris, P. Tsarouchi, T. Guasch, D. Kontovrakis, and G. Chryssolouris, "Design considerations for safe human-robot collaborative workplaces," *Procedia* {*CIRP*}, vol. 37, pp. 248 – 253, 2015.

[17] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[18] P. E. Rybski, P. Anderson-Sprecher, D. Huber, C. Niessl, and R. G. Simmons, "Sensor fusion for human safety in industrial workcells," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, 2012, pp. 3612–3619. [Online]. Available: http://dx.doi.org/10.1109/IROS.2012.6386034

[19] B. Schmidt and L. Wang, "Contact-less and programming-less human-robot collaboration," *Procedia* {*CIRP*}, vol. 7, pp. 545 – 550, 2013.

[20] M. Vasic and A. Billard, "Safety issues in human-robot interactions." in *ICRA*. IEEE, 2013, pp. 197–204.