**Noname manuscript No.**
(will be inserted by the editor)

# iMRK - Demonstrator for Intelligent and Intuitive Human-Robot Collaboration in Industrial Manufacturing

**José de Gea Fernández · Dennis Mronga · Martin Günther · Malte Wirkus · Martin Schröer · Stefan Stiene · Elsa Kirchner · Vinzenz Bargsten · Timo Bänziger · Johannes Teiwes · Thomas Krüger · Frank Kirchner**

Received: date / Accepted: date

**Abstract** This report describes an intelligent and intuitive dual-arm robotic system for industrial human-robot collaboration which provides the basis for further work between DFKI (Robotics Innovation Center) and Volkswagen Group (Smart Production Lab) in the field of intuitive and safe collaborative robotics in manufacturing scenarios. The final robot demonstrator developed in a pilot project possesses multiple sensor modalities for environment monitoring and is equipped with the ability for online collision-free dual-arm manipulation in a shared human-robot workspace. Moreover, the robot can be controlled via simple human gestures. The capabilities of the robotic system were validated at a mockup of a gearbox assembly station at a Volkswagen factory.

## 1 Introduction

Industrial manufacturing is undergoing major changes and transformations. New technologies are introduced by digitization; consumers are demanding that the manufactured products are produced in an increasing number of variants while the workforce's average age is shifted by the demographic change. All of these factors are important drivers for introducing workplaces where human and robot work together. On the other side, the area of human-robot collaboration has experienced a significant increase of interest in the past years, first from the research community and, more recently, from the industrial community as well. The reason lies in key enabling technologies appearing on the market, probably most importantly a new generation of lightweight robots which incorporate different concepts (control software or mechatronic design, see for instance [4, 2, 1]) to allow the interaction with humans while ensuring a certain degree of safety.

In this context, the current work aims at integrating current technologies in different areas to create an innovative robotic system for a safe and intuitive human-robot collaboration. Multisensor-based workspace monitoring and tracking algorithms were linked to collision avoidance algorithms to create a robotic system with the capability of real-time avoidance of collisions both with itself and with external objects. A jacket was sensorized to be used as a gesture recognition device worn by the user in order to command the robotic system. In addition, a set of whole-body controllers is used as building block that describes single actions of a high-level robot behaviour plan. Finally, a modular, robot-agnostic software control framework was used to seamlessly bind all components together and allow reusing generic software components to describe a variety of complex manipulation behaviours, whilst keeping independence from the particular robot hardware.

---

José de Gea Fernández · Dennis Mronga · Martin Günther · Martin Schröer · Stefan Stiene · Elsa Kirchner · Frank Kirchner
DFKI Robotics Innovation Center
Robert-Hooke Str. 1
28359 Bremen, Germany
E-mail: jose.de_gea_fernandez@dfki.de

Vinzenz Bargsten · Elsa Kirchner · Frank Kirchner
Robotics Group of the University of Bremen
Robert-Hooke Str. 1
28359 Bremen, Germany

Timo Bänziger · Johannes Teiwes · Thomas Krüger
Volkswagen AG
Berliner Ring 2
38436 Wolfsburg, Germany

## 2 System Description

The developed robotic system is based on two KUKA iiwa lightweight robots equipped with 3-finger grippers from Robotiq (see Fig. 1). Moreover, three RGB-D cameras (ASUS Xtion Pro Live) monitor the common human-robot shared workspace to ensure real-time collision-free robot movements. For monitoring the surroundings of the system, two SICK LMS100 laser scanners are used, which are mounted on opposite corners of the table and jointly perceive a 360° view of the area around the robot.

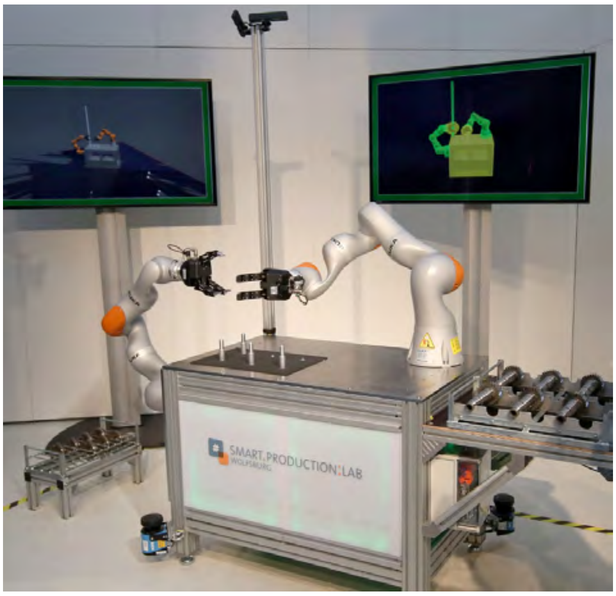

**Fig. 1** Robotic system developed.

The accompanying developed control software is based on the DFKI's software framework Rock [3] which integrates the multiple software components and allows a fast reconfiguration of the task to be automated. The current robotic system possesses three working modes:

- Automatic mode: There are no humans detected in the shared human-robot workspace or approaching it. The robot executes an automatic task at the desired operating speed.
- Approaching mode: The robot detects the intention of humans to enter the robot's workspace based on the information from the laser scanners; in such a case, the robot goes on with its automatic task but at reduced speed as long as the human is around. In this mode, the robot is ready to receive commands via gestures.
- Interaction mode: The robot is set in compliant mode to allow a direct physical contact with humans (e.g., to inspect the part being held by the robot).

The automatic switch between these modes occurs by processing the real-time sensor data and/or the gestures performed by the operator. Hence, the collaborative skills of the robot are based on the seamless integration of:

- Multimodal sensor-based person tracking using RGB-D and laser scanner data
- Real-time identification and tracking of point clouds in the workspace to be used for collision avoidance
- Real-time dual-arm self-collision avoidance and dynamic collision avoidance with external objects
- Robot's speed and compliance automatically adjusted depending on the current working mode and real-time environment data
- Intention recognition and recognition of simple human gestures

A whole-body control software framework is used as a key control element whose controller-constraints combinations constitute the basic building blocks that describe actions of a high-level action plan to be sequentially and autonomously executed. A modular, robot-agnostic software control framework is used to bind all components together and allow reusing generic software components to describe a variety of complex manipulation behaviours.

## 3 Software Components

The software architecture used in this work can be divided into three main blocks (see Fig. 2):

- the Software Component Network
- the Component Network Supervision
- the Application Logic .

The Software Component Network is composed of Rock components using Syskit and the extensions that were introduced in [7]. In the Software Component Network, sensor processing or control algorithms are arranged into a topology creating the controller loop that generates the control commands for the robots. A schematic overview of the Software Component Network is given in Figure 2 (top). Sensor data is fed into the sensor processing pipelines that are used for detecting external objects in the environment. The resulting object point clouds are processed by a potential field-based collision avoidance controller.

The robot manipulation tasks are described as a number of joint waypoints or object-relative Cartesian waypoints. The corresponding components are referred to as *Setpoints Generation* in Figure 2. A Whole-Body Control (WBC) component weighs and prioritizes the different controller inputs and, taking into account a set

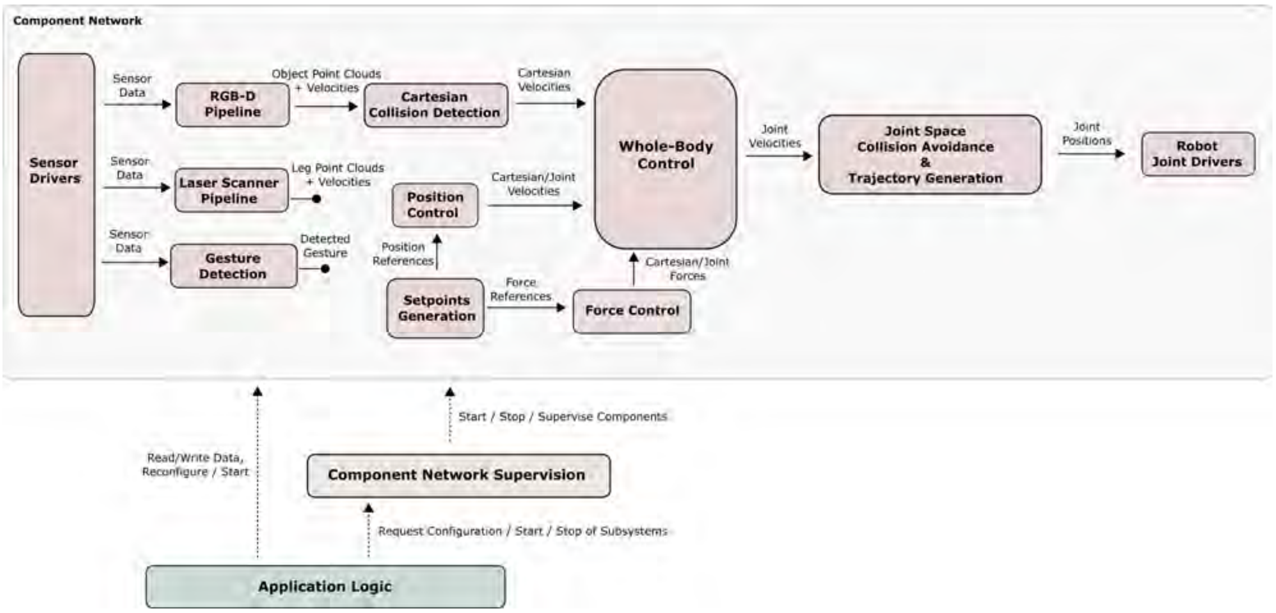

**Fig. 2** Overview of the main software components.

of constraints, outputs the most optimal joint velocity commands.

A further control layer is included between the WBC component and the robot drivers, in which both self and external collisions are ascertained in joint space. Moreover, a trajectory generation component generates smooth and synchronized motion commands for the two robot arms.

The laser scanner pipeline – detecting whether someone entered the workspace – as well as the gesture detection module are also part of the Software Component Network, but have no direct data flow connection to other Rock control components. The data generated by these components is interpreted by the Application Logic.

Finally, a Component Network Supervision module takes care of launching the corresponding processes of the components and supervise their execution. In this way, if any component fails, it will be detected and the robot control will be immediately stopped.

## 4 Collision Avoidance

The demonstrator robot is equipped with a heterogeneous set of sensors: three RGB-D cameras and two 2D laser scanners. Through a series of processing steps, point clouds corresponding to potential collision objects (such as persons, tools or carts) are extracted from the RGB-D data and tracked. These object point clouds are directly passed on to the dynamic collision avoidance module, which (if necessary) rapidly executes an avoidance motion using our whole-body control framework.

Our approach for self-collision avoidance is based on the KCCD library [6], which contains methods to rapidly evaluate distances between links of a robotic manipulator and compute ideal brake timing for its actuators. In KCCD, a rigid body is described by the means of a convex hull enclosing a finite set of points $p_i$, $i = 1, \dots, n$, which is extended by a buffer radius $r$. The complete collision model of the dual arm system can be seen in Figure 4. If the robot is moving, the collision volumes grow and shrink according to a braking model that includes the current moving velocity and desired deceleration of each joint. This way, the optimal brake timing can be determined, slowing down the manipulator smoothly rather than performing a hard stop.

In order to deal with collisions with external objects, the library was extended with an interface for adding arbitrary bodies to the model at run-time. Furthermore, an approach for quickly converting clustered 3D point clouds into KCCD body representations was developed. This process works as follows:

1. Add a KCCD supporting point for every point in the point cloud
2. Iteratively remove each supporting point from the KCCD volume and check if the volume still encloses the complete point cloud
3. If the number of points $n$ is bigger than the desired maximum number of points, increase the buffer radius $r$ by a fixed step and go to step 2. Else, terminate.

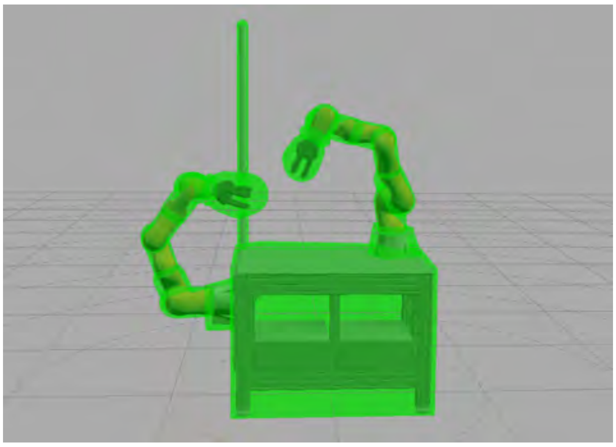

**Fig. 3** Self-collision avoidance model of the dual arm system.

Given the distance computations between the robot and external objects, avoidance motions are computed using virtual repulsive potential fields [5]. The repulsive vector is spanned between the two closest points on obstacle and manipulator, respectively, pointing away from the obstacle. To execute avoidance motions, our whole-body control framework was used. Each relevant collision body of the manipulator is assigned an avoidance controller and a corresponding constraint with a certain priority. This way, the avoidance behaviours can be integrated with the actual task goals of the robot.

## 5 Initial Evaluation

In order to underline the practical relevance as well as the future potential of this demonstrator, a representative handling and assembly scenario from a gearbox manufacturing plant was chosen [1]. Here, the gear shaft and the respective coupling need to be assembled together manually with a tight tolerance, which is a tiring job since the parts are heavy, the surfaces easily scratched, and the gear wheels are shock-sensitive. At any time during the process, the system can be paused by the operator by executing a *pause-gesture*, as shown in Fig. 4(e). When paused, the robots are switched to compliant mode and can be safely moved by the operator (Fig. 4(f)). When receiving the *forward-gesture*, the assembly is resumed (Fig. 4(g)). The arm located on top of the table is responsible for most of the assembly steps and for interaction with the worker. In this specific scenario, the arm grasps the gear shafts from the container mounted at the right side of the table and puts these onto the bolts on the left side of the table using force feedback (shown in Fig. 4(h,i)). The arm mounted on the side of the table uses a pinch movement to grasp a coupling from the storage located on the floor (Fig. 4(j)). The coupling is then moved to a transfer position above the table where the workspaces of both robots overlap. Then the coupling is exchanged between the two arms (shown in Fig. 4(k,l)) and the top-mounted arm executes the final assembly step by inserting the coupling into the gear shaft (Fig. 4(m)). This final step is executed using force control because the fit between both parts has low tolerances. During the whole operation, the robots' speed is adapted based on the distance to the operator. The current working mode is indicated by LEDs on the front of the system and on a monitoring screen (Fig. 4(n,o)). By executing the *collaboration-gesture*, shown in Fig. 4(p), the worker can request the interaction mode. In this mode, the robot grasps the last assembled piece and suspends it in a comfortable position for the operator to inspect it (Fig. 4(q,r)).

The demonstrator is currently in use and will be further developed in the joint research project 'Hybr-iT' funded by BMBF (German Federal Ministry of Research and Education) which is coordinated by DFKI in the area of Industry 4.0 and human-robot collaboration of hybrid teams of robots, humans, and software-based assistance systems. The project includes Volkswagen Group and Airbus Operations GmbH as industrial end-users.

**Acknowledgements** This project has been partially funded by Volkswagen AG (Smart Production Lab).

## References

1. Franka robot from KBee AG . `https://www.franka.de/` (2016). [Online; accessed 16-August-2016]
2. KUKA LBR iiwa. `www.kuka-lbr-iiwa.com` (2016). [Online; accessed 16-August-2016]
3. Rock, the Robot Construction Kit. `http://rock-robotics.org/stable/` (2016). [Online; accessed 16-August-2016]
4. Universal Robots. `www.universal-robots.com` (2016). [Online; accessed 16-August-2016]
5. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. In: Robotics and Automation. Proceedings. 1985 IEEE International Conference on, vol. 2, pp. 500–505 (1985)
6. Taeubig, H., Baeuml, B., Frese, U.: Real-time swept volume and distance computation for self collision detection. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, CA, USA (2011)
7. Wirkus, M.: Towards Robot-independent Manipulation Behavior Description. In: Proceedings of the 5th International Workshop on Domain-Specific Languages and models for ROBotic systems. arxiv.org, Bergamo, Italy (2014)

---

[1] see complete video under `https://youtu.be/VoU3NbTyFtU`

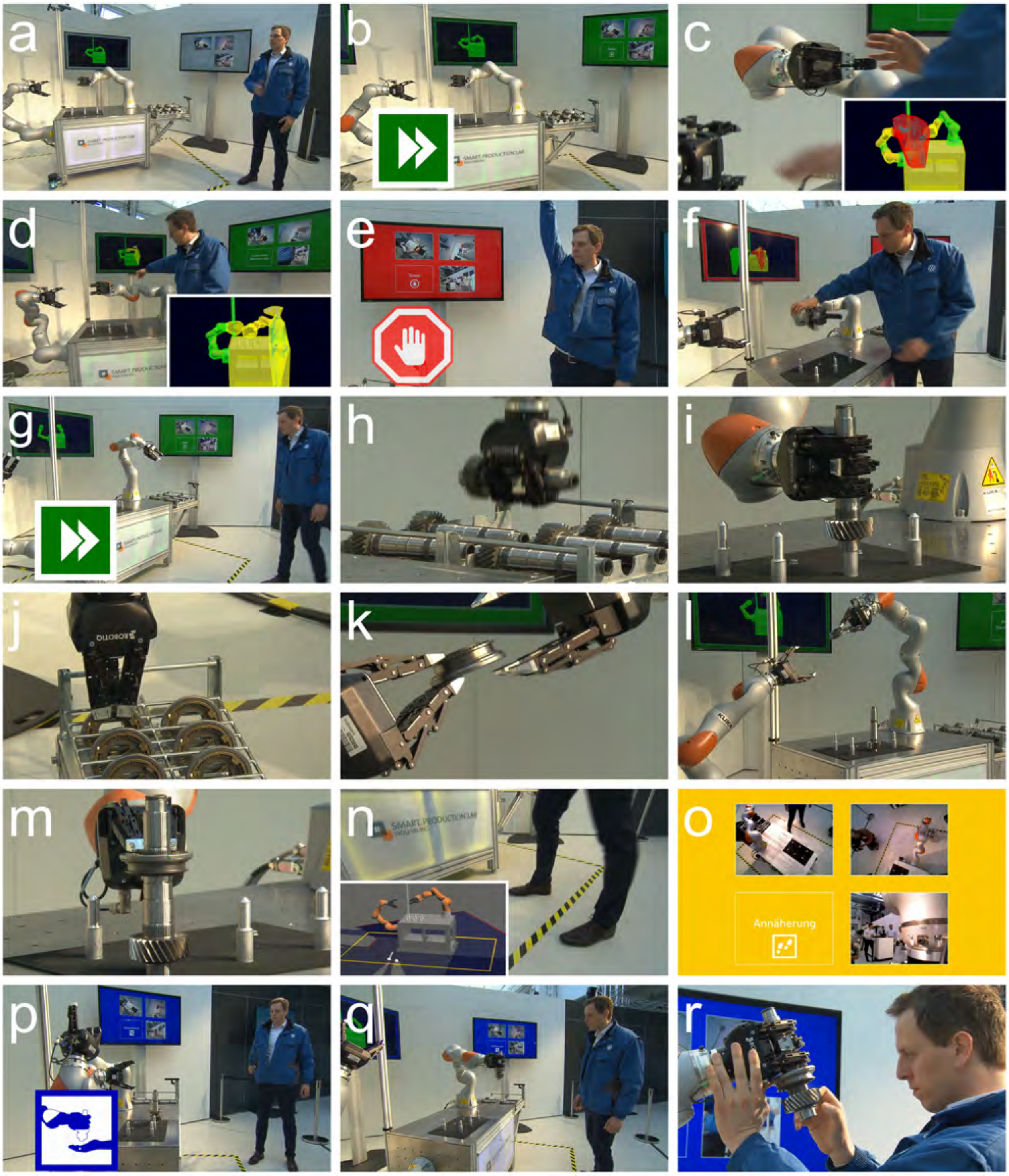

**Fig. 4** Timeline of the demonstration at a mockup of a gearbox assembly station at a Volkswagen factory. It shows the self-collision avoidance (a–d), pause mode (e,f), the assembly process (g–m), the adaption of the working mode by workspace monitoring (n,o), and the interaction mode (p–r).