



HUMAN-ROBOT INTERACTION

Intrinsic sense of touch for intuitive physical human-robot interaction

Maged Iskandar^{1,*}, Alin Albu-Schäffer^{1,2}, Alexander Dietrich¹

The sense of touch is a property that allows humans to interact delicately with their physical environment. This article reports on a technological advancement in intuitive human-robot interaction that enables an intrinsic robotic sense of touch without the use of artificial skin or tactile instrumentation. On the basis of high-resolution joint-force-torque sensing in a redundant arrangement, we were able to let the robot sensitively feel the surrounding environment and accurately localize touch trajectories in space and time that were applied on its surface by a human. Through an intertwined combination of manifold learning techniques and artificial neural networks, the robot identified and interpreted those touch trajectories as machine-readable letters, symbols, or numbers. This opens up unexplored opportunities in terms of intuitive and flexible interaction between human and robot. Furthermore, we showed that our concept of so-called virtual buttons can be used to straightforwardly implement a tactile communication link, including switches and slider bars, which are complementary to speech, hardware buttons, and control panels. These interaction elements could be freely placed, moved, and configured in arbitrary locations on the robot structure. The intrinsic sense of touch we proposed in this work can serve as the basis for an advanced category of physical human-robot interaction that has not been possible yet, enabling a shift from conventional modalities toward adaptability, flexibility, and intuitive handling.

Copyright © 2024, the
Authors, some rights
reserved; exclusive
licensee American
Association for the
Advancement of
Science. No claim to
original U.S.
Government Works

INTRODUCTION

The steadily increasing capabilities of modern robotic systems will make them excellent collaborators in various scenarios in the future, ranging from manufacturing setups (1, 2), space cooperation tasks (3), and health care and medical applications (4) to daily life assistance (5). Thus, combining and fusing human problem solving, reasoning, and perception with the precision and speed of robots is a highly active field of research. In the literature, one can find numerous modalities in the context of human-robot interaction (HRI) that serve as interfaces (6–9), including vision-based approaches (10–13), voice-recognition methods (14, 15), or physical-contact interaction techniques (16, 17). However, the question of how to achieve intuitive physical HRI is challenging and still demands substantial advancements. Enabling a sense of touch is one step in the right direction so that robots can efficiently interact with humans and their environment in a safe way. Moreover, a wide variety of applications will benefit from this advanced feature, including the accurate identification of physical contacts in terms of location, force intensity, and direction and the use of this information as an additional communication dimension.

In the classical approaches for physical HRI, force-torque and joint-torque sensors are used for control purposes (compliance, vibration suppression, and collision reaction), whereas precise tactile information, such as the location and intensity of contact forces at any point on the robot, requires explicit tactile sensing. The latter is not yet an industrial standard because of several limitations to be mentioned below.

Explicit tactile sensors and artificial skins placed on the surface of the robot provide effective means for detecting physical interactions (18, 19). However, the forming procedure with respect to

the curved surfaces of a robot can be challenging. Despite various sensors reaching a high level of maturity, coverage of the entire robot, the corresponding wiring, communication, and power supply remain points of considerable complexity. Moreover, depending on the technology, it can be hard to ensure sufficient robustness against impacts and in the case of collisions. In (20), touch modules that form a grid to cover specific parts of the robotic structure were developed. Although the maximum resolution for the identification depends on the size of the elements (21), the immense potential of such an approach lies in the full coverage of the entire system, such as in humanoid robots (22). Conversely, using soft materials can enable partial robot coverage through modular patches, albeit with a limited range of applied force. In this context, a bioinspired tactile skin was empowered by deep learning algorithms to decode contact forces (23), and the authors of (24) used room-temperature liquid metals attachable to human skin with the ability to map the signals to the robotic system as a result of sensor deformation. The robotic skin developed in (25) with mechanically gated electron channels achieves sliding tactile perception with high sensitivity and real-time feedback, which can be used as artificial fingerprints for grasping. A soft magnetic tactile sensor is presented in (26), consisting of a flexible magnetic film that deforms in response to external forces detected by a Hall sensor through changes in magnetic flux densities, with the property of force self-decoupling. Furthermore, a biomimetic robotic skin based on hydrogel-elastomer hybrids and tomographic imaging, designed as a soft, resilient whole-body tactile sensor with reparability, is detailed in (27). In the fabrication of displays, normal-direction force measurements can serve as an alternative for the standard technologies integrated in touchscreens or to enhance the pressure sensitivity (28, 29).

The identification and isolation of external interactions via integrated joint-torque sensors in combination with model-based dynamics observers can be regarded as one of the standard approaches in modern torque-controlled lightweight robots (30). Having knowledge about the contact forces supports active control, for example, to

¹Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Münchener Str. 20, 82234 Wessling, Germany. ²Technical University of Munich (TUM), 80333 München, Germany.

*Corresponding author. Email: maged.iskandar@dlr.de

maintain contact during grasping, apply a specific time-dependent force profile (31), or optimize the contact in terms of permissible forces as a collision reaction strategy (32). The estimation capability can be further improved by integrating additional instrumentation in various places in the kinematic chain of the robot, which is usually realized by means of a distal, six-dimensional (6D) force-torque sensor close to the end effector. We refer to such a setup with more available force-torque measurements than the number of actuated degrees of freedom as “sensing redundancy” in the following. However, in the literature, the direct use of such additional sensor signals is constrained because of the need to correctly compensate for dynamic effects to obtain the actual quantities that describe the external interaction. In practice, this implies the measurement or estimation of accelerations, which is known to be problematic in terms of accuracy, sensor noise, or availability in general. Besides model-based methods, the localization of contacts on the basis of the interaction force information can be achieved using computer vision-based approaches, machine learning, or optimization techniques (33–36). Yet, some limitations and challenges still exist, including aspects of real-time capability, the restriction to nonsingular configurations, or the limitation to an area near the end effector. Although this classical setup restricts the estimation of physical interactions primarily because of the configuration-dependent mapping between applied external forces and measurable joint torques, we have recently shown that the introduction of considerable sensing redundancy can overcome all of these issues and yield highly accurate estimates (37). Apart from that, the approach enables the detection and isolation of multiple contacts simultaneously and in real time, even in singular configurations.

In the context of human-robot collaboration, it is desirable to have robots and operators working within a shared workspace, where robot and human perform tasks concurrently or even jointly (38). However, there is always the question of how to physically communicate the human intentions to the robotic system. From the HRI point of view, different sensory approaches with different perception goals are used to detect and classify the human interaction. Among common interaction methods, one finds proximity-based sensing for halting conditions (39) and force-based sensing for initiating tasks (40). Naturally, the general interest goes beyond the sheer identification of human contacts as binary signals to include the much richer possibilities of physical interactions.

Here, we targeted this latter category and used integrated sensors to equip a robot with intrinsic tactile capabilities that enabled us to exploit the rich nature of sensitive interactions (Movie 1). We shifted the scope of contact forces applied to a robotics structure beyond their conventional use and presented an intuitive method for physical human-robot communication and interaction by simply touching the surface of the robot (see Fig. 1). In contrast with the use of explicit tactile instrumentation, the use of internal sensors has the advantage that the range of permissible and manageable interaction forces is considerably larger, particularly in the high-load region. Because of the effective combination of model- and machine learning-based algorithms, there is no need for any explicit tactile sensors in our approach. We introduce a unified sensing concept for control, safety, and tactile interaction that involves the entire body of the robot, and we present the entire pipeline required to achieve this breakthrough. Multiple contacts on the structure can be detected simultaneously and in real time, without limitations such as the restriction to nonsingular configurations or regions close to the end



Movie 1. Overview of the intrinsic robotic sense of touch method.

effector. By means of the introduced intrinsic sense of touch, one can write or even draw on the surface, and the robot is able to automatically identify and interpret these abstract commands. We established this intuitive communication link by applying both manifold learning techniques and rotation-invariant convolutional neural networks (RICNNs) in an intertwined way. Additionally, we introduced the concept of virtual buttons, which can be flexibly placed anywhere on the robot surface and assigned any desired functions such as slider bars or switches. Our proposed technology can serve as the basis for an unexplored class of interaction modalities between humans and robots in the future.

RESULTS

The presented approach to enable the sense of touch was achieved through the effective intertwining of various technologies. On the hardware side, the unique mechatronic design of the SARA robot provided the sensing redundancy, whereas the precise multitouch point localization introduced in the following was grounded in the recently developed momentum-based monitoring method (37) that directly exploits this hardware feature.

In this section, we summarize our main results and contributions, starting with the identification and extraction of time-dependent trajectories on the surface of the robotic structure, which describe the physical interaction in terms of location and intensity. Afterward, we geometrically unfold these trajectories through manifold learning techniques to obtain a flat representation, which can be fed to a deep learning-based algorithm for recognition in a follow-up step. In this way, we close the gap between the analog physical HRI and its digital use to command the robot, for example, through written machine-readable code [American Standard Code for Information Interchange (ASCII) code] or the activation of buttons on the surface of the robot structure.

Hardware overview

Here, the German Aerospace Center (DLR) SARA system, short for Safe Autonomous Robotic Assistant, was used (see Fig. 1). It is the successor of the DLR lightweight robot (41), a worldwide well-known force-sensitive arm, which has been established in various research and industry applications. The actuators in SARA are equipped with both motor- and link-side position sensors. These high-resolution magnetoresistive-type encoders measure the joint positions before and after the gear transmission, and they are essential for the low-level joint control to properly handle the intrinsic elasticity in such lightweight robots (42). All force-torque sensors

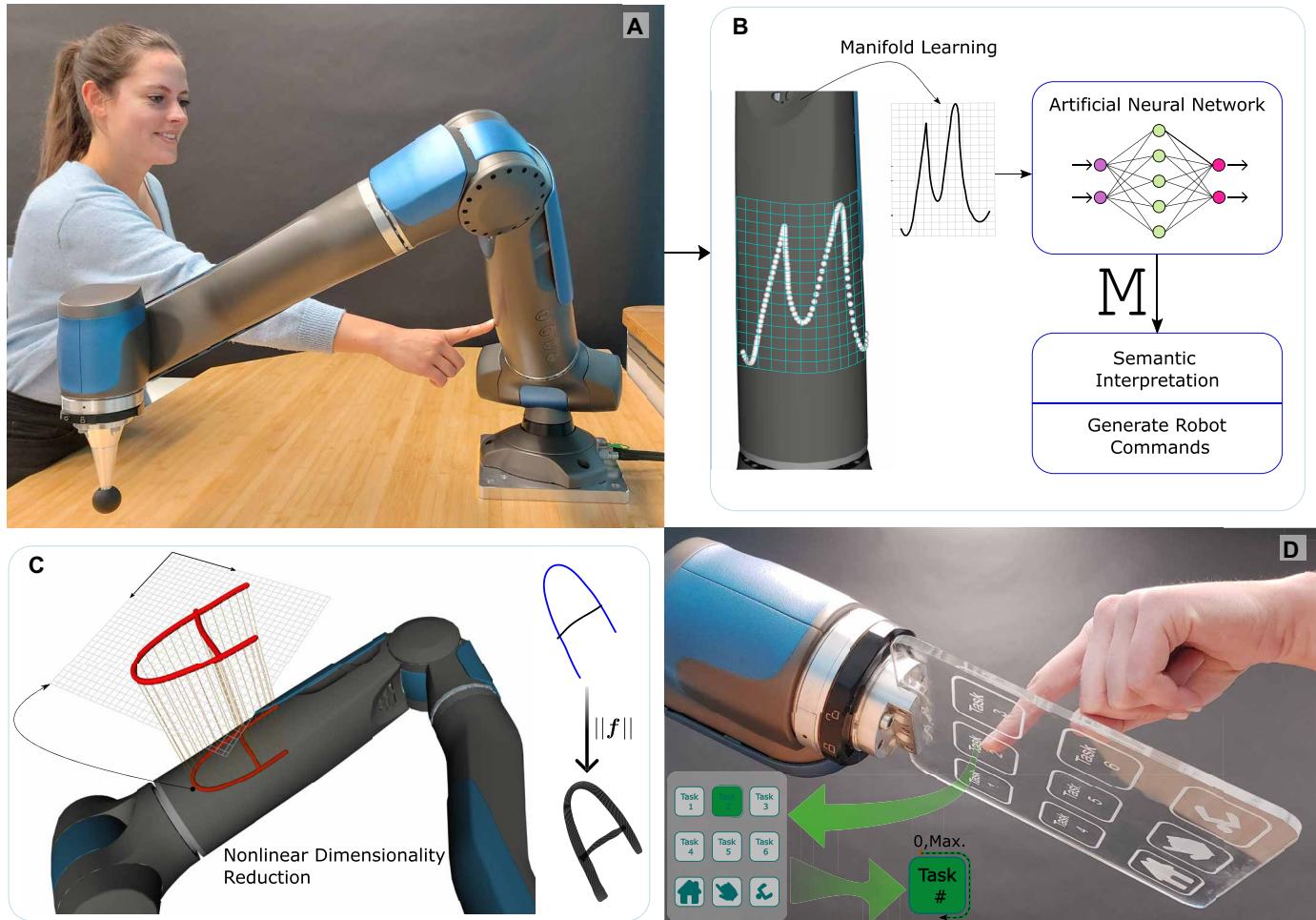


Fig. 1. The intrinsic sense of touch. Sensitive writing or drawing on the structure (**A**) is automatically interpreted using convolutional neural networks (**B**). (**C**) The accurate reconstruction of the physical interaction is achieved through nonlinear dimensionality reduction via machine learning techniques. (**D**) The provided intrinsic sense of touch offers various interaction modalities without the necessity of any explicit tactile sensors.

are fully integrated within the mechanical structure for safety and control purposes. Being of a strain-gauge-based type with optimized design, they feature high sensitivity and resolution. Two 6D force-torque sensors are deployed; one is integrated in the base and another in the wrist of the manipulator. In addition, the system has torque sensors in each of the revolute joints along the articulated kinematic chain, four in total. Consequently, with these 16 measurements, the robot features sensing redundancy with respect to the seven actuated degrees of freedom. All signals and sensory data were provided at a high rate of 8 kHz with a 16-bit resolution, enabling the robot to realize Cartesian contact stiffness values of up to approximately 20,000 N/m at the end effector.

Touch trajectory

In the following, we refer to the combination of applied contact forces and the corresponding time sequence of contact points as the “touch trajectory.” Touches could be applied to arbitrary places on the robot body and irrespective of the joint configuration. This feature was enabled by the underlying fusion of multiple sensor signals by means of the dynamically decoupled momentum-based monitoring approach. In this way, we established an intrinsic tactile

property associated with the robot surfaces, thereby unlocking an unexplored sense of touch. Figure 2A depicts the example of drawing a digit “8” on the curved surface of the robot structure. Because such touch trajectories are naturally subjected to noise, an intermediate smoothing step was introduced. After filtering, a smooth touch trajectory was generated as shown in Fig. 2A, visualizing the result on the robot structure and in 3D space with the corresponding force vectors. Optionally, one could use an additional smoothing layer to obtain a piecewise approximation such as represented through splines or Bézier curves.

Unfolding with manifold learning

To interpret the touch trajectories, we obtained a lower-dimensional representation with manifold learning techniques for nonlinear dimensionality reduction. The goal was to eliminate the effect of the curvature on the links of the robot and provide an independent, flat representation of the extracted trajectory. Originally, the applied trajectory was a path in 3D Euclidean space, which was located on and constrained to the surface of the robot structure. This resulted in a 2D manifold embedded in 3D space, as depicted in Fig. 2B, because the dimension could be reduced without any loss of information.

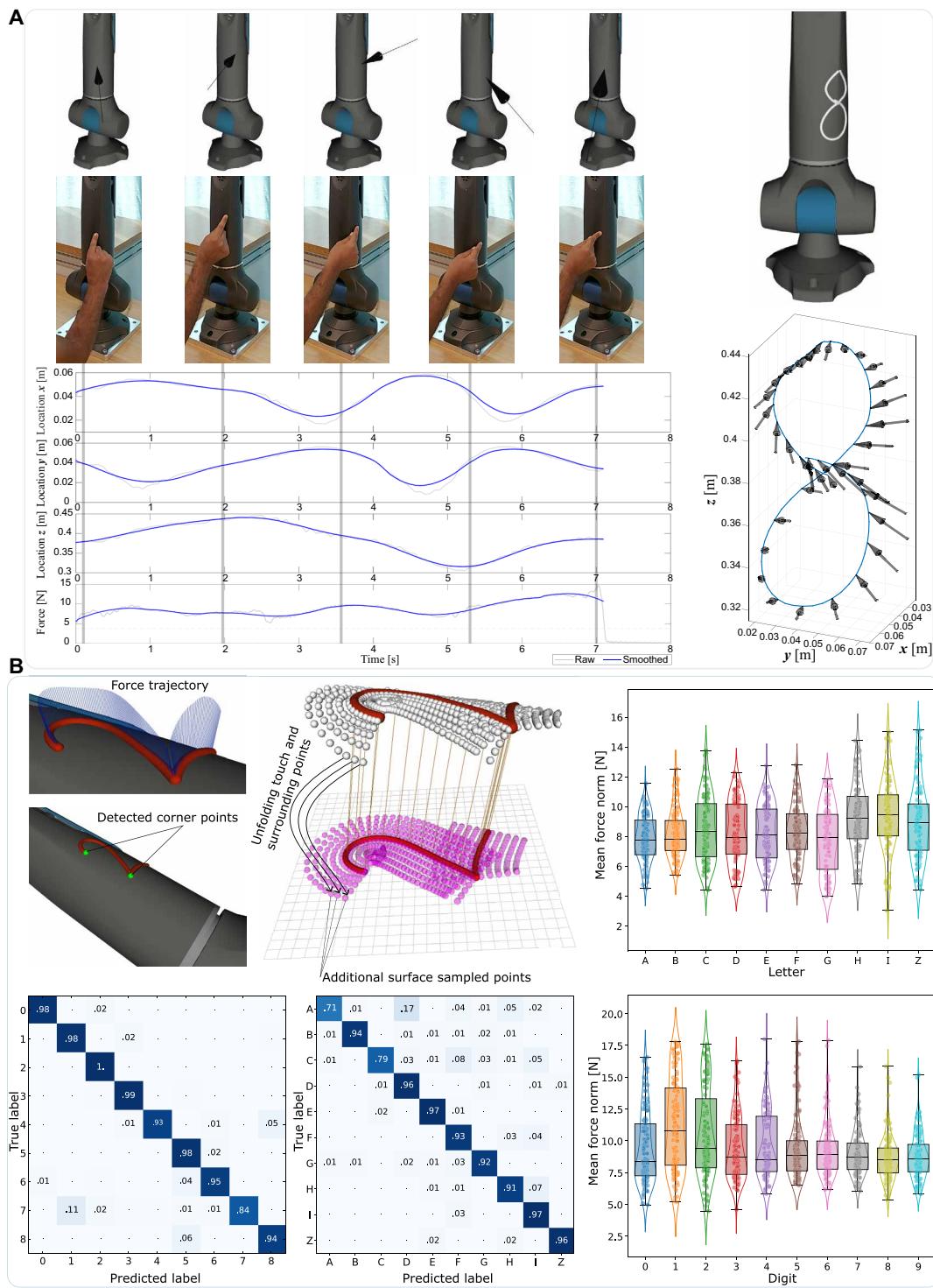


Fig. 2. External forces and contact points (touch trajectory). (A) Extracting the touch trajectory by continuously tracking the contact point and the intensity of interaction. (B) Unfolding of the touch trajectory applied on the robotic structure and its subsequent interpretation. To obtain a flat representation of the original touch trajectory, a nonlinear dimensionality reduction was performed using manifold learning techniques. In addition, the performance of the neural network-based interpretation of digits and letters is illustrated using the touch trajectories from 20 persons writing on the robot surface. Recognition accuracies of 95.5% and 90.4% for the digits “0” to “8” and the characters “A,” “B,” “C,” “D,” “E,” “F,” “G,” “H,” “I,” and “Z,” respectively, were achieved. In addition, the mean writing forces and their distributions are shown over the collected data of digits and letters. Here, the violin plot highlights the force distribution, the black box includes the second and third quartile, and the horizontal line goes through the median.

The 2D equivalent of the touch trajectory was approximated via Iso-map (43), which yielded a suitable format to be processed by deep learning-based algorithms for handwriting recognition in the subsequent step. The obtained local manifold included both the touch trajectory and adjacent, extra points to accurately unfold the local area around the points of interest. The relations between the original points on the curved surface and the unfolded planar representation are highlighted in Fig. 2B. The situation after this unfolding step was similar to the writing on a modern touchscreen device. If a simple plane projection was used instead of proper unfolding, the distortion due to the curvature would notably affect the result, because the original shape of the touch trajectory would not be adequately preserved.

Touch recognition with group-equivariant deep learning

Extracting the textual interpretation of the applied touch trajectories could be exploited for intuitive physical interaction. Among others, the textual information could be used to program the robot, trigger the execution of commands, or parameterize specific tasks. We achieved the trajectory classification through the incorporation of the rotation-invariance property of the network design using group convolutional neural networks. Flat trajectories were obtained with random orientations resulting from 3D original touch trajectories because of different robot configurations and writing perspectives of the user. The model was trained using upright handwritten data, whereas the evaluation was done on arbitrarily oriented touch data. The processing and analysis of the touch trajectory started as soon as a contact force threshold was exceeded and ended if no more contact points were registered. An additional time threshold was added to trigger the recognition and separate individual touches. As a consequence, the interpretation of symbols (characters, signs, and numbers) was achieved, for example, as standard ASCII code to be subsequently processed by a higher-level artificial intelligence (AI) instance. In Fig. 2B, statistical results for the predicted labels of the neural network are shown for the digits “0” to “8” and the characters “A, B, C, D, E, F, G, H, I, Z,” with prediction accuracies of 95.5% and 90.4%, respectively. Additionally, a statistical analysis of the collected data with respect to the writing forces was reported. Although the training of the neural network was based on a publicly available handwritten digits and letters dataset [EMNIST (Extended Modified National Institute of Standards and Technology)], the statistical evaluation on the SARA system involved 20 persons writing on the robot structure with a total of 2300 samples. Note that the digit “9” was deliberately omitted here because of the ambiguity with regard to “6.” Figures S1 and S2 include the additional evaluation of all digits “0” to “9,” and fig. S3 illustrates a selected sample of handwritten letters. A sequence of applied touch trajectories and their corresponding recognition are illustrated in movie S1. The user in Fig. 3A used the automatic interpretation of written digits to command the robot to approach different Cartesian end-effector positions that directly related to the numbers “1” and “3.” Commanding the robot to achieve various goals through touch trajectories with different applied symbols is demonstrated in movie S2.

Virtual buttons and applications

Besides AI-driven touch recognition and interpretation, we introduced the concept of customized virtual buttons, which represent an interaction modality to enable the digital activation of tasks or realize specific functionalities. The buttons could be custom-designed to

take different shapes, and they could be arbitrarily assigned along the robot structure (see movie S3) to accommodate both discrete and continuous touches depending on the desired output quantity. Various control modalities could be performed (see fig. S4). In the application illustrated in Fig. 3B, pressing the button directly led to a (pre)programmed motion at the end effector. Alternatively, interacting by continuously pushing a button could be used to trigger time-dependent higher-level commands. Figure 4A illustrates this case. After assigning the virtual button to a specific area, the functionality of continuously moving the end effector upward and downward was enabled as illustrated in movie S4. Figure 4 (B and C) demonstrates the activation procedures for discrete and continuous buttons, respectively. As soon as a force threshold was exceeded, the corresponding command was executed. In the first case, the four functionalities F1 to F4 were triggered, whereas the latter scenario related to the activation of the continuous translational motion in the Cartesian space of the end effector. The button shapes and locations could be taught through human demonstration by selecting the feature points of the button shapes as exemplified in movie S5. A template could be used for common shapes, and their parameters could be specified by the user’s touch as well (for more details, see fig. S5).

Optionally, additional tools could be physically attached to the robot to actively write on. For instance, a flat plate could be mounted at the end effector for this specific purpose, as illustrated in Fig. 1D. This option is advantageous if the robot features automatic tool-changing capabilities such that the additional tool can be automatically requested when needed as an extended list of functions or auxiliary flat area to handwrite commands. The SARA robot provides such a feature so that the user can assign a virtual button to request the additional functions. In principle, this reduces the programming effort and offers enormous flexibility, particularly during human-robot cooperation scenarios. The tool could be used to define a sequential process that contains several defined functions or to choose a complete manufacturing routine (see movie S6 and fig. S6).

Given that slider bars have been established as practical tools in various contexts of our everyday life such as in smartphones or tablet computers, their adoption for intuitive robot programming seems natural. The sense-of-touch feature straightforwardly enabled this modality as illustrated in Fig. 5. Potential use cases involved the variable, stepless tuning of the controlled Cartesian stiffness at the end effector for setups such as in movie S7, the adjustment of thresholds to define safety boundaries, or more trivial instances such as simple volume control. We envision a flexible way to adapt and reconfigure the whole interaction setup, for example, through flexible placement (Fig. 5A), continuous activation (Fig. 5, B to D), and intuitive drag-and-drop actions (Fig. 5E).

DISCUSSION

In contrast with the conventional ways of physical HRI, the exploitation of sensing redundancy enabled us to provide an intuitive and handy communication link between human and robot that offers substantial advancements compared with the state of the art. Because of the use of the integrated force-torque sensors instead of explicit tactile instrumentation located on the robot surface, the total range of interaction forces was considerably extended. This result comes with the advantage of using only one type of sensor without the need for artificial skins, which are usually expensive and feature

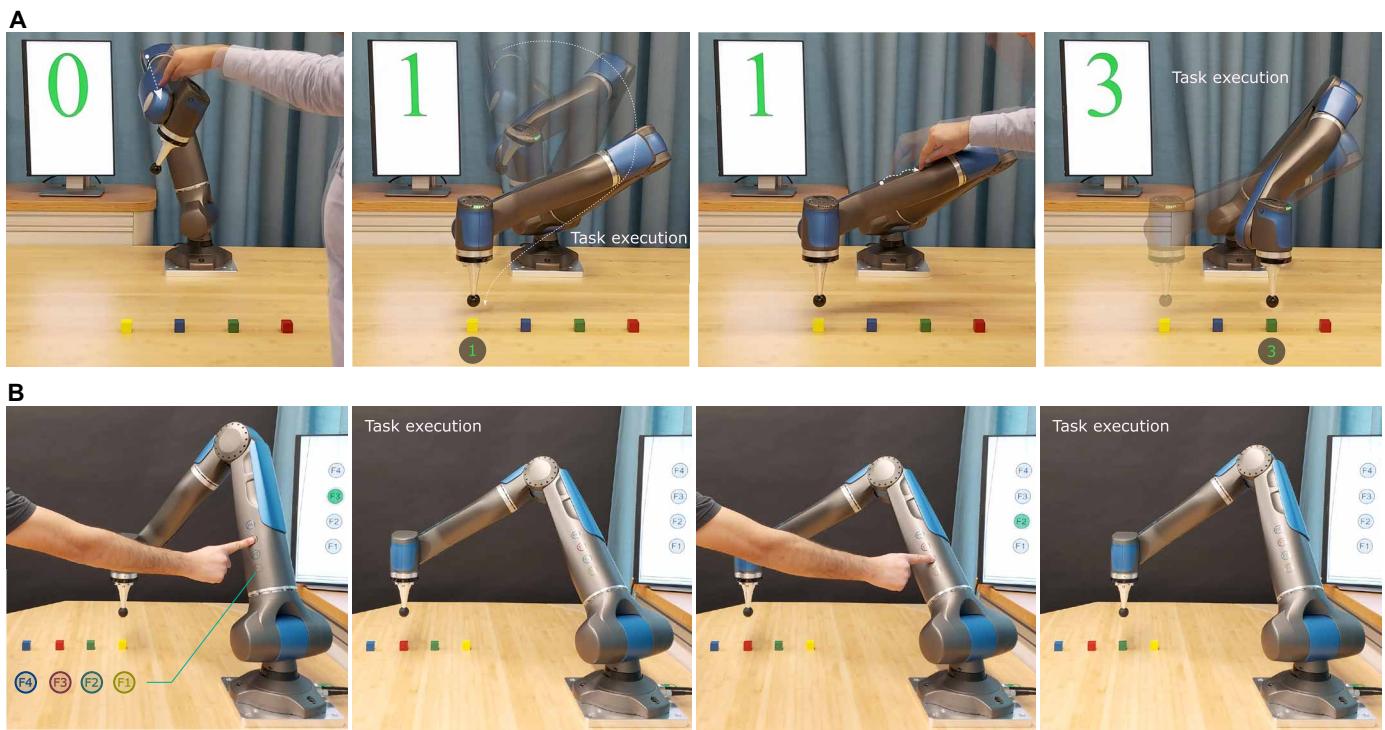


Fig. 3. Touch recognition. (A) The interpretation of written digits on the robot surface as machine-readable code is used to intuitively command the robot. The touch trajectory for writing digit “1” is applied, the trajectory is successfully recognized, and the assigned task is executed accordingly. Similarly, applying digit “3” triggers the execution of the corresponding task. (B) Likewise, virtual functional buttons can be placed anywhere on the structure to assign high-level tasks.

a high level of complexity when used to cover the entire robot. The permissible intensity was only limited by the characteristics of the proprioceptive sensors themselves, which were chosen in the design phase to naturally handle the entire operational range of the robot. Moreover, all locations on the robotic structure can be used with this approach.

By means of multilayer learning techniques, we were able to lift the intrinsic sense of touch to high levels of interaction modalities because even abstract communication such as through written text was automatically interpreted and processed in real time. One has to note that the training data currently cover digits and letters only. Yet, the extension to new symbols is actually straightforward and describes the next evolutionary step to improve the performance of our approach.

Although some of the current experimental validations cover static situations for the sake of transparency, our proposed HRI communication link can also be established and active during highly dynamic motions of the robot as can be seen in movie S8. This feature is because of the precise estimation of external interactions using an internal dynamic model (37) in combination with the sensing redundancy that, in turn, further improves the overall accuracy.

The main limitations of the proposed approach relate to the maximum number of simultaneous points to be detected with the current amount of sensing redundancy. Currently, only two contacts can be properly detected. This is sufficient for most interaction scenarios, whereas the integration of more force-torque sensors in the kinematic chain of the robot would increase this number. Furthermore, the calibration effort of the sensors is time consuming because the performance of the contact detection depends on the accuracy of each sensor. Last, it has to be noted that, depending on the location of the

physical interaction and the current joint configuration, the resolution is about 0.1 N, but it might be less accurate than tactile skins in the low-force region, for example, for measuring very light touches.

In terms of future work, we believe that the fusion of tactile instrumentation with the proposed method will synergistically combine the benefits of two physically different approaches, which might be highly efficient in terms of absolute accuracy, robustness, and reliability. Moreover, additional tactile instrumentation will endow the system with the capability to detect areal touches, which is advantageous in applications such as tactile exploration. Such a combination of technologies would further increase the amount of sensing redundancy in the system, naturally facilitating more simultaneous contact points to be detected in parallel.

MATERIALS AND METHODS

In the following, the touch recognition procedure is outlined on the basis of the scheme in Fig. 6. Using the monitoring of the physical interaction, the multipoint trajectory can be concluded. The dimensionality reduction in the subsequent steps provides the basis for the classification of this trajectory and the recognition and textual interpretation of potential characters, digits, or symbols. Last, virtual buttons as efficient instances of the application-driven use of this approach are described, followed by their deployment in the task execution on the robotic hardware.

Momentum-based interaction monitoring

External forces and torques can be estimated by means of the generalized momentum (30) while avoiding the problematic inversion

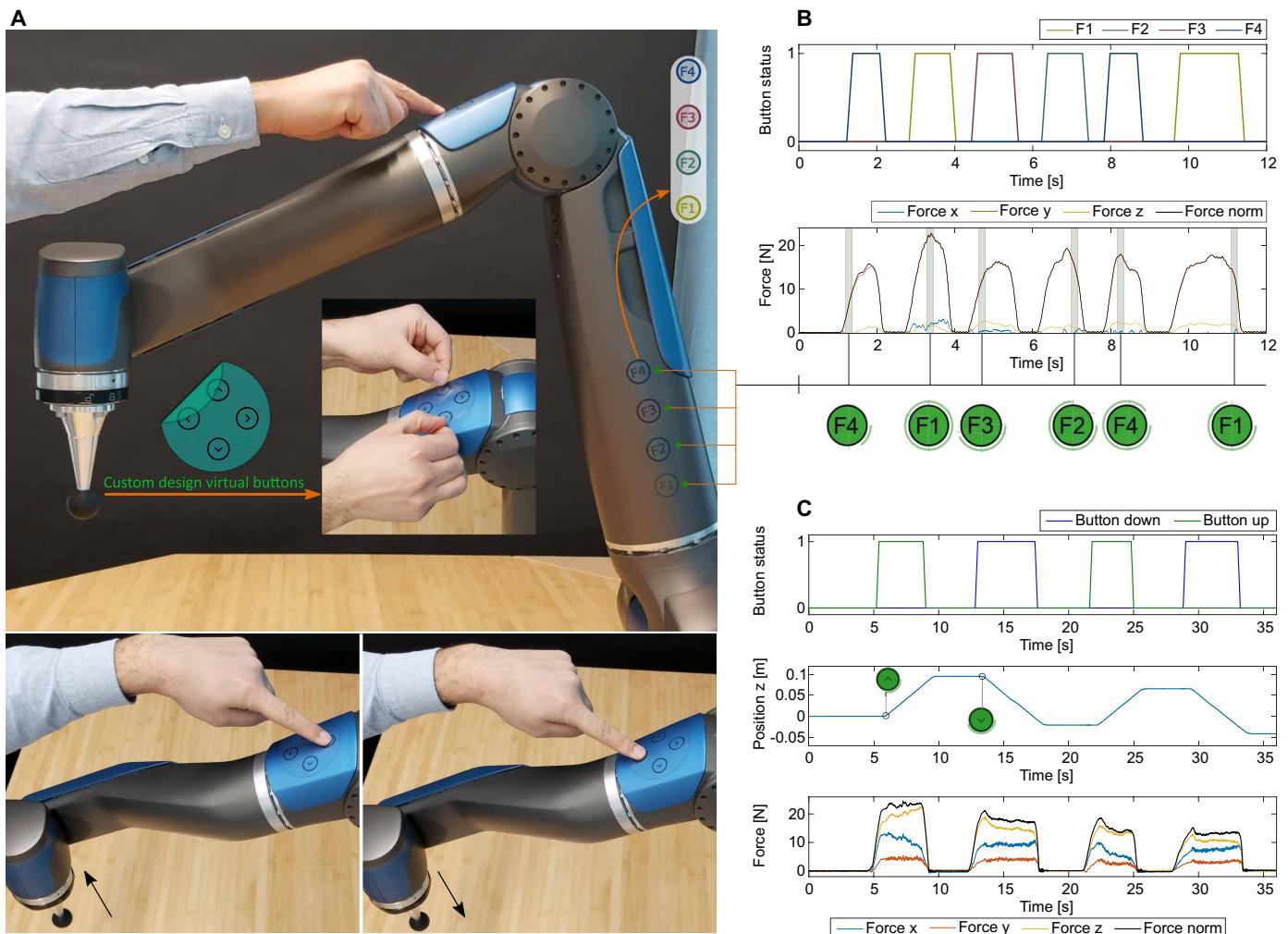


Fig. 4. Concept of virtual functional buttons. The physical placement of such buttons can be freely specified on the robot surface, and each button can be assigned to a specific function or task as shown in (A). The set of buttons in (B) contains numbered functions (F1 to F4), which are triggered randomly. In (C), a continuous, single-direction motion of the end effector is commanded by pressing the corresponding directional arrow button.

of the inertia matrix and eliminating the need for measured or estimated joint accelerations. Conventionally, such feedback on acceleration is required, even in the presence of force-torque sensors. Here, the sensing redundancy was exploited in the dynamics formulation through the introduction of additional virtual joints that are kinematically locked by definition. This strategy resulted in extended equations of motion subject to external contacts:

$$\bar{M}(\bar{q})\ddot{\bar{q}} + \bar{C}(\bar{q}, \dot{\bar{q}})\dot{\bar{q}} + \bar{g}(\bar{q}) = \bar{\tau} + \bar{\tau}^{\text{ext}} + A(\bar{q})^T \lambda \quad (1)$$

where $A(\bar{q})\dot{\bar{q}} = \mathbf{0}$, with the generalized extended coordinates $\bar{q} \in \mathbb{R}^{\bar{n}}$ expressed as

$$\bar{q} = \begin{bmatrix} q \\ q_b \\ q_{ui} \\ q_{ee} \end{bmatrix} \quad (2)$$

which contains the original joint coordinates $q \in \mathbb{R}^n$, the virtual coordinates $q_b \in \mathbb{R}^{n_b}$ of a base sensor, $q_{ui} \in \mathbb{R}^{n_{ui}}$ of a user-interface sensor, and $q_{ee} \in \mathbb{R}^{n_{ee}}$ of an end-effector sensor. Herein, $\bar{n} = n + k$ is the number of extended coordinates in consideration of the physical presence of the additional sensors, where n is the number of actuated joints with the respective positions q and $k = n_b + n_{ui} + n_{ee}$ equals the number of additional redundant sensor signals that result in the kinematically locked joints in Eq. 1. The symmetric and positive definite inertia matrix is $\bar{M}(\bar{q}) \in \mathbb{R}^{\bar{n} \times \bar{n}}$, the Coriolis and centrifugal matrix is $\bar{C}(\bar{q}, \dot{\bar{q}}) \in \mathbb{R}^{\bar{n} \times \bar{n}}$, and the generalized gravity forces are described by $\bar{g}(\bar{q}) \in \mathbb{R}^{\bar{n}}$. Furthermore, the term $\bar{\tau}$ denotes the generalized joint forces collocated to the extended coordinate vector \bar{q} . The external wrench $F^{\text{ext}} \in \mathbb{R}^m$ with total task space dimension m is mapped to generalized external forces $\bar{\tau}^{\text{ext}} \in \mathbb{R}^{\bar{n}}$ via the geometric contact Jacobian matrix $J_c(\bar{q}) \in \mathbb{R}^{m \times \bar{n}}$ according to

$$\bar{\tau}^{\text{ext}} = J_c(\bar{q})^T F^{\text{ext}} \quad (3)$$

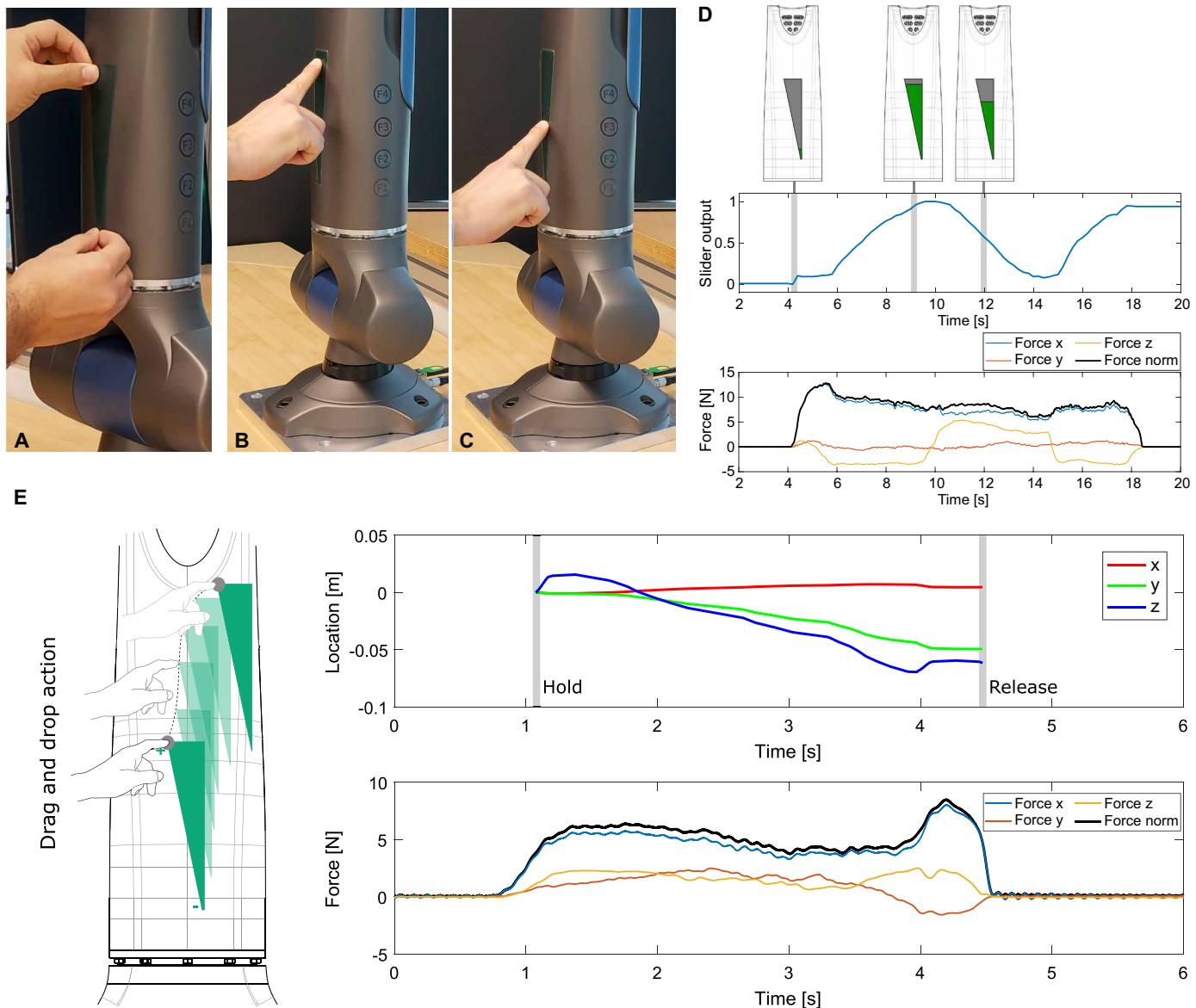


Fig. 5. Slider bar as instance of a continuous virtual button. Apart from discrete inputs such as in switches, continuous or quasi-analog modes such as the depicted slider bar can be realized with the proposed method. Such objects can be easily placed (**A**) on a predefined area along the robot structure. The user can actively change the level of the virtual slider (**B** and **C**), and its normalized output (**D**) is potentially mapped to a task parameter (for example, end-effector stiffness). Via drag and drop on the surface of the robot, the setup can be adapted in a flexible and intuitive way (**E**).

The term $\tau^{\text{ext}} \in \mathbb{R}^{\bar{n}}$ represents the extended generalized external forces due to the sensing redundancy. The constraint Jacobian matrix imposed by the velocity constraints of the locked joints is defined by $A(\bar{q}) \in \mathbb{R}^{k \times \bar{n}}$, where $\lambda \in \mathbb{R}^k$ describes the set of Lagrange multipliers. Although the redundant sensors may feature various levels of complexity in terms of the number of directions of measurements, the most common type is certainly the classical 6D force-torque sensor. Accordingly, the generalized external forces are augmented to

$$\tau^{\text{ext}} = \begin{bmatrix} \tau^{\text{ext}} \\ \tau_b^{\text{ext}} \\ \tau_{ui}^{\text{ext}} \\ \tau_{ee}^{\text{ext}} \end{bmatrix} \quad (4)$$

with the additional external wrenches defined as

$$\tau_b^{\text{ext}} = \begin{bmatrix} f_b^{\text{ext}} \\ m_b^{\text{ext}} \end{bmatrix}, \quad \tau_{ui}^{\text{ext}} = \begin{bmatrix} f_{ui}^{\text{ext}} \\ m_{ui}^{\text{ext}} \end{bmatrix}, \quad \tau_{ee}^{\text{ext}} = \begin{bmatrix} f_{ee}^{\text{ext}} \\ m_{ee}^{\text{ext}} \end{bmatrix} \quad (5)$$

Here, $f_b^{\text{ext}}, f_{ui}^{\text{ext}}, f_{ee}^{\text{ext}} \in \mathbb{R}^3$ and $m_b^{\text{ext}}, m_{ui}^{\text{ext}}, m_{ee}^{\text{ext}} \in \mathbb{R}^3$ describe the external force and moment components exerted at the base, user interface, and the end effector, respectively.

This concept is illustrated in Fig. 7, where the constrained sensorized 6D joints of the SARA robot are expanded to represent the force and torque-sensing capabilities at specific places in the articulated chain. On the basis of the concept of sensing redundancy and the proposed dynamics formulation, the extended generalized momentum of the system is given by

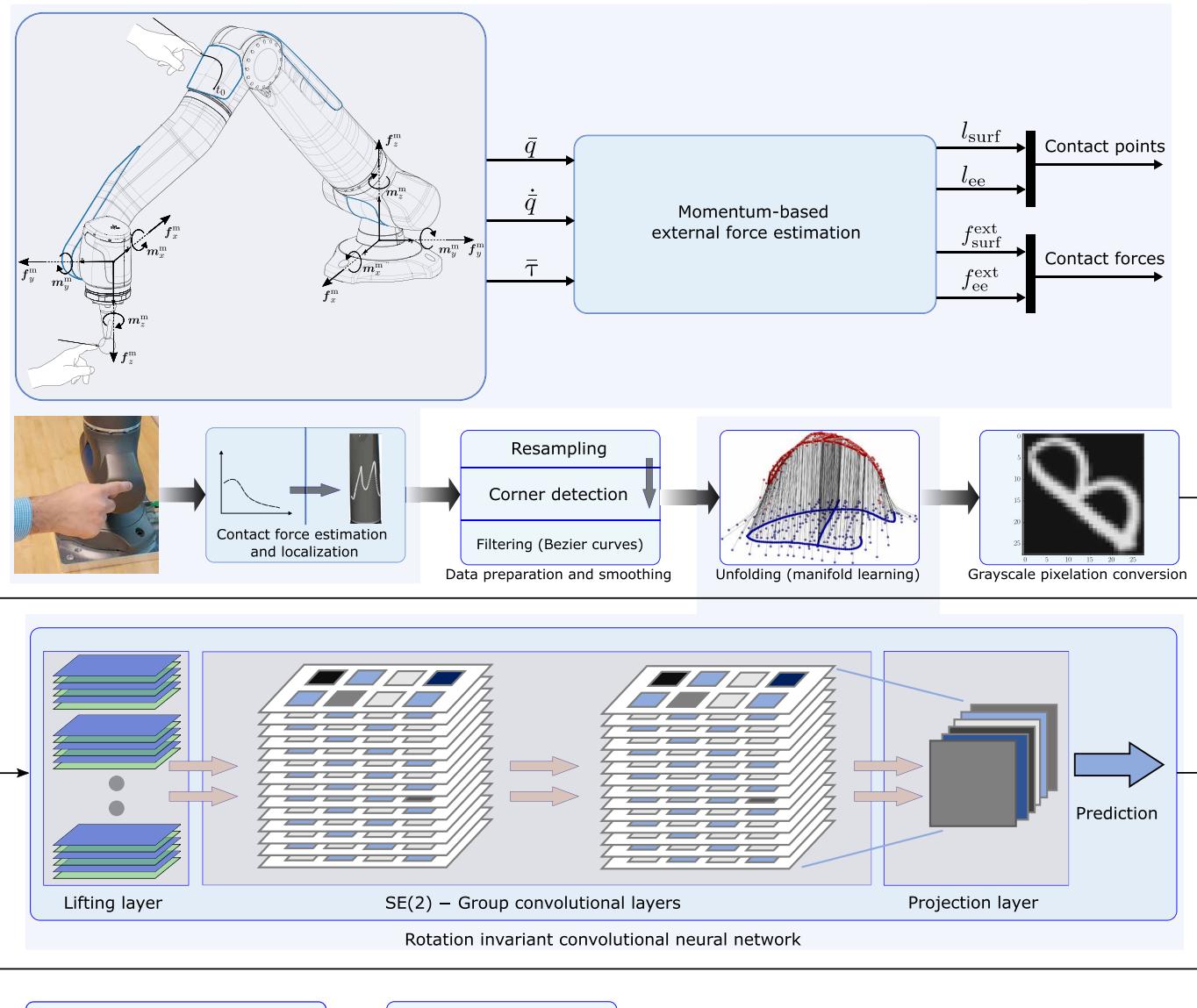


Fig. 6. Overview of the touch recognition approach. On the basis of the external force estimation, the physical interaction is detected, localized, and geometrically unfolded. The obtained pixel image is processed through a convolutional neural network, followed by the semantic interpretation of the predicted result and its use in the commanding of the robot.

$$\bar{p} = \bar{M}(\bar{q})\dot{\bar{q}} \quad (6)$$

resulting in the observer dynamics

$$\dot{\hat{\tau}}^{\text{ext}} = \bar{K}_o (\dot{\bar{p}} - \dot{\hat{\bar{p}}}) \quad (7)$$

where

$$\dot{\hat{\bar{p}}} = \bar{\tau} - \bar{n}(\bar{q}, \dot{\bar{q}}) + \dot{\hat{\tau}}^{\text{ext}}(t) \quad (8)$$

with the diagonal gain matrix $\bar{K}_o = \text{diag}(\bar{K}_{o,1}, \dots, \bar{K}_{o,n})$ and $\bar{n} = \bar{g} + \bar{C}\dot{\bar{q}} - \dot{\bar{M}}\dot{\bar{q}}$. For compactness and the sake of readability, the dependencies on the states have been omitted. Consequently, the observer output is

$$\dot{\hat{\tau}}^{\text{ext}}(t) = \bar{K}_o \left(\bar{p}(t) - \int_0^t \left(\bar{\tau} - \bar{n}(\bar{q}, \dot{\bar{q}}) + \dot{\hat{\tau}}^{\text{ext}} \right) dt - \bar{p}(0) \right) \quad (9)$$

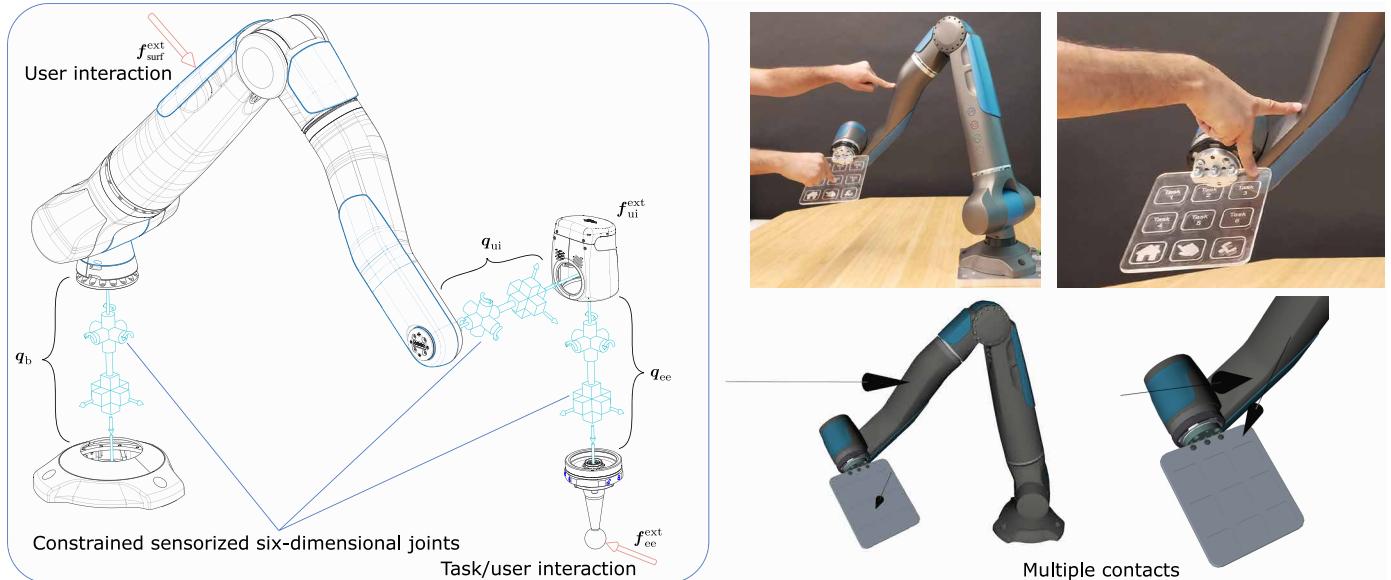


Fig. 7. The concept of force-torque sensing redundancy on the SARA robot. The expanded regions show a possible arrangement of 6D force-torque sensors, which is considered here as sensorized constrained joints. This illustrates a typical example for a collaborative manipulator with fully integrated sensing redundancy. The contact location is defined by the intersection between the line of force action and the geometrical surfaces. Because of the sensing redundancy concept and momentum monitoring, the multicontact locations can be determined in real time. The external force $\mathbf{f}_{\text{surf}}^{\text{ext}}$ applied along the robot structure is identified with its direction and localized at $I_{\text{surf}}(\alpha_{\text{surf}})$. Similarly, the applied external force $\mathbf{f}_{\text{ee}}^{\text{ext}}$ for the end effector is identified and localized correspondingly with $I_{\text{ee}}(\alpha_{\text{ee}})$.

The relation between the extended residual (9) and the external generalized forces is determined by $\dot{\tilde{\tau}}^{\text{ext}} = \bar{K}_o(\tilde{\tau}^{\text{ext}} - \hat{\tilde{\tau}}^{\text{ext}})$, which represents a first-order filter applied to the acting external forces. Further definitions of the used and sensed quantities can be found in the Supplementary Materials.

Multipoint localization

The sensing redundancy arrangement in combination with the momentum monitoring method facilitates the identification of multiple simultaneous contacts and their locations. We computed both the contact location and the intensity of interaction for single and multiple contact scenarios, and we unified the representation of the external forces by mapping them from different locations to the robot base. The base sensor acts, in this case, as a collection point that measures all of the interactions to be counterbalanced by the environment. Therefore, it can be used to split the external wrenches applied along the structure, that is,

$$\mathcal{B}\boldsymbol{\tau}_{\text{surf}}^{\text{ext}} = \boldsymbol{\tau}^{\text{ext}} - \text{Ad}_{g_B}^T \boldsymbol{\tau}_{\text{ui}}^{\text{ext}} - \text{Ad}_{g_B}^T \boldsymbol{\tau}_{\text{ee}}^{\text{ext}} \quad (10)$$

where $\mathcal{B}\boldsymbol{\tau}_{\text{surf}}^{\text{ext}} = [\mathcal{B}\mathbf{f}_{\text{surf}}^{\text{ext}} \ T \ \mathcal{B}\mathbf{m}_{\text{surf}}^{\text{ext}} \ T]^T \in \mathbb{R}^6$ is defined in the base frame \mathcal{B} . Touching the robot at random locations can be regarded as pushing or impulse forces with zero local torque; thus, the locations can be determined uniquely. We extracted the contact points along the structure by means of the associated external wrenches through the vector between the base frame and the line of force action using the pseudoinverse solution $\mathbf{x}_{\text{surf}} = -\mathbf{S}(\mathcal{B}\boldsymbol{\tau}_{\text{surf}}^{\text{ext}})^{\#} \mathcal{B}\mathbf{m}_{\text{surf}}^{\text{ext}}$. Herein, $\mathbf{S}(\cdot)$ describes the skew-symmetric matrix operator, and Ad is the adjoint of the transformation. The line of force action crossing \mathbf{x}_{surf} can be expressed as

$$I_{\text{surf}}(\alpha) = \mathbf{x}_{\text{surf}} + \alpha (\mathcal{B}\mathbf{f}_{\text{surf}}^{\text{ext}} / \|\mathcal{B}\mathbf{f}_{\text{surf}}^{\text{ext}}\|) \quad (11)$$

Having full knowledge of the robot geometry, we can identify the actual contact point along the line of the force action through the intersection with the surface S_{i_c} of the involved link i_c . This point corresponds to a specific value α_{surf} of the varying parameter α and accordingly to a specific location along the structure $I_{\text{surf}}(\alpha_{\text{surf}})$ as shown in Fig. 7. Similarly, we can map the external wrench at the end effector to the base frame as $\mathcal{B}\boldsymbol{\tau}_{\text{ee}}^{\text{ext}} = \text{Ad}_{g_B}^T \boldsymbol{\tau}_{\text{ee}}^{\text{ext}}$, where $\mathcal{B}\boldsymbol{\tau}_{\text{ee}}^{\text{ext}} = [\mathcal{B}\mathbf{f}_{\text{ee}}^{\text{ext}} \ T \ \mathcal{B}\mathbf{m}_{\text{ee}}^{\text{ext}} \ T]^T \in \mathbb{R}^6$, and determine the minimum distance \mathbf{x}_{ee} to the line of force action analogous to \mathbf{x}_{surf} . With reference to Eq. 11, the corresponding contact location $I_{\text{ee}}(\alpha_{\text{ee}})$ can be directly obtained. This process can be iteratively repeated until all available or redundant sensors in the areas of interest have been incorporated. Additionally, we can also find the contact at the user interface $I_{\text{ui}}(\alpha_{\text{ui}})$. The localization of the contacts can be robustly achieved over different regions on the structure, because the proposed method does not require a minimum number of involved joints. Even contacts between the base and the first joint of the robot can be correctly identified. The accurate contact localization based on the momentum monitoring method offers a practical way to provide tactile information using the intrinsic sensing redundancy. Naturally, the maximum number of simultaneous interactions to be detected is limited by the number of sensors or the richness of the sensing redundancy, respectively.

Touch trajectory extraction

We call the chronological sequence of contact points simply touch trajectory. Such trajectories basically describe the tracking of the contact force over space and time t , described by the respective

components $\mathbf{l}(t)$ (location) and $\mathbf{f}(t)$ (force intensity). In general, multiple trajectories can simultaneously occur. In this section, we considered the touch trajectory as user input in the form of ink for writing or drawing, which can be divided into separate pen strokes. The goal was to extract the textual interpretation of the input data. The touch trajectory is a sequence of N contact location points $\mathbf{l}(t) = (x(t), y(t), z(t))$ or $\mathbf{l}_i(t_i) = (x(t_i), y(t_i), z(t_i))$, where $i = \{1, \dots, N\}$, and it is also associated with the contact forces $\mathbf{f}(t) = (f_x(t), f_y(t), f_z(t))$. Usually, the force signal \mathbf{f} and its location \mathbf{l} are recorded at a high rate; thus, resampling the location data to a lower rate reduces the computation time for further processing. This resampling was conducted through the extraction of relevant points and the elimination of duplicates in the trajectory. Moreover, an equidistant linear sampling over the geometrical path was performed during these steps, making the approach independent of system-specific, native sampling or clock rates, respectively. For SARA, the number of data points was drastically reduced because the raw data were recorded at 8 kHz. The contact forces \mathbf{f} (which can be further specified as f_{surf} for the surface or f_{ee} for the end effector) are already filtered through the first-order dynamics of the momentum-based observer, parameterized by \bar{K}_o . For improved robustness, a localization contact force threshold was used, namely, 2 to 4 N for contacts on the main structure and 0.5 to 2 N for touches after the end effector, on the basis of the robot velocity. Additionally, the touch trajectory was filtered on the position level to remove any jitter resulting from writing on the robotic surface.

One way to approximate trajectories in space is through Bézier curves, as is commonly done in the representation of handwritten data on digital devices to obtain natural results (44). It provides both filtering and smoothing effects as well as a compact representation of the original input, which can be used to render the handwritten data smoothly or even upsample it afterward, if necessary. Here, parametric cubic Bézier polynomials were used to fit the 3D original trajectory. The polynomial coefficients Ω were computed to minimize the sum of squared errors (SSEs) between the smoothed trajectory $\hat{\mathbf{l}}$ with index s and the original one

$$\min_{\Omega} \sum_{i=1}^N \left(\hat{\mathbf{l}}(\mathbf{x}_s, \mathbf{y}_s, \mathbf{z}_s, \Omega) - \mathbf{l}(\mathbf{x}, \mathbf{y}, \mathbf{z}) \right)^2 \quad (12)$$

Additionally, the SSE was used as splitting criteria to define piecewise Bézier curves. If the error was larger than a specified value, the curve was split to meet this constraint, because the set of parameters Ω is individually determined for each piecewise Bézier curve. The splitting helps to preserve the intended sharp corners in the applied touch trajectory. In case of multistroke characters, the ink (touch trajectory) was segmented to split the different strokes on the basis of a force threshold. In other words, the SSE was used for separations within one stroke (between corners), whereas the force threshold was used to separate individual strokes, which are not connected. Figure 8 shows one example of a multistroke character, namely, the character “B,” where the splitting was performed on the basis of the SSE and the applied force threshold.

Dimensionality reduction

The touch trajectories were expressed as 3D curves in Euclidean space constrained by the surface of the robotic structure. This resulted in a 2D manifold embedded in 3D space. Consequently, the dimension

can be reduced without any loss of information. Manifold learning techniques offer tools to map from \mathbb{R}^D onto a lower-dimensional space \mathbb{R}^d (with $d < D$, which represents the unfolding of the originally applied shape. By deploying Isomap (43), one can perform the dimensionality reduction by means of a measured local metric to capture the underlying global geometry of the input data. Additionally, it provides a globally optimal solution for a broad class of nonlinear manifolds, and it guarantees asymptotic convergence to the actual geometric structure.

Let us consider the set of touch points $\mathbf{l} = \{\mathbf{l}_i\}_{i=1..N} \in \mathbb{R}^D$ (which can be further specified as \mathbf{l}_{surf} for the surface or \mathbf{l}_{ee} for the end effector) that belongs to the manifold S_{i_c} to be mapped to $\mathbf{x}_p = \{\mathbf{x}_{pi}\}_{i=1..N} \in \mathbb{R}^d$, where $D = 3$ and $d = 2$ for a planar unfolding. To increase the accuracy of the unfolded trajectory, the area of application can be extracted by sampling a set of surface points surrounding the original touch points. The points were placed along the perpendicular direction of the surface normal, and their number was kept limited (for example, five augmented points in each direction; see fig. S7) for faster computation but large enough to provide sufficient information about the manifold such that the unfolded result was less distorted. In this way, the local region around the touch points could be identified and used for the unfolding. More details are provided in fig. S7. Practically, the surface areas with low curvature were expected to be predominantly used as application regions because they might be preferred by humans. These segments are close to a surface with zero Gaussian curvature such that there exists a mapping in low-dimensional space that is also close to being isometric, causing little distortion on the 2D representation while using manifold learning approaches.

Typically, the resultant touch points are $(\mathbf{x}, \mathbf{y}, t)$ with position (\mathbf{x}, \mathbf{y}) and time stamp t . Extra information was associated with the touch trajectory, that is, the touch force amplitude $\|\mathbf{f}\|$ (norm of the contact force), which can be regarded as the pen pressure. The force was originally linked to the curved trajectory, but we kept the relation to the folded result for further usage. For example, this additional piece of information could be used to modulate the linewidth (ink width) of the handwriting to generate more natural shapes or to infer various commands on the basis of the specific intensity of the push. The generated 2D flat trajectories were independent of the area of application and the curvature of the writing surface. These resemble typical touch trajectories generated from digital touch devices. Beneficially, one can make use of rich, open-access handwriting datasets. Given that the original touch trajectory is randomly oriented in 3D space on the basis of the robot configuration and the writing perspective, the resultant 2D representation was expected to be randomly oriented in the plane likewise.

Trajectory classification and character recognition

At this point, no assumptions on the orientation of the handwritten trajectories were necessary. Actually, the touch trajectory is, by nature, randomly oriented depending on the robot configuration and the user’s perspective. Consequently, the unfolding yielded a 2D trajectory that was also randomly oriented likewise. It is desirable that the classification is independent of the orientation of the input data. Therefore, a rotation-invariant classifier was deployed to deal with arbitrarily rotated trajectories. Convolutional neural networks are known to robustly deal with shape variation given that they identify and learn position-independent features from input data. We used the RICNN with $SE(2)$ group convolution layers in the network

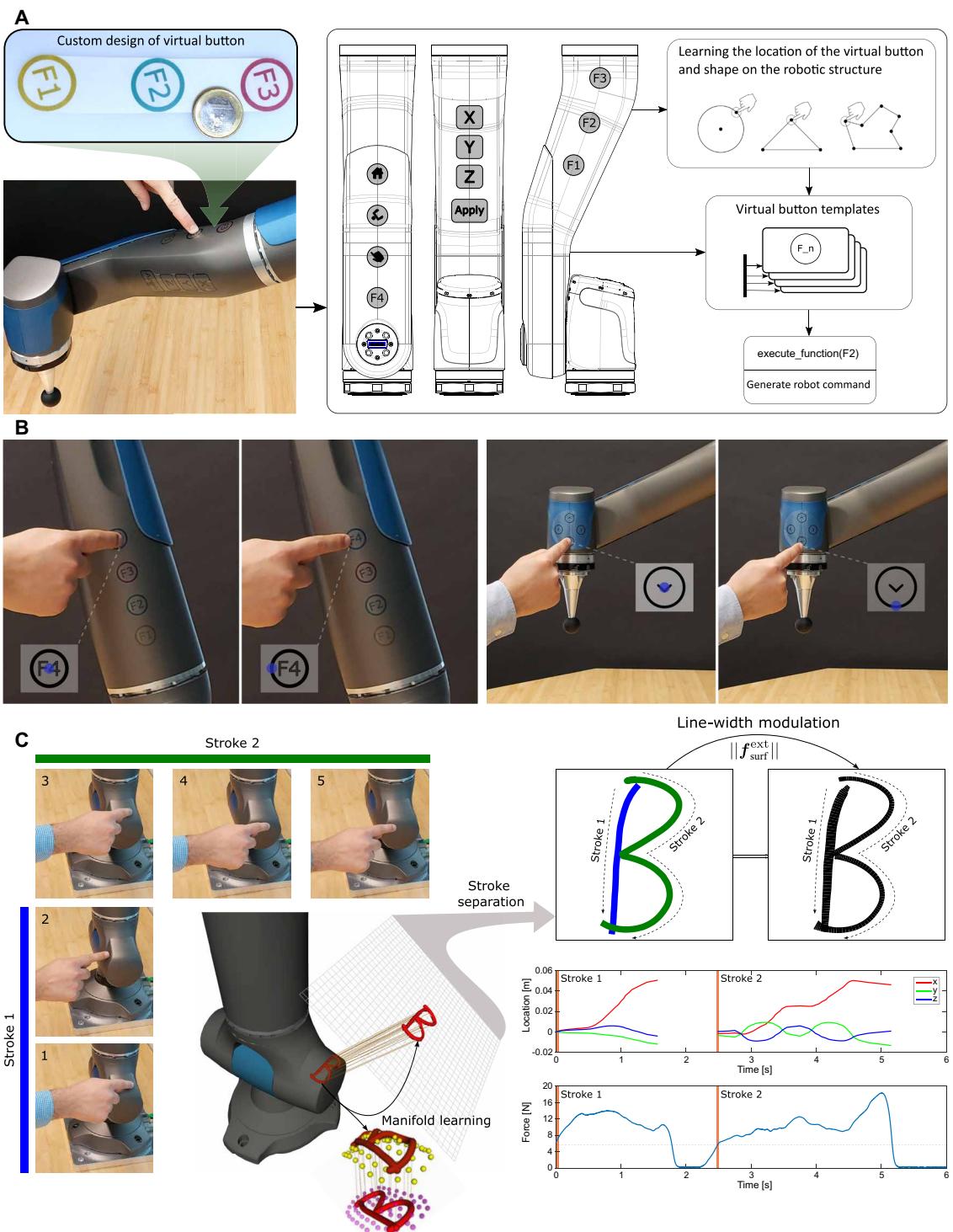


Fig. 8. Virtual button template, teaching, and multistroke segmentation. (A) The physical placement of buttons can be freely designed, and their shapes and locations can be taught by touching the robot surface (B). Each virtual button can be assigned to a specific function or task. In (C), the sequence of writing a multistroke character is shown: Images 1 and 2 are for the first stroke, whereas images 3 to 5 are for the second. The resultant of the unfolding through manifold learning is highlighted over a strongly convex part of the robot surface, and the second part depicts the separation of different strokes on the basis of the contact force magnitude.

(45), which encode the geometric structure of the special Euclidean group. This strategy enabled the use of standard datasets of handwritten characters to train the classifier. One of the most commonly used datasets for digits is MNIST, because its recent extension EMNIST (46) additionally includes 145,600 letters and 280,000 digits from more than 500 writers. The dataset is formatted as 28 pixel-by-28 pixel grayscale images, thus requiring the transformation of the unfolded trajectory points per stroke with variable linewidth into grayscale. Because the 3D trajectories are already smooth, no additional smoothing actions for the flat trajectories are required because they will not affect the downsampled pixelated version. We used a similar formatting procedure as in (46) to convert the 2D touch trajectory to the 28-by-28 vector of grayscale values, which was considered to be the input for the classifier. This conversion process ran online and resulted in a character image with random orientation depending on the direction of the application on the robot and its current configuration.

Initially, the flat coordinates of the applied trajectory were converted to black and white (bilevel) binary images, and the linewidth was modulated by the norm of the touch force (see fig. S8). Including the touch force was beneficial to weight the importance of specific parts of the symbol and to better match the linewidth variation observed on the training data (see fig. S9). In case of applying characters, the strokes were split on the basis of a recognition force threshold defined empirically (4 to 6 N). The steps of converting black and white images to the downsampled version can be seen in fig. S10. The online downsampling was conducted as follows: The edges of the lines in the black and white images were softened through a Gaussian filter; then, any empty spaces around the character were excluded. Afterward, the symbol was centered and placed in a square frame while preserving the aspect ratio of the extracted area. Then, the data were resized to a 28 pixel-by-28 pixel image using bicubic interpolation, resulting in an 8-bit grayscale resolution image. The implementation of the entire conversion process was optimized for online execution on the robotic system because it is commonly an offline operation.

Afterward, the classification was applied to the extracted vector that consisted of roto-translation covariant structures. These network elements allowed training on default-oriented (upright orientation) data and performing the prediction using the rotated input images, given that the created feature maps are rotation invariant. The invariance was achieved through the use of group convolutions, whereby each group represents a rotation. The network was constructed with a lifting layer that lifts a 2D vector-valued image to the position orientation space $\mathbb{R}^2 \times S^1 \equiv SE(2)$. This was achieved through the rotation of the filters by a predefined number of steps, defined by N_θ (see table S1 for the detailed architecture of the network). The shape of the filter was, as in regular convolutional neural networks, defined through the kernel size and number of output channels N_c . Applying each of the rotated filters to the input corresponded to a shift that represents a group. The following group convolution layer allowed the application of the filter kernels to each of the rotation groups. This enabled the feature extraction from each rotated group independently until the last layer, which projected back to \mathbb{R}^2 . The extracted features from each rotation group in the last layer were in a different order, depending on the degree of rotation of the input image. This last layer then selected the feature vectors with the highest score in each group, indicating the classification result.

We evaluated the classification method using recorded touch data under varying parameters to choose a suitable network structure for our application (see figs. S1 and S2). The proposed approach provides high flexibility to be extended to other sets of symbols or other languages. The user might include emojis for specific tasks, for example, or other elements to be used in a more generic semantic context.

Virtual buttons

The flexible user interface of virtual buttons was introduced, which can be arbitrarily placed on the robotic structure. A user has complete freedom to design, (re)configure, and assign functions to such buttons without the need for additional hardware or devices. After physically placing the buttons at desired locations, their shapes and locations can be taught by touching the surface of the robot. One simple way is to print the virtual buttons as stickers (see Fig. 8A). The output signal can be categorized as a discrete button or as a continuous one including the magnitude of the contact force. In the case of a digital button, a Boolean signal/variable was assigned to it according to the point of interaction and the constant (specified) force threshold. Figure 8B depicts the teaching sequence of functional buttons through the selection of the feature points. In the case of a circular shape, the feature points were chosen within the circumference and the center. A template could be designed with the most common geometrical shapes, or alternatively, the area of interest could be manually taught.

Statistical analysis

We used the confusion matrix to calculate the accuracy of each class with respect to actual classes and the total recognition accuracy score as a performance measure. The overall classification accuracy was used for performance evaluation and the selection of the neural network parameters. The writing force average for the single applied touch trajectories was used to quantify the median and distribution using density curves of contact forces within the 2300 collected samples.

Supplementary Materials

The PDF file includes:

Methods

Figs. S1 to S10

Table S1

Other Supplementary Material for this manuscript includes the following:

Movies S1 to S8

REFERENCES AND NOTES

1. A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, P. Fraisse, Collaborative manufacturing with physical human–robot interaction. *Robot. Comput. Integrat. Manuf.* **40**, 1–13 (2016).
2. A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, O. Khatib, Progress and prospects of the human–robot collaboration. *Auton. Robots* **42**, 957–975 (2018).
3. M. Panzirsch, A. Pereira, H. Singh, B. Weber, E. Ferreira, A. Gherghescu, L. Hann, E. den Exter, F. van der Hulst, L. Gerdes, L. Cencetti, K. Wormnes, J. Grenouilleau, W. Carey, R. Balachandran, T. Hulin, C. Ott, D. Leidner, A. Albu-Schäffer, N. Y. Lii, T. Krüger, Exploring planet geology through force-feedback telemanipulation from orbit. *Sci. Robot.* **7**, eabl6307 (2022).
4. P. E. Dupont, B. J. Nelson, M. Goldfarb, B. Hannaford, A. Menciassi, M. K. O’Malley, N. Simaan, P. Valdastri, G.-Z. Yang, A decade retrospective of medical robotics research from 2010 to 2020. *Sci. Robot.* **6**, eabi8017 (2021).

5. J. Vogel, D. Leidner, A. Hagengruber, M. Panzirsch, B. Bäuml, M. Denninger, U. Hillenbrand, L. Suchenwirth, P. Schmaus, M. Sewitz, A. Bauer, T. Hulin, M. Iskandar, G. Quere, A. Albu-Schäffer, A. Dietrich, An ecosystem for heterogeneous robotic assistants in caregiving: Core functionalities and use cases. *IEEE Robot. Autom. Mag.* **28**, 12–28 (2020).
6. S. Liu, X. Wu, D. Zhang, C. Guo, P. Wang, W. Hu, X. Li, X. Zhou, H. Xu, C. Luo, J. Zhang, J. Chu, Ultrafast dynamic pressure sensors based on graphene hybrid structure. *ACS Appl. Mater. Interfaces* **9**, 24148–24154 (2017).
7. S. Hong, H. Lee, J. Lee, J. Kwon, S. Han, Y. D. Suh, H. Cho, J. Shin, J. Yeo, S. H. Ko, Highly stretchable and transparent metal nanowire heater for wearable electronics applications. *Adv. Mater.* **27**, 4744–4751 (2015).
8. H. Liu, T. Fang, T. Zhou, L. Wang, Towards robust human-robot collaborative manufacturing: Multimodal fusion. *IEEE Access* **6**, 74762–74771 (2018).
9. V. Villani, F. Pini, F. Leali, C. Secci, Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics* **55**, 248–266 (2018).
10. R.-J. Halmé, M. Lanz, J. Kämäriinen, R. Pieters, J. Latokartano, A. Hietanen, Review of vision-based safety systems for human-robot collaboration. *Procedia CIRP* **72**, 111–116 (2018).
11. L. Bascetta, G. Ferretti, P. Rocco, H. Ardö, H. Bruyninckx, E. Demeester, E. Di Lello, Towards safe human-robot interaction in robotic cells: An approach based on visual tracking and intention estimation, in 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE, 2011), pp. 2971–2978.
12. H. Liu, L. Wang, Gesture recognition for human-robot collaboration: A review. *Int. J. Ind. Ergon.* **68**, 355–367 (2018).
13. E. Theodoridou, L. Cinque, F. Mignosi, G. Placidi, M. Polzinelli, J. M. R. S. Tavares, M. Spezialetti, Hand tracking and gesture recognition by multiple contactless sensors: A survey. *IEEE Trans. Hum. Mach. Syst.* **53**, 35–43 (2023).
14. R. Stiefelhagen, C. Fugen, R. Gieselmann, H. Holzapfel, K. Nickel, A. Waibel, Natural human-robot interaction using speech, head pose and gestures, in 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE, 2004), vol. 3, pp. 2422–2427.
15. H. Yan, M. H. Ang, A. N. Poo, A survey on perception methods for human–robot interaction in social robots. *Int. J. Soc. Robot.* **6**, 85–119 (2014).
16. A. De Santis, B. Siciliano, A. De Luca, A. Bicchi, An atlas of physical human–robot interaction. *Mech. Mach. Theory* **43**, 253–270 (2008).
17. S. Haddadin, E. Croft, “Physical human–robot interaction” in *Springer Handbook of Robotics*, B. Siciliano, O. Khatib, Eds. (Springer, 2016), pp. 1835–1874.
18. C. Bartolozzi, L. Natale, F. Nori, G. Metta, Robots with a sense of touch. *Nat. Mater.* **15**, 921–925 (2016).
19. R. S. Dahiya, G. Metta, M. Valle, G. Sandini, Tactile sensing—From humans to humanoids. *IEEE Trans. Robot.* **26**, 1–20 (2009).
20. G. Cheng, E. Dean-Leon, F. Bergner, J. R. G. Olvera, Q. Leboulet, P. Mittendorfer, A comprehensive realization of robot skin: Sensors, sensing, control, and applications. *Proc. IEEE* **107**, 2034–2051 (2019).
21. P. Mittendorfer, G. Cheng, Humanoid multimodal tactile-sensing modules. *IEEE Trans. Robot.* **27**, 401–410 (2011).
22. S. Armleder, E. Dean-Leon, F. Bergner, G. Cheng, Interactive force control based on multimodal robot skin for physical human-robot collaboration. *Adv. Intell. Syst.* **4**, 2100047 (2022).
23. L. Massari, G. Fransvea, J. D’Abbraccio, M. Filosa, G. Terruso, A. Aliperta, G. D’Alesio, M. Zaltieri, E. Schena, E. Palermo, E. Sinibaldi, C. M. Oddo, Functional mimicry of Ruffini receptors with fibre Bragg gratings and deep neural networks enables a bio-inspired large-area tactile-sensitive skin. *Nat. Mach. Intell.* **4**, 425–435 (2022).
24. Y. Park, J. Jung, Y. Lee, D. Lee, J. J. Vlassak, Y.-L. Park, Liquid-metal micro-networks with strain-induced conductivity for soft electronics and robotic skin. *NPJ Flex. Electron.* **6**, 81 (2022).
25. S. Li, X. Chen, X. Li, H. Tian, C. Wang, B. Nie, J. He, J. Shao, Bioinspired robot skin with mechanically gated electron channels for sliding tactile perception. *Sci. Adv.* **8**, eade0720 (2022).
26. Y. Yan, Z. Hu, Z. Yang, W. Yuan, C. Song, J. Pan, Y. Shen, Soft magnetic skin for super-resolution tactile sensing with force self-decoupling. *Sci. Robot.* **6**, eabc8801 (2021).
27. K. Park, H. Yuk, M. Yang, J. Cho, H. Lee, J. Kim, A biomimetic elastomeric robot skin using electrical impedance and acoustic tomography for tactile sensing. *Sci. Robot.* **7**, eabm7187 (2022).
28. G. Walker, A review of technologies for sensing contact location on the surface of a display. *J. Soc. Inf. Disp.* **20**, 413–440 (2012).
29. S. J. Dempsey, M. Szablewski, D. Atkinson, Tactile sensing in human–computer interfaces: The inclusion of pressure sensitivity as a third dimension of user input. *Sens. Actuators A. Phys.* **232**, 229–250 (2015).
30. S. Haddadin, A. De Luca, A. Albu-Schäffer, Robot collisions: A survey on detection, isolation, and identification. *IEEE Trans. Robot.* **33**, 1292–1312 (2017).
31. M. Iskandar, C. Ott, A. Albu-Schäffer, B. Siciliano, A. Dietrich, Hybrid force- impedance control for fast end-effector motions. *IEEE Robot. Autom. Lett.* **8**, 3931–3938 (2023).
32. E. Magrini, F. Flacco, A. De Luca, Control of generalized contact motion and force in physical human-robot interaction, in 2015 IEEE International Conference on Robotics and Automation (ICRA) (IEEE, 2015), pp. 2298–2304.
33. E. Magrini, F. Flacco, A. De Luca, Estimation of contact forces using a virtual force sensor, in 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE, 2014), pp. 2126–2133.
34. F. Flacco, T. Kröger, A. De Luca, O. Khatib, A depth space approach to human-robot collision avoidance, in 2012 IEEE International Conference on Robotics and Automation (IEEE, 2012), pp. 338–345.
35. L. Manuelli, R. Tedrake, Localizing external contact using proprioceptive sensors: The contact particle filter, in 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE, 2016), pp. 5062–5069.
36. A. Zwirner, C. Geckeler, A. Zell, Contact point localization for articulated manipulators with proprioceptive sensors and machine learning, in 2018 IEEE International Conference on Robotics and Automation (ICRA) (IEEE, 2018), pp. 323–329.
37. M. Iskandar, O. Eiberger, A. Albu-Schäffer, A. De Luca, A. Dietrich, Collision detection, identification, and localization on the DLR SARA robot with sensing redundancy, in 2021 IEEE International Conference on Robotics and Automation (ICRA) (IEEE, 2021), pp. 3111–3117.
38. L. Wang, R. Gao, J. Váncza, J. Krüger, X. V. Wang, S. Markis, G. Chrysoulouris, Symbiotic human-robot collaborative assembly. *CIRP Ann.* **68**, 701–726 (2019).
39. S. E. Navarro, S. Mühlbacher-Karrer, H. Alagi, H. Zangl, K. Koyama, B. Hein, C. Duriez, J. R. Smith, Proximity perception in human-centered robotics: A survey on sensing systems and applications. *IEEE Trans. Robot.* **38**, 1599–1620 (2021).
40. S. Haddadin, M. Suppa, S. Fuchs, T. Bodenmüller, A. Albu-Schäffer, G. Hirzinger, Towards the robotic co-worker, in *Robotics Research: The 14th International Symposium ISRR* (Springer, 2011), pp. 261–282.
41. G. Hirzinger, N. Sporer, A. Albu-Schäffer, M. Hahnle, R. Krenn, A. Pascucci, M. Schedl, DLR’s torque-controlled light weight Robot III—Are we reaching the technological limits now?, in *Proceedings 2002 IEEE International Conference on Robotics and Automation* (IEEE, 2002), vol. 2, pp. 1710–1716.
42. M. Iskandar, C. Ott, O. Eiberger, M. Keppler, A. Albu-Schäffer, A. Dietrich, Joint-level control of the DLR lightweight robot SARA, in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE, 2020), pp. 8903–8910.
43. J. B. Tenenbaum, V. de Silva, J. D. Langford, A global geometric framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323 (2000).
44. V. Carbune, P. Gonnet, T. Deselaers, H. A. Rowley, A. Daryin, M. Calvo, L.-L. Wang, D. Keysers, S. Feuz, P. Gervais, Fast multi-language LSTM-based online handwriting recognition. *Int. J. Doc. Anal. Recog.* **23**, 89–102 (2020).
45. E. J. Bekkers, M. W. Lafarge, M. Veta, K. A. J. Eppenhof, J. P. W. Pluim, R. Duits, Roto-translation covariant convolutional networks for medical image analysis, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, 2018), pp. 440–448.
46. G. Cohen, S. Afshar, J. Tapson, A. Van Schaik, Emnist: Extending MNIST to handwritten letters, in 2017 International Joint Conference on Neural Networks (IJCNN) (IEEE, 2017), pp. 2921–2926.

Acknowledgments: We would like to thank F. Schmidt, O. Eiberger, A. Stemmer, S. Gebel, and M. Kreiner for their support on the hardware and software parts. **Funding:** Funding was provided by the Institute of Robotics and Mechatronics, German Aerospace Center (DLR) to all authors. **Author contributions:** M.I. generated and formulated the project ideas, implemented the system, led the research project, performed the experiments, analyzed the data, wrote the paper, and is the main author. A.A.-S. revised and modified the manuscript and provided funding. A.D. contributed to the project conception, data analyses, paper writing, and supervision and provided funding. **Competing interests:** The following patent applications related to this work have been filed: German patent application DE102021113636 and European patent application EP22175265.2 (M.I., A.D., and A.A.-S.), German patent application DE102021106251 and European patent application EP21212707.0 (M.I. and A.D.), and German patent application DE102020134260.8 and European patent application EP21202126.5 (M.I., A.D., and O. Eiberger.). **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials. Other materials can be found at DOI:10.5281/zenodo.13237141.

Submitted 12 December 2023

Accepted 23 July 2024

Published 21 August 2024

10.1126/scirobotics.adn4008