Controlling Robotic Manipulations via Bimanual Gesture Sequences

Petr Vanc, Karla Stepanova, and Jan Kristof Behrens

***Abstract*— Gestures are a natural part of human communication and improve its expressiveness. In human-robot interaction, gestures are often manually mapped to be a trigger for specifically designed operations. In this paper, we propose a probabilistic framework that enables to use a combination of static and dynamic gestures to describe a context-dependent parametrized robot action. The interactive specification of new gestures and their grounding to robot motions skills or a parameter (e.g., goal pose, speed, distance, etc.) is enabled via GUI. We demonstrate the proposed system in simulated block world experiments.**

I. Introduction

Gestures are an important part of human communication. They develop in early childhood prior to language [1] and serve together with protolinguistic verbal communication as an important tool how to communicate with an adult. They stay an integral part of nonverbal communication also in adulthood. Similarly, in robotics, they might enable communication on the long-distance, in a noisy environment, or as an addition to verbal commands. They can convey efficiently many different concepts like positions, distances, angles, selections, approval, or rejection.

A lot of work has been done on how individual static and dynamic gestures might be learned and classified (e.g., [2], [3], [4]). However, gestures are not much utilized in industrial robotics because the meaning of gestures is very much context-dependent. Furthermore, a single gesture is not expressive enough to communicate to a robot the human intent or even the next action to be executed. Current works utilizing gesture-based human-robot interaction typically use gestures to trigger predefined robot motions (e.g., [5], [6]). On contrary, for humans, the same gesture might have a different meaning based on the context.

In this paper, we present our progress on combining several gestures to lift the expressiveness and precision. Most of the robot motions are represented as probabilistic motion primitives, which enables us to further extend the variability of the motions and take a context into account (e.g., by setting a goal position, or other parameters of the motion). Our approach closes a control loop where the user receives visual and audio feedback from a graphical user interface (GUI) and the robot (see Fig. 1). The user's hands are continuously analyzed to extract gestures from the sequence of hand states via the proposed probabilistic framework. The human might switch between different modes of operation - direct operation of the robotic arm and operating via gestures. The system enables also to specify new gestures and connect them to corresponding robot motions or actions. To test the system we collected our dataset consisting of 12 static and 6 dynamic gestures, each gesture has about 60 samples.

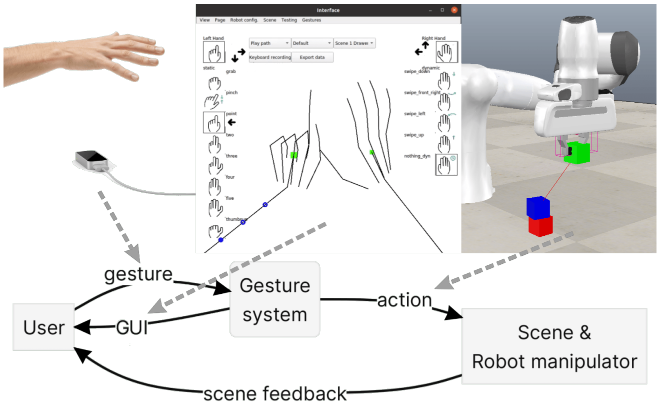

Fig. 1: Human-Robot control loop diagram. GUI (top center) and a simulated robot provide visual feedback to the user about the detected gestures (left hand: point static gesture, right hand: no gesture) and intended robot movement. Stacking cubes scenario in CoppeliaSim simulator (right).

The system is demonstrated on use cases in simulated blocks world. A robotic manipulator shall manipulate colored blocks on a table. The user instructs the robot using gestures.

II. System for Continuous Gesture Detection

In this section, individual parts of the system are described including the representation of the gestures and robot actions.

1) Gesture definitions: We consider two types of hand gestures - static and dynamic and our system enables us to create also more complex (compound) gestures. The gestures are classified in real-time, using a sliding window of the last n observations $[O_{t-1}, \cdots, O_{t-n}]$, where n is a time horizon (preset to 3 seconds). If the duration of the gesture would extend this time horizon, multiple gestures would be detected. Observation O is defined as a vector of hand parameters processed from raw hand bone structure. For static gestures $O = [a_1, \cdots, a_n, d_1, \cdots, d_m]$, where $a_1, \cdots, a_n$ are the bone angles of the hand, $d_1, \cdots d_m$ are combinations of fingers tip distances. For dynamic gestures, we consider only values of hand position: $O_i = [x, y, z]$.

Static gestures are primarily used to express different parameters (e.g., speed, distance, goal position), trigger action

J.K.B. was supported by the European Regional Development Fund under project Robotics for Industry 4.0 (reg. no. CZ.02.1.01/0.0/0.0/15_003/0000470). K.S. was supported by the Czech Science Foundation (project no. GA21-31000S). P.V by CTU Student Grant Agency (reg. no. SGS21/184/OHK3/3T/37).

All authors are with the Czech Institute of Informatics, Robotics, and Cybernetics, CTU in Prague, CR [jan.kristof.behrens, karla.stepanova]@cvut.cz

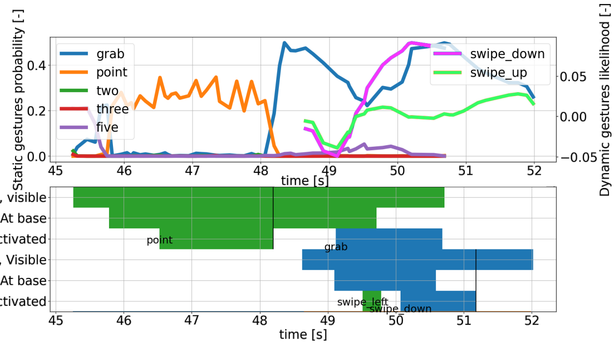

Fig. 2: Gesture detection output. The upper plot shows likelihoods for static (left legend) and dynamic (right legend) gestures. The bottom plot shows visibility of the hands and activation of the gestures. First was detected the static gesture *point* followed by *grab* and dynamic gesture *swipe down*.

(e.g., open/close gripper), switch between individual modes (direct operation/gesture operation/building mode), or select object on which the action will be performed. Dynamic gestures are mainly used for specification of individual dynamic motions (e.g., moving up/down, left/right, in a spiral, etc.). The dynamic gestures are represented as probabilistic motion primitives (ProMPs) and classified using Dynamic time warping (DTW), static gestures are classified using the Bayesian Neural Network in PyMC3. In Fig.2 are shown probabilities of detection of individual static and dynamic gestures over time. The gesture is signalized as detected if the given threshold of accumulated probability is reached.

2) Robotic actions: Robotic actions/motions are represented as a trajectory of robot joints, including the gripper positions. We developed a trajectory generator that takes a motion name together with attached variables (e.g., goal/start position/orientation, etc.) and outputs a trajectory. User-defined generators might be added. In our experiments, we worked with a ProMP generator to represent robotic motions.

3) Mapping gestures to robotic actions: Mapping between robot actions $\mathbf{R} = \{R_1, ..., R_K\}$ and gestures $\mathbf{G} = \{G_1, ..., G_M\}$ is given by a one-to-one mapping: $f_i : \mathbf{G}_m \rightarrow \mathbf{R}_k,\ \mathbf{R}_k \subset \mathbf{R}, \mathbf{G}_m \subset \mathbf{G}$. The mapping is use-case specific and adjustable by a user. Only a subset of robot actions or gestures might be used in each use-case. The set of mappings $\mathbf{F} = \{f_i\}$ for all use-cases i is saved in the database.

4) Scenario modelling: To demonstrate the capabilities of the system, we created a simple simulated block-world use-case. The goal is to showcase the smart pick-and-place method. See a stacking boxes use-case in Fig 1 (top right). A subset of 7 static gestures $\mathbf{G}_m^s$ (1-5 fingers, grab and open hand) is mapped to the following robot actions: *object_focus_picker* enabling switching between objects and open/close gripper action. A subset of 4 dynamic gestures $\mathbf{G}_m^d$ (swipe gestures) is mapped to the following robot actions: move to the home position, touch the focused object with the end-effector, and kick the focused object (the trained ProMP enables conditioning of the start and end position).

5) Proposed system design: In Fig. 1 is visualized the interaction of the user with the system. In Fig. 3 is shown the underlying ROS architecture.

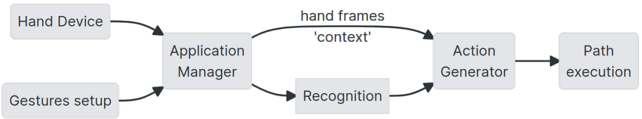

Fig. 3: The system diagram displayed as ROS nodes.

III. Experimental Setup and Results

The experimental setup consists of a Leap motion controller and CopeliaSim [7] simulator with Franka Emika Panda as a robot manipulator.

System functionality was tested on mentioned scenarios, so far only from the user's perspective. Stack of 3 boxes via gestures was performed under 40 seconds. An example of the recognition in Fig. 2 shows when the used method for static and dynamic gestures has accurately predicted *point* and *swipe down* gestures. On top of that, the logic was applied to make the detection robust against incoherence.

IV. Conclusions

In this paper we presented a system for controlling a robot via gestures, taking into account the context of the situation. The system shows good robustness in the case that the selected set of gestures is enough distinct. In the current form the system enables a good user-experience for the evaluated scenarios. In the future, we want to evaluate accuracy of individual components and different sets of gestures, perform an extended user-study, develop more advanced use-cases, and focus on the semantic meaning of the gestures. We would also like to take an inspiration from models of gestures and language development, e.g. social babbling [8] to increase the understanding of the hand movements.

References

[1] M. C. Caselli, P. Rinaldi, S. Stefanini, and V. Volterra, "Early action and gesture "vocabulary" and its relation with word comprehension and production," *Child development*, vol. 83, no. 2, pp. 526–542, 2012.

[2] G. Marin, F. Dominio, and P. Zanuttigh, "Hand gesture recognition with leap motion and kinect devices," in *2014 IEEE International conference on image processing (ICIP)*. IEEE, 2014, pp. 1565–1569.

[3] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 131–153, 2019.

[4] A. Mujahid, M. J. Awan, A. Yasin, M. A. Mohammed, R. Damaševičius, R. Maskeliūnas, and K. H. Abdulkareem, "Real-time hand gesture recognition based on deep learning YOLOv3 model," *Applied Sciences*, vol. 11, no. 9, p. 4164, 2021.

[5] P. Neto, M. Simão, N. Mendes, and M. Safeea, "Gesture-based human-robot interaction for human assistance in manufacturing," *The Int. Jour. of Adv. Manuf. Tech.*, vol. 101, no. 1, pp. 119–135, Mar. 2019.

[6] B. Burger, I. Ferrané, F. Lerasle, and G. Infantes, "Two-handed gesture recognition and fusion with speech to command a robot," *Autonomous Robots*, vol. 32, no. 2, pp. 129–147, 2012.

[7] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 1321–1326.

[8] L. Cohen and A. Billard, "Social babbling: The emergence of symbolic gestures and words," *Neural Networks*, vol. 106, pp. 194–204, 2018.