Pose Forecasting in Industrial Human-Robot Collaboration Supplementary Material

Alessio Sampieri[1], Guido Maria D'Amely di Melendugno[1], Andrea Avogaro[2], Federico Cunico[2], Francesco Setti[2], Geri Skenderi[2], Marco Cristani[2], and Fabio Galasso[1]

[1] Sapienza University of Rome
{sampieri, damely, galasso}@di.uniroma1.it
[2] University of Verona
{andrea.avogaro, federico.cunico, francesco.setti, geri.skenderi, marco.cristani}@univr.it

We supplement the main paper submission with extra information in this document. The extra information in this document is organized as follows:

- Sec. 1 supplements the main paper with more detailed descriptions and illustrations of each action in CHICO. This extends the descriptions and illustrations in the *main paper*.
- Sec. 2 discusses some additional details regarding the implementation and learning of SeS-GCN.

1 The CHICO Dataset

Here we illustrate additional details on the scenario of the dataset and the acquisition process; furthermore, we report a graphical explanation of all the 8 actions. For further details, please watch the additional video, enclosed in https://github.com/AlessioSam/CHICO-PoseForecasting.

1.1 Details on the dataset and the data acquisition process

The dataset has been acquired in the October '21 - March '22 period, on a $500m^2$ Industry 4.0 lab, which includes a configurable a 11m production line, 4 cobots, a quality inspection cell, a (dis-)assembly station and other equipment. We worked on the 0.9 m $\times$0.6 m workbench of the (dis-)assembly station in front of a Kuka LBR iiwa 14 R820 cobot. The declared positioning accuracy is ± 0.1 mm and the axis-specific torque accuracy is $\pm 2\,\%$ [6]. Thanks to its joint torque sensors, the robot can detect contact and reduce its level of force and speed, being compliant to the ISO/TS 15066:2016 [4] standard. Since collisions between the operator and the cobot were expected in CHICO, the maximum allowed Cartesian speed of each link is set to 200 mm/s, slightly lower than the ISO/TS 15066:2016 requirements. The safety torque limit allowed before the mechanical brakes activation is set to 30 N m for all joints and 50 N m for the end-effector. Additionally, a programmable safety check of 10 N was set on the Cartesian force.

A total of 20 subjects (17 males, 3 females, average age 23 years) have been hired for building the dataset. They worked for the entire acquisition period, after having signed an informed consent and participated on a crash course on how to cooperate with the Kuka cobot. During the acquisition season, we have selected some excerpts which capture collisions made by the operators, reaching 226 collisions. In average we have 37 collisions for each action, with the sole exception for *hammering*. For this specific action, the cobot stands still, holding the object being hammered, while the human agent moves repeatedly close to the robotic arm. Sequences containing this action are still part of the collision detection dataset, i.e. they are useful to check that there are no false positives.

1.2 Details on the actions

In this section we expand the short explanations of each action given in Sec.4 of the main paper. For each action, we report the original description, and a detailed graphical storyboard.

- ***Lightweight pick and place*** (*Light P&P*). The human operator is required to move small objects of approximately 50 grams from a loading bay to a delivery location within a given time slot. The bay and the delivery location are at the opposite sides of the workbench. Meanwhile, the robot loads on of this bay so that the human operator has to pass close to the robotic arm.

In many cases the distance between the limbs and the robotic arm is few centimeters.

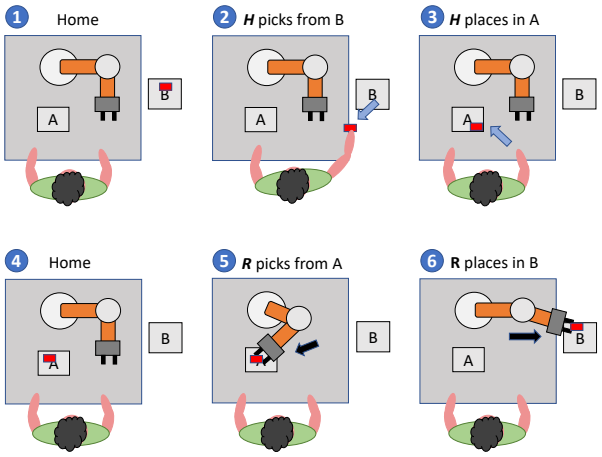

Fig. 1. ***Lightweight pick and place*** illustration; "***H***" stands for "human operator", "***R***" for "robot". A single item (the red brick) is showed here for clarity. In practice, a dozen of items was available.

– ***Heavyweight pick and place*** (*Heavy P&P*). The setup of this action is the same as before, but the objects to be moved are floor tiles weighing 0.75 kg. This means that the actions have to be carried out with two hands.

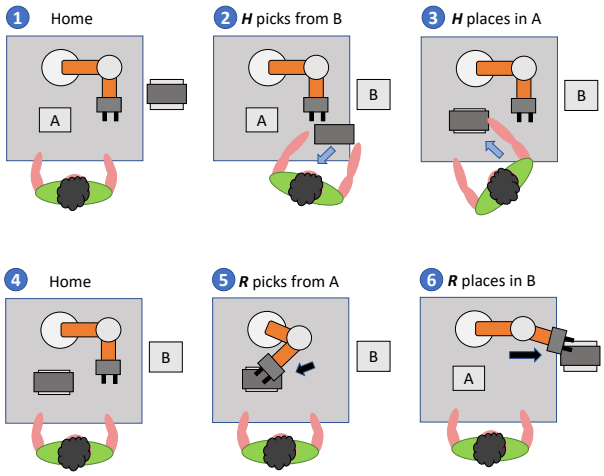

Fig. 2. ***Heavyweight pick and place*** illustration; "***H***" stands for "human operator", "***R***" for "robot". Moving the object with two hands requires to strongly rotating the torso, and this partially hides the robot from the operator, being back to him/her. This was the main cause of the collisions occurred with this action.

– ***Surface polishing*** (*Polishing*). This action was inspired by [7], where the human operator polishes the border of a 40 by 60cm tile with some abrasive sponge, and the robot mimics a visual quality inspection.

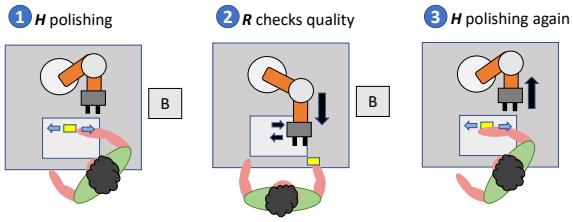

Fig. 3. ***Surface polishing*** illustration; "***H***" stands for "human operator", "***R***" for "robot". The human has an abrasive sponge used to remove some material from the metallic tile. This action created most of the collisions, since the action required the user to be prone on the surface to polish, blocking the view of the robot.

– ***Precision pick and place*** (*Prec. P&P*). The robot places four plastic pieces in the four corners of a 30×30cm table in the center of the workbench, and the human has to remove them and put on a bay, before the robot repeats the same unloading.

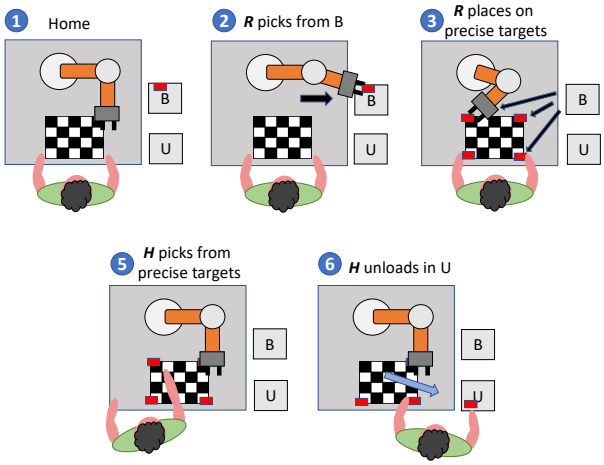

Fig. 4. ***Precision pick and place*** illustration; "***H***" stands for "human operator", "***R***" for "robot". This action is important, since it allows to measure how precise the prediction could be in individuating particular endpoints which will be targeted by the human operator.

– ***Random pick and place*** (*Rnd. P&P*). Same as the previous action, except for the plastic pieces which were continuously placed by the robot randomly on the central 30×30cm table, and the human operator has to remove them.

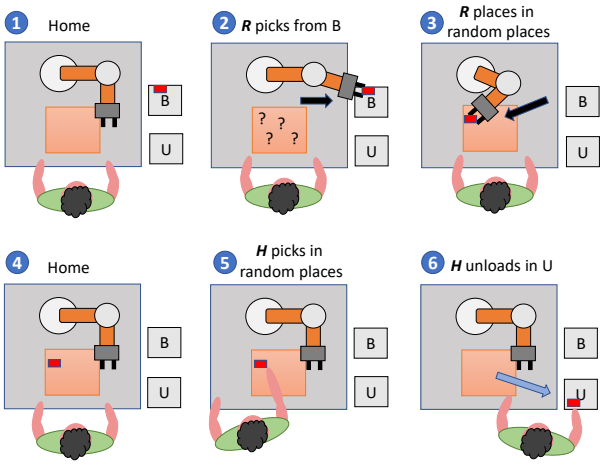

Fig. 5. ***Random pick and place*** illustration; "***H***" stands for "human operator", "***R***" for "robot". The action is interesting, since the robot puts objects randomly on the workplace, and this created some collisions during the interaction. A single item (the red brick) is showed here for clarity. In practice, a dozen of items was available.

– ***High shelf lifting*** (*High lift*). The goal was to pick light plastic pieces (50 grams each) on a sideway bay filled by the robot, putting them on a shelf located at 1.70m, at the opposite side of the workbench. Due to the geometry of the workspace, the arms of the human operator were required to pass above or below the moving robotic arm. In this way, close distances between the human arm and forearm and the robotic links were realized.

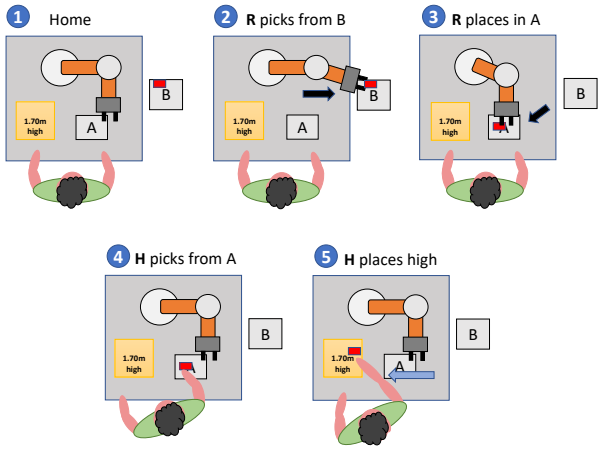

Fig. 6. ***High shelf lifting*** illustration; "***H***" stands for "human operator", "***R***" for "robot".

– ***Hammering*** (*Hammer*). The operator hits with a hammer a metallic tide held by the robot. In this case, the interest was to check how much the collision detection is robust to an action where the human arm is colliding close to the robotic arm (that is, on the metallic tile) without properly colliding *with the robotic arm*.

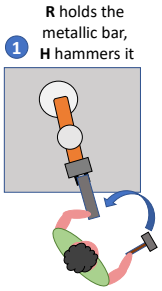

Fig. 7. ***Hammering*** illustration; "***H***" stands for "human operator", "***R***" for "robot". This action is important since it tests the collision prediction task against false positive alarms. In facts, this action requires the human to be very close to the robot, keeping the item to hammer with one hand, the other doing the hammering action. In facts, this action was creating the most false positive alarms, around 80%.

2 Implementation and Learning

Further to what included in Secs. 3 and 5 *(main paper)*, we discuss here a few additional details on implementation and learning.

2.1 Implementation details

The proposed SeS-GCN is written in Pytorch (the source code will be distributed). The model adopts residual connections at each GCN layers, it is regularized with batch normalization [2] at the end of each GCN layer, and it is optimized with ADAM [5].

2.2 Learning time

On Human3.6M [3], training of SeS-GCN proceeds for 60 epochs for both the teacher and the student models. We used batch size of 256, learning rate of 0.1, and decay rate of 0.1 at epochs 5, 20, 30 and 37. On an Nvidia RTX 2060 GPU, the learning process takes 30 minutes.

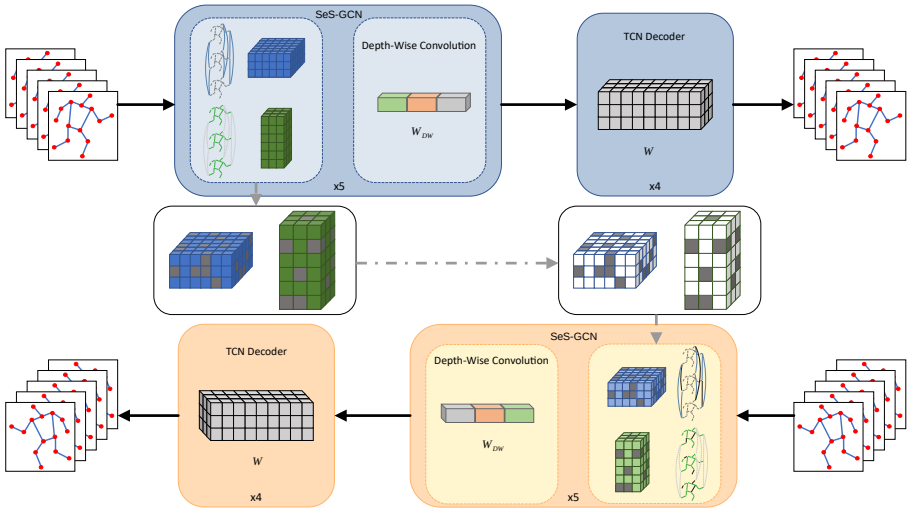

Fig. 8. Overview of the proposed pipeline. Given a sequence of observed 3D poses, the Teacher network (depicted with blue boxes) encodes the spatio-temporal body dynamics with 5 SeS-GCN layers, composed by the space-time separable encoder followed by the Depth-Wise convolution. The future trajectories are then predicted with 4 TCN layers. After the train of the Teacher, we threshold the values of the spatial and temporal adjacency matrix to obtain the masks which are then applied during the Student model (depicted in orange boxes) training.

2.3 Loss function

Following literature [9,8,1], the loss function differs from the test metric, Eq. (5) main paper; namely, the loss function considers the average of MPJPEs over the entire predicted sequence:

$$L_{\text{MPJPE}} = \frac{1}{VT} \sum_{t=0}^{T} \sum_{v=1}^{V} ||\hat{\boldsymbol{x}}_{vt} - \boldsymbol{x}_{vt}||_2$$

where, in accordance with Eq. (5), $\hat{\boldsymbol{x}}_{vt}$ and $\boldsymbol{x}_{vt}$ are the 3-dimensional vectors of a target joint j_v $(0 \leq v \leq V)$ in a fixed frame f_t $(0 \leq t \leq T)$ for the ground truth and the predictions, respectively.

References

1. Dang, L., Nie, Y., Long, C., Zhang, Q., Li, G.: MSR-GCN: Multi-scale residual graph convolution networks for human motion prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021) 10
2. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: The International Conference on Machine Learning (ICML) (2015) 9
3. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7) (2014) 9
4. ISO: ISO/TS 15066:2016. Robots and robotic devices — Collaborative robots (2021), https://www.iso.org/obp/ui/#iso:std:iso:ts:15066:ed-1:v1:en 2
5. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: The International Conference on Learning Representations (ICLR) (2015) 9
6. KUKA: LBR iiwa 14R820 User Manual (2021), https://www.oir.caltech.edu/twiki_oir/pub/Palomar/ZTF/KUKARoboticArmMaterial/Spez_LBR_iiwa_en.pdf 2
7. Magrini, E., Ferraguti, F., Ronga, A.J., Pini, F., De Luca, A., Leali, F.: Human-robot coexistence and interaction in open industrial cells. Robotics and Computer-Integrated Manufacturing **61** (2020) 5
8. Mao, W., Liu, M., Salzmann, M.: History repeats itself: Human motion prediction via motion attention. In: The European Conference on Computer Vision (ECCV) (2020) 10
9. Mao, W., Liu, M., Salzmann, M., Li, H.: Learning trajectory dependencies for human motion prediction. In: The IEEE International Conference on Computer Vision (ICCV) (2019) 10