2021 20th International Conference on Advanced Robotics (ICAR)
December 6-10, 2021. Ljubljana, Slovenia

2021 20th International Conference on Advanced Robotics (ICAR) | 978-1-6654-3684-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICAR53236.2021.9659328

Neural Networks based Human Intent Prediction for Collaborative Robotics Applications

Federico Formica[1], Stefano Vaghi[2], Niccolò Lucci[3] and Andrea M. Zanchettin[4]

Abstract— **Industry 5.0 has laid the necessity to relocate the human at the center of the manufacturing cycle, where everything should be designed to ease his/her work. This implies one to redefine the human-robot collaboration, making it not only safe, but also more inclusive for the operator. Nowadays, this goal is viable thanks to the integration of new sensors in the work cell. Coupled with advanced control strategies, they give the robot a better understanding of both the surrounding environment and the human movement, allowing a more organic cooperation. This paper exploits an RGB-D camera and Deep Learning algorithms to retrieve the 3D positions of the manipulated objects in the workspace and to infer the most likely future human destinations in a pick and place case study. Merging this information, a proper control logic is defined and tested in a real robotics application, with the final intent of minimizing human-robot collisions during the collaboration and making the process more reliable and efficient.**

I. Introduction

With the advent of Industry 5.0, the human is located at the center of the production line [1], regaining his primary role inside the shop floor.

Everything is designed, evolves and adapts to satisfy his/her needs. In collaborative robotics, this paradigm has been intensively studied not only for safety purposes, but also to accommodate and ease the task for the operator, facilitating his/her job.

With this objective in mind, the cobot should be more aware both of its surroundings and of where his/her fellow coworker is going or about to go. This last piece of information, combined with optimal decision strategies, plays a paramount role to achieve an extensive and full human-robot collaboration.

Thanks to the proliferation of economical RGB-D cameras and wearable devices like smart gloves [2], [3], the possibility of collecting data from operators and the working cell has been steadily introduced. Therefore, target detection and the prediction of future operator actions can be computed. This is of fundamental importance both for safety reasons, and for making the collaboration smoother and less error-prone. In this framework, the major challenge is understanding the working environment and the human behaviour, with the final objective of anticipating the human motion and choose the most suited response by the robot. In the State of the Art, there are several applications dealing with Human Intent Prediction (HIP) that use statistical approaches such as Hidden Markov Models (HMM) [4], [5], [6], Gaussian Mixture Models (GMM) [7], [8], [9] and Bayesian Inference [10], [11], [12].

On the other hand, in this work Deep Learning algorithms have been used both for the 3D positions estimation of objects of interest and prediction of the human future destination among a small set of possible ones. Our case study consists in a pick and place operation where the manipulated objects are apples, and both the worker and the robot need to cooperate in palletizing the targets.

This result is achieved thanks to the Kinect V2 sensor, which also allows tracking in real-time the position of a person's skeletal joints. Combining this information, we developed a control strategy that exploits these data and decides optimally which action to perform. This, paired with an underlying certified safety system, guarantees a high safety level for the workers and reduces the stops to the bare minimum, ensuring a high efficiency.

The present work contributes in increasing human-robot collaboration by giving the latter knowledge of the surroundings, focusing on the human intention, and developing a deterministic control strategy based on what is the actual best decision for the robot to avoid any kind of interruption. To prove our approach, we have tested it in a real case scenario with several participants and we have demonstrated how the Human Intent Prediction is paramount for a correct and less error-prone collaboration.

The remainder of this work is organized as follows. Section II contains a review of the State of the Art for HIP. Section III describes the HIP algorithm and the real-time control strategy. Section IV presents the implementation details as well as the results of the experimental campaign. Finally, Section V draws the conclusions.

II. State of the art

The goal of Human Intent Prediction is to estimate the most likely future action or trajectory of a human operator [13], performing a specific task. This technology is important for several fields, including autonomous driving vehicles. For example, Best *et al.* [11] model pedestrians as intelligent agents and use a Monte-Carlo sampling technique to produce a probability map of their next positions.

For collaborative robotics applications, several methods have been applied to perform such an operation, with the most common strategies being a model-based approach, statistical inference or a combination of both.

[1]Federico Formica and [2]Stefano Vaghi are with Politecnico di Milano, federico.formica@mail.polimi.it; stefano4.vaghi@mail.polimi.it

[3]Niccolò Lucci and [4]Andrea M. Zanchettin are with Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, Piazza L. Da Vinci 32, 20133, Milano, Italy. niccolo.lucci@polimi.it; andreamaria.zanchettin@polimi.it

978-1-6654-3684-7/21/$31.00 ©2021 IEEE

Lenz *et al.* [4] use a composite HMM for each hand to predict the future actions of the operator in a collaborative assembly application. On the other hand, Zanchettin *et al.* [5] highlight the main limitation of this approach as it considers only the state in the previous timestep and improve this method by employing higher order Markov chains.

Tortora *et al.* [7] combine HMM with a GMM to classify the movements and use Gaussian Mixture Regression to identify the most likely future trajectory. However, this algorithm shows a lack of robustness on trajectories never previously seen or in the case of a sudden change in the operator trajectory.

To cope with this issue, Zanchettin *et al.* [10] propose a method based on Bayesian Inference that combines a Probability Map with the minimum jerk model.

In recent years, Human Intent Prediction exploits the flexibility and computational speed of Neural Network based algorithms. In particular, Landi *et al.* [14] propose a RNN coupled with the minimum jerk model to predict the future human arm trajectory in a collaborative robotics context.

Wang *et al.* [15] couple a CNN to locate the worker hands and a RNN network, based on LSTM cells, to predict their future movements.

Finally, Nicolis *et al.* [16] propose an RNN architecture to infer the future position of the operator and, thus, the most probable target that the user wants to reach. This method has been used in a cooperative guidance task, in combination with a traditional admittance controller.

III. Human Intent Prediction and Robot Control Logic

This section contains the details of the Human Intent Prediction (HIP) algorithm and the control logic that exploits these prediction to guide the robot.

Before entering into the specifics of the strategy proposed, it is important to highlight a key point: this method has been designed for collaborative picking applications of objects placed on a planar surface. The structure of the network and the control logic are thus optimized for such a situation.

To properly define the network, the workspace has been divided into several areas, as shown in Figure 1, and each of the operator's hands occupies one of the above sectors at a time.

A. HIP Input and Outputs

Two networks have been defined to infer left and right hand movements of the operator. The network structure and the training hyperparameters are exactly the same for both hands. The difference relies on the inputs, which are related to the movements of one side of the human body and, consequently, on the outputs. The networks take as input the positions of Shoulders (Sh), Elbows (El) and Wrists (Wr) of right and left side of the human body on the form:

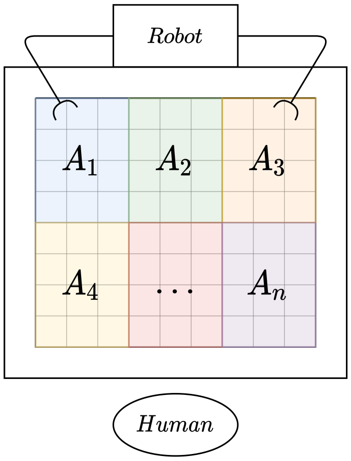

Fig. 1. Example of workspace classification in n different areas

$$
\begin{aligned}
\underline{Input}_R^{<i>} = [&Sh^{<i>}_{L\ x},\ Sh^{<i>}_{L\ y},\ Sh^{<i>}_{L\ z},\ \dots \\
&Sh^{<i>}_{R\ x},\ Sh^{<i>}_{R\ y},\ Sh^{<i>}_{R\ z},\ \dots \\
&El^{<i>}_{R\ x},\ El^{<i>}_{R\ y},\ El^{<i>}_{R\ z},\ \dots \\
&Wr^{<i>}_{R\ x},\ Wr^{<i>}_{R\ y},\ Wr^{<i>}_{R\ z}]^T
\end{aligned}
\tag{1}
$$

$$
\underline{\underline{Input}}_R^{<t>} = \begin{bmatrix} \underline{Input}_R^{<t-N_p>} \\ \dots \\ \underline{Input}_R^{<t-1>} \\ \underline{Input}_R^{<t>} \end{bmatrix}
\tag{2}
$$

where $< ... >$ indicates the time instant.

As it is possible to see from (1), both shoulders are included in the input vector, so that the network can infer their orientation, as it is a good predictor of future movements. This equation represents the input coordinates for the prediction of the right side of the human body. The complete input of the network is the union of multiple vectors containing joints positions acquired at fixed timesteps, as shown in (2). The number of rows N_p is one of the hyperparameters of the network and it represents the number of past timesteps for which the body pose is provided to the RNN.

The output of the network is a class probability, for one of the two hands, to belong to each of the n areas after a predefined number of timesteps in the future (N_f), as shown in (3) and (4). The sum of the elements on each row is equal to 1, as the elements express a probability.

$$
\underline{\underline{Output}}_R^{<t>} = \begin{bmatrix} \underline{Class}_R^{<t+1>} \\ \underline{Class}_R^{<t+2>} \\ \dots \\ \underline{Class}_R^{<t+N_f>} \end{bmatrix}
\tag{3}
$$

$$
\begin{aligned}
\underline{Class}_R^{<i>} = \ [&Prob^{<i>}_{R\in A_1},\ Prob^{<i>}_{R\in A_2}, \\
&\dots,\ Prob^{<i>}_{R\in A_n}]^T
\end{aligned}
\tag{4}
$$

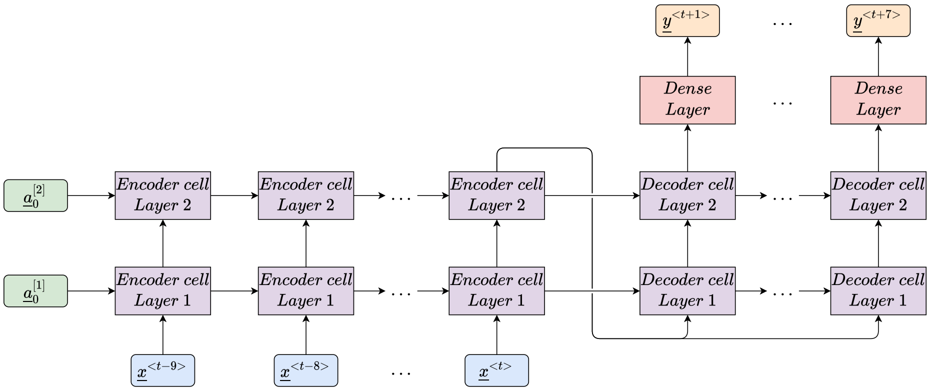

Fig. 2. Encoder-Decoder Network Structure for Human Intent Prediction.

B. *HIP Network structure*

Since inputs and outputs have different lengths, the network is based on an Encoder-Decoder architecture, which exploits the GRU [17] cells to make predictions. The structure of the network has been defined parametrically, allowing a simple description of the model through few hyperparameters. This way, an iterative algorithm can test different combinations of the hyperparameters and find the one with the best performance indicators on the specific dataset.

Since the experimental pick and place task is quick, the time window is chosen to be small, neglecting less significant data. In this case, the network uses the ten past timesteps to infer the next seven ones. Both the Encoder and Decoder are composed of two GRU layers and on top of the decoder there is a Dense layer, which provides the final probability. Figure 2 summarizes the structure of the network, while Table I details some additional hyperparameters used for the network definition and training.

TABLE I

ADDITIONAL HIP NETWORK HYPERPARAMETERS FOR TRAINING.

HYPERPARAMETER	**VALUE**
GRU units	30
Dense units	5
Dropout	10 %
Number of epochs	500
Batch size	8
Training samples	19 124
Loss function	Categorical Cross-Entropy

C. *Control Strategy*

The following control strategy has been designed to provide the robot with the most suitable target positions to avoid the operator with the objective of maximizing the collaboration fluency and reducing the collisions between the present agents to the bare minimum.

In addition to the human operator movement prediction and the target positions, the control strategy requires also the knowledge of which action the robot is performing. For the sake of simplicity, the robot trajectory has been divided in 6 phases that describe a complete picking movement:

- Phase 0: the robot moves from its current position to the approach point communicated by the control logic.
- Phase 1: the robot approaches from above the estimated position of the target centroid and closes the gripper.
- Phase 2: the robot goes back to the approach position and performs a grasping check. If it fails the robot goes back to Phase 0, otherwise it continues to Phase 3.
- Phase 3: the robot moves to the first free buffer approach position.
- Phase 4: the robot goes to the release position and the gripper opens.
- Phase 5: the robot raises the end effector and the cycle is restarted.

Also two additional statuses have been defined. Phase 6 is when the robot is in home position and Phase 7 is used to notify that the buffer is full. Figure 3 summarizes how the phases are connected and how the robot would act without a human worker.

One of the most crucial element is deciding whether a hand is going to enter the collaborative area or not.

The output of each HIP network is summarized in a single label. This procedure is carried out by comparing the number of occurrences of collaborative and non-collaborative areas. This means that the output prediction vector is squeezed in a single label that summarizes the operator intent. Such squeezing is performed by considering the number of occurrences of collaborative and non-collaborative areas: in case two or less of the seven predicted labels belong to collaborative area then a non-collaborative output is predicted. Otherwise, the hand destination is estimated as the collaborative area with the higher frequency in the output prediction.

Therefore, the control strategy requires the following information at each time step:

- List of available targets in the collaborative area
- Current pose of the human hands
- Squeezed prediction for each human hand
- Current status for each robot arm

To avoid needless corrective actions, the human presence is

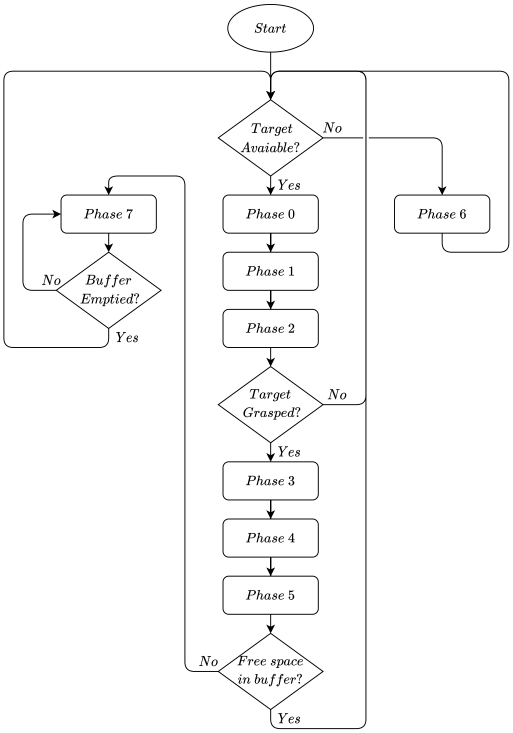

Fig. 3. This chart represents the baseline of the robot cyclic motion, neglecting the human presence. It can be modified if the control logic deems a corrective action as necessary, as will be explained below.

considered only when the hands velocity are directed toward the collaborative region and they are above a threshold or the current human position is within a fixed distance from the collaborative regions. This allows the robot to operate as non-collaborative if the human worker is performing some tasks unrelated to the picking motion.

Besides the target positions, the control strategy evaluates, if necessary, the most suitable corrective action for each robot arm. These are coded in 6 status flags:

- GO: neutral flag code, the robot arm continues its current motion path.
- HOME: interrupts the current motion and go home.
- SLOW: reduces the maximum velocity of the end effector.
- SWITCH: interrupts the current motion and moves to the target position just received.
- AVOID: interrupts the current motion, performs an evasion movement and moves to the target position just received.
- STOP: interrupts the current motion until a different flag is received, then the previous trajectory is resumed.

IV. Experimental validation

In order to analyze the performances of the Human Intent Prediction algorithm, we have tested it on a real collaborative robotics application.

This section contains a description of the application, how the HIP dataset and Object Detection algorithm are structured and the results obtained.

A. Experimental setup and implementation details

The robotic application is a collaborative picking task, where a robot and a human worker have to cooperate to pick and place in buffers a set of targets randomly arranged in the workspace.

The robot used in these tests is an ABB YuMi dual-arm robot, equipped with a custom two-fingers gripper. The vision system is the Smart Robots device which embeds a Kinect V2 that provides both a RGB image, on which the Object Detection algorithm localizes targets, and the 3D positions of the operator joints.

For this application, the workspace has been divided into five areas, three of them (areas A, B and C) are collaborative areas, while the other two (areas D and E) are not. The targets used for the experiments are apples, since their spherical shape makes it possible to grasp them from any direction. As shown in Figure 4, there are 3 buffers in which they can be placed: one for each robot arm and one for the operator. Given the limited space, the robot buffers can contain only three apples each, after which the robot must stop. A reset button is also connected to the robot to notify it that the buffers have been emptied and that the picking cycle can restart.

The use case is defined as the picking of 10 apples, with the worker and robot starting at the same time. When all the targets are placed inside one of the buffers, the experiment is finished, so the operator can move the apples back to the collaborative area and press the reset button, as the buffers are empty again.

The control strategy, at each timestep, chooses a target belonging to one of the areas that is not occupied or going to be occupied by the human worker. Since the B zone is between the two robot arms, it is split in two: BR (B Right) and BL (B Left), as shown in Figure 4. This gives each robot arm two areas from which it can pick targets. As a result, even if the human is occupying two collaborative areas in the same moment, at least one of the robot arms is still able to perform the picking operation.

The application has been defined in a way that leaves to the worker complete freedom in choosing the pace, how many targets to pick and which hand to use. Since the picking is asynchronous, the robot can take advantage of its full speed and minimize the time required to complete the task.

B. Object Detection

This section defines the computer vision algorithm chosen for this work. The *YOLOv3* from Redmon *et al.* [18] represents a state-of-the-art Convolutional Neural Network (CNN) based Object Detection algorithm, and it has been used to carry out the detection task.

The network structure used is the Tiny-version of the algorithm, which is characterized by 23 layers. The slenderness of the network allows to detect targets in real-time with an high frame rate and a sufficient degree of accuracy.

After a data augmentation procedure (mirroring, cropping,

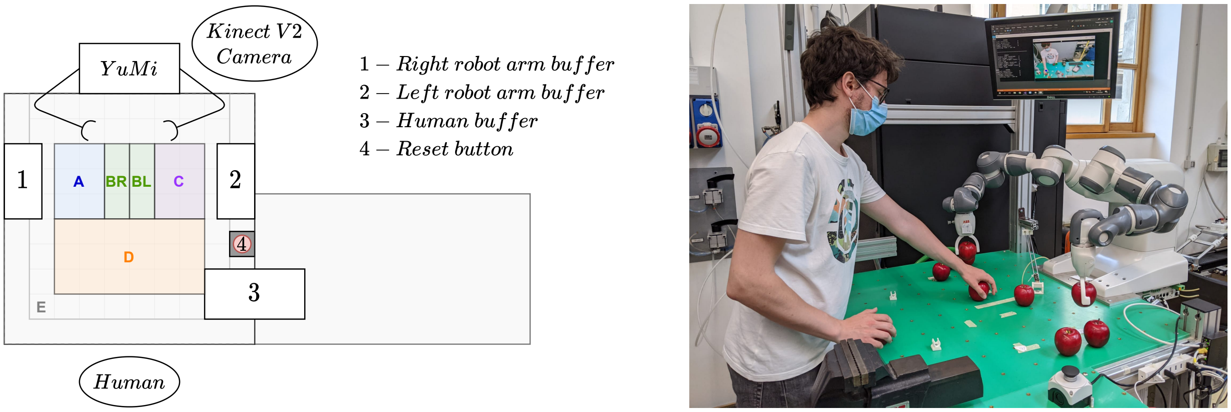

Fig. 4. On the left side, a schematic view of the complete work cell is represented. As it is possible to infer from the figure, several areas are defined: A, BR, BL and C belong to the collaborative regions, D and E are regions not reachable by the robot. On the right side, it is depicted the robotic cell used for the experimental validation.

brightness and color shift), the complete training and validation dataset is composed of ∼1000 images representing the target to be detected. The starting dataset contains both images taken in the robotic cell and with other backgrounds to improve the generality of the system. In order to minimize the computational power needed to train the CNN, the transfer learning [19] technique has been used, imposing the initial weights obtained by Redmon *et al.* [18] on the ImageNet 1000.
The depth map provided by the Kinect V2 allows to retrieve the 3D position of the targets centroid starting from the 2D bounding boxes parameters.

Both YOLO and HIP algorithms require significant computational power. To use them in a real-time application with fast movements, such as in this picking task, the YOLO Neural Network has been run partly on the GPU thanks to the CUDA API. These two algorithms are run on a laptop using Intel Core i7-8550U CPU processor and NVIDIA GeForce 940MX graphic card, reaching a refresh rate of approximately 6 FPS.

C. HIP Dataset Generation

Body joints data are acquired with a Kinect V2 sensor. To enhance the generalization of the dataset, ten participants have been recorded while picking objects from the workspace. The movements have been acquired with a fixed sampling time of $\Delta t = 150\ ms$. The final dataset contains 74000 timesteps, corresponding approximately to 3 hours of recording.
The dataset is generated using a *Sliding-Window Algorithm*: the window considers 17 timesteps at a time, saving it as a sample.
Data are then normalized with a value between $[0.1 - 0.9]$ to speed up the training procedure. The last operation performed on the dataset regards the generation of output values: the wrists positions are classified in a *one-hot array*, whose values represent the belonging of the wrist to one of the 5 areas of the workspace.

D. Experimental Results

In order to compare our algorithm to a baseline, the use case has been repeated with three different control strategies: neglecting the human worker using only the Object Detection algorithm (YOLO), coupling Object Detection with the control logic that considers only the actual human position - Human Tracking - (YOLO+HT) and, finally, the complete framework which includes Object Detection and Human Intent Prediction (YOLO+HIP)[1]. The considered KPIs are the number of safety stops due to collisions (both with the operator and the environment) and the cycle time of the task. To have significant results, 16 participants have been involved in the experimental campaign, each of whom has performed all the three experiments four times.

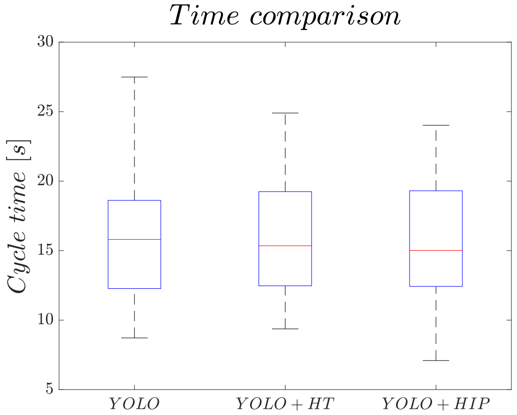

Fig. 5. Box Plot of the cycle times for the three control strategies. The median value slightly decreases by increasing the complexity of the control strategy.

The experimental campaign shows that the proposed strategy (YOLO+HIP) reduces the number of collision by 38 % with respect to the sole tracking (YOLO+HT), or by 70 %

[1] The video showing the experimental results is available **here** (https://www.youtube.com/watch?v=kOMk_3P5MbU).

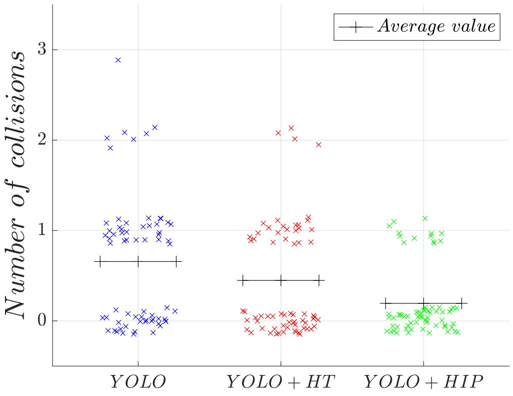

Fig. 6. Plot representing the dispersion of collisions around the average value. The number of collisions per cycle drastically decreases with the proposed approach.

compared to only Object Detection (YOLO). In addition, the average cycle time does not increase since the robot path, instead of being restored after a collision, is modified to perform corrective actions. Figures 5 and 6 summarize the results obtained.

A Wilcoxon Rank Sum test has been performed to analyse the distribution of the number of collisions. Each of the three strategies has been compared with the other two, confirming the decreasing trend of the number of impact with the proposed strategy. The results show that:

- YOLO+HT has a median number of collisions lower than YOLO only with a p-value of 0.038.
- YOLO+HIP has a median number of collisions lower than YOLO only with a p-value of 8.7×10^{-7}.
- YOLO+HIP has a median number of collisions lower than YOLO+HT with a p-value of 0.0048.

V. Conclusion

In this work, a novel Neural Network based framework for Object Detection and Human Intent Prediction is proposed. YOLOv3 algorithm is used for targets localization in the input image. The 3D position of the targets is retrieved exploiting the depth map provided by the Kinect V2 sensor. The camera is also employed for tracking the human operator; this allow to define a Neural Network for Human Intent Prediction taking the body joints position in several past timesteps as input. A proper control strategy is defined taking into account the outputs of both networks together with the robot status, to provide the most suitable target positions and optimal corrective actions.

This framework requires an accurate tracking of the operator body joints. The functions of Kinect SDK are not always robust on tracking, specifically on determining the limbs length. This introduces a significant level of uncertainties in the measurements. A more refined tracking algorithm, or exploiting wearable devices, will likely improve the performance of the HIP algorithm.

Further development of this work include an additional training phase at the end of each work session, in order to fine tune the HIP network on the movements of a single operator. This should improve considerably the accuracy of the HIP algorithm predictions.

Acknowledgment

This work was supported by Progetto Prin 2017 “TIGHT: Tactile InteGration for Humans and arTificial systems”, prot. 2017SB48FP.

References

[1] S. Nahavandi, “Industry 5.0—a human-centric solution,” *Sustainability*, vol. 11, no. 16, 2019.

[2] W. Chen, C. Yu, C. Tu, Z. Lyu, J. Tang, S. Ou, Y. Fu, and Z. Xue, “A survey on hand pose estimation with wearable sensors and computer-vision-based methods,” *Sensors*, vol. 20, no. 4, 2020.

[3] C. Gkournelos, P. Karagiannis, N. Kousi, G. Michalos, S. Koukas, and S. Makris, “Application of wearable devices for supporting operators in human-robot cooperative assembly tasks,” *Procedia CIRP*, vol. 76, pp. 177–182, 2018. 7th CIRP Conference on Assembly Technologies and Systems (CATS 2018).

[4] C. Lenz, A. Sotzek, T. Röder, H. Radrich, A. Knoll, M. Huber, and S. Glasauer, “Human workflow analysis using 3d occupancy grid hand tracking in a human-robot collaboration scenario,” pp. 3375–3380, 2011.

[5] A. M. Zanchettin, A. Casalino, L. Piroddi, and P. Rocco, “Prediction of human activity patterns for human-robot collaborative assembly tasks,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 3934–3942, 2019.

[6] M. Wu, T. Louw, M. Lahijanian, W. Ruan, X. Huang, N. Merat, and M. Kwiatkowska, “Gaze-based intention anticipation over driving manoeuvres in semi-autonomous vehicles,” pp. 6210–6216, 2019.

[7] S. Tortora, S. Michieletto, F. Stival, and E. Menegatti, “Fast human motion prediction for human-robot collaboration with wearable interface,” pp. 457–462, 2019.

[8] J. Kang, K. Jia, F. Xu, F. Zou, Y. Zhang, and H. Ren, “Real-time human motion estimation for human robot collaboration,” in *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 552–557, 2018.

[9] P. Zhang, P. Jin, G. Du, and X. Liu, “Ensuring safety in human-robot coexisting environment based on two-level protection,” *Industrial Robot*, vol. 43, no. 3, pp. 264–273, 2016.

[10] A. M. Zanchettin and P. Rocco, “Probabilistic inference of human arm reaching target for effective human-robot collaboration,” vol. 2017-September, pp. 6595–6600, 2017.

[11] G. Best and R. Fitch, “Bayesian intention inference for trajectory prediction with an unknown goal destination,” vol. 2015-December, pp. 5817–5823, 2015.

[12] S. Pellegrinelli, H. Admoni, S. Javdani, and S. Srinivasa, “Human-robot shared workspace collaboration via hindsight optimization,” vol. 2016-November, pp. 831–838, 2016.

[13] Y. Kong and Y. Fu, “Human action recognition and prediction: A survey,” 2018.

[14] C. Landi, Y. Cheng, F. Ferraguti, M. Bonfe, C. Secchi, and M. Tomizuka, “Prediction of human arm target for robot reaching movements,” pp. 5950–5957, 2019.

[15] Y. Wang, X. Ye, Y. Yang, and W. Zhang, “Collision-free trajectory planning in human-robot interaction through hand movement prediction from vision,” pp. 305–310, 2017.

[16] D. Nicolis, A. M. Zanchettin, and P. Rocco, “Human intention estimation based on neural networks for enhanced collaboration with robots,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1326–1333, 2018.

[17] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” 2014.

[18] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” 2018.

[19] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.