CIRP Annals - Manufacturing Technology 69 (2020) 9–12

Contents lists available at ScienceDirect

CIRP Annals - Manufacturing Technology

journal homepage: https://www.editorialmanager.com/CIRP/default.aspx

Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly

Jianjing Zhang[a], Hongyi Liu[b], Qing Chang[c], Lihui Wang (1)[b], Robert X. Gao (1)[a,*]

[a] Department of Mechanical and Aerospace Engineering, Case Western Reserve University, Cleveland, OH, United States
[b] Department of Production Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
[c] Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, VA, United States

ARTICLE INFO

Article history:
Available online 20 May 2020

Keywords:
Assembly
Motion
Machine learning

ABSTRACT

Effective and safe human-robot collaboration in assembly requires accurate prediction of human motion trajectory, given a sequence of past observations such that a robot can proactively provide assistance to improve operation efficiency while avoiding collision. This paper presents a deep learning-based method to parse visual observations of human actions in an assembly setting, and forecast the human operator's future motion trajectory for online robot action planning and execution. The method is built upon a recurrent neural network (RNN) that can learn the time-dependent mechanisms underlying the human motions. The effectiveness of the developed method is demonstrated for an engine assembly.

1. Introduction

Humans have long been the sources of flexibility and versatility in manufacturing, performing varied tasks under the dynamic conditions based on contextual understandings. While robots entered the workforce nearly 40 years ago, they are primarily pre-programmed to perform both the routine and ergonomically challenging tasks. As such, robots are not intended for use in operations that require decision-making and adjustment using the real-time input from the target process. As the 4th industrial revolution transforms the factory into a more dynamic and smart production space, there is an increasing need for breaking the barrier between the human and the robot, and allowing the joint activity in a shared workspace in order to accomplish a set of given tasks more effectively and efficiently through a highly integrated human-robot collaboration (HRC) [1,2].

Effective HRC consists of four basic elements: human motion perception, recognition, trajectory prediction and robot action [2], as shown in Fig. 1. Recent advances in sensing and machine learning algorithms have improved the state of research in human motion recognition, providing the basis for human motion prediction. In [3], a hidden Markov model (HMM) has been investigated to analyze the human motion during assembly and recognize the incorrect actions. In [4], a deep convolutional neural network has been developed to detect the human action-related motion pattern. Deep learning has also shown to help the robot to improve the recognition accuracy of actions from the human worker by analyzing the body motion [5].

Adaptive robot control has been another active research field in HRC, which allows the robots to adaptively maneuver in the workspace, avoid collision and execute manufacturing tasks [2]. Real-time

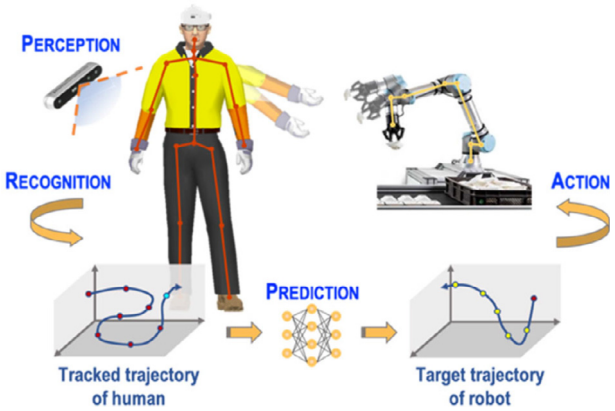

Fig. 1. Elements of HRC, robot image from [9].

sensor-driven and model-based robot control have shown to reduce the communication traffic and significantly improve the robot controllability [6]. In [7], a depth sensor-based system has been developed to enable the robot to detect potential collision with a human worker and respond accordingly. Behavior control in cognitive robotic disassembly has been developed in [8], which permit robots to learn to become fully autonomous after several iterations of disassembling the same product models.

In comparison, human motion prediction, which provides the information on where a human worker will likely move to in the future, has not been widely reported in the published literature. Accurate and robust prediction of human motion trajectory provides the guidance for a robot to act proactively to accommodate human action, thereby enhancing the robot's flexibility, efficiency and safety in manufacturing [2]. As a deep learning architecture specialized in

* Corresponding author.
E-mail addresses: robert.gao@case.edu, gao@ieee.org (R.X. Gao).

https://doi.org/10.1016/j.cirp.2020.04.077
0007-8506/

sequential pattern analysis [10], RNN has demonstrated effectiveness in a variety of manufacturing applications, such as prognosis of structural defect propagation based on vibration data [11]. For human trajectory prediction, an encoder-decoder RNN structure has been reported [12], where the encoder analyzes the past trajectory and the decoder progressively predicts the future path. The structure was further enhanced [13] with residual connections that exploit the first-order derivatives to minimize jump in the predicted trajectory. One limitation of the prior studies is that the interactive patterns among the body parts that can potentially help improve the prediction accuracy, have not been exploited. Another limitation is that uncertainty associated with human workers in trajectory prediction was not accounted for, which however is critical to ensure reliability and safety in HRC. Leveraging the abilities of deep learning, this paper presents an RNN-based model for human motion trajectory prediction. The developed RNN model accounts for the interactions among human body parts, and provides an uncertainty estimation for robust trajectory prediction. The developed model is experimentally evaluated using the collaborative assembly of an automotive engine, and good results have been demonstrated.

2. RNN for motion trajectory prediction

The key to accurate motion trajectory prediction is to capture characteristics of the evolution patterns of the human poses when performing actions that are represented by body joints. RNNs provide the technical foundation for analyzing such patterns.

2.1. RNN for HRC in assembly

RNNs capture the motion evolution patterns by analyzing the *influence* of the motion state at each of the time steps on the state at the subsequent step. Motion state can be considered a high-level feature of the motion pattern condensed from the raw observation of the human poses.

The state at time step n, h_n, depends on both the associated observation x_n and the preceding state h_{n-1}, expressed as:

$$h_n = \varphi(W_{h\text{-}h} h_{n-1} + W_{x\text{-}h} x_n + b) \quad (1)$$

where $W_{h\text{-}h}$ denotes the weights representing the influence of the preceding state on the current state, $W_{x\text{-}h}$ represents the weights connecting the observation to the state, b is the bias, and φ is a nonlinear function. As h_{n-1} also depends on its preceding state, the generation of h_n makes use of all the available observations up to time n, as shown in Fig. 2. The motion evolution pattern is captured through the network weights.

The standard RNN structure can be ineffective in motion prediction due to the interaction among different body parts (e.g. arm, leg, and spine). As a solution, two types of functional units have been added to the network structure: *component* and *coordination*. The *component* unit analyzes the trajectory of a specific body part associated with a certain human pose, and the *coordination* unit analyzes the interaction among the body parts. A total of five *component* (for two arms, two legs and one spine) and four *coordination* (for arm-arm, arm-spine, leg-leg, leg-spine) units were added to model a complete human body.

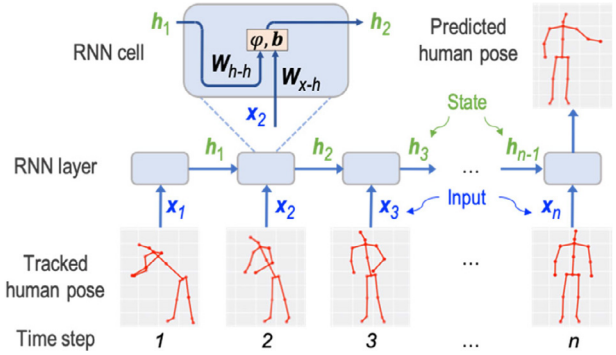

Fig. 2. Overview of RNN for motion prediction.

The network structure of each functional unit is based on the RNN structure in Fig. 2, where the input to each component unit is the sequence of the past observations of the related body joints locations in the x–y–z coordinates. For example, for the "arm" unit, the related body joints are wrist, elbow and shoulder. In addition, the output from the coordination units that are associated with the corresponding body parts is also used as the input to the component unit. For example, the "arm" unit takes as the input the output from the "arm-arm" and "arm-spine" units. Similarly, the input to each coordination unit is the concatenation of body joints locations associated with the interacting body parts. As an example, the input to the "arm-spine" unit is the concatenation of the information on the joints from both the arm and the spine. The RNN structure with functional units is trained in a collective manner by the Backpropagation algorithm. In Fig. 3, a sample structure of the enhanced RNN is illustrated.

Such an enhanced network structure allows for a more accurate prediction of the human motion trajectory in the context of both its own history and the interaction among the various body parts, resulting in reduced prediction error. Specifically, the mean deviation between the predicted and actual body joints locations has been reduced by nearly 40% when compared to the RNN structure without the functional units.

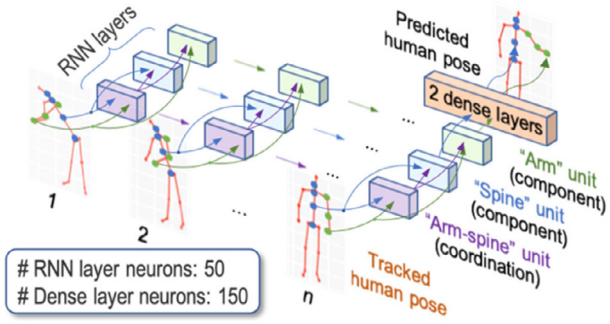

Fig. 3. Enhanced RNN for improved accuracy in motion prediction (three functional units shown: arm, spine and arm-spine).

2.2. Monte-Carlo dropout for uncertainty estimation

Human motion exhibits uncertainties. For examples, when a worker moves an arm slightly, it is difficult for the prediction algorithm to determine whether the person intends to extend the arm, or it is just the natural movement while maintaining a pose. On the other hand, if the arm moves further away, it becomes less uncertain that his intention must be to extend the arm. To account for the uncertainty, probabilistic inference has been incorporated into the motion prediction algorithm, which is formulated as determining the conditional probability distribution $p(y|X_n)$, where $X_n = x_1, x_2, \ldots, x_n$ represents the evolving human pose up to time n, and y is the future human pose.

To enable the varying weights selection for estimating $p(y|X_n)$ after network training, the method of Monte-Carlo (MC) dropout has been investigated where the network weights were selected randomly by using the Bernoulli distribution [14]. Accordingly, $p(y|X_n)$ is approximated by performing multiple predictions and averaging the result to arrive at the expected future human pose:

$$\mathbb{E}_{p(y|X_n)}(\boldsymbol{y}) \approx \frac{1}{K}\sum_{k=1}^{K} \hat{\boldsymbol{y}}(X_n,\ \boldsymbol{W}_{\text{dropout}}) \quad (2)$$

where $\hat{y}$ represents the individual predictions, and the network weights after dropout is expressed as:

$$W_{\text{dropout}} = W \cdot \text{diag}([z]),\ z \sim \text{Bernoulli}(q) \quad (3)$$

where q represents the dropout rate. Statistically, such a probabilistic approach to obtaining $p(y|X_n)$ has the advantage of improving the accuracy in human pose prediction and reducing the uncertainty-induced mis-triggering of the robot action.

3. Experimental evaluation

To evaluate the performance of the developed method, human-robot collaborative assembly of a car engine was conducted. Assembly often involves repeated actions that are ergonomically challenging, such as moving large and heavy parts, and therefore has been identified as the main area to benefit from HRC [2]. The engine assembly (Fig. 4) consists of a large cover, a cover cap, two wire collectors, four plugs, a screwdriver, and eleven screws. The assembly workspace is shown in Fig. 5, where the engine is placed to the right of the worker and an UR-5 robot is installed on a workstation to his left. Parts and tools are sorted in color-coded containers. The human pose is tracked by a Kinect sensor at 30 Hz, serving as the basis for real-time human motion trajectory prediction. The collaboration between the human and the robot is presented by the robot responding to the predicted human motion trajectory, following in real time the human hand as it extends for handover and picking up the relevant part/tool during installation.

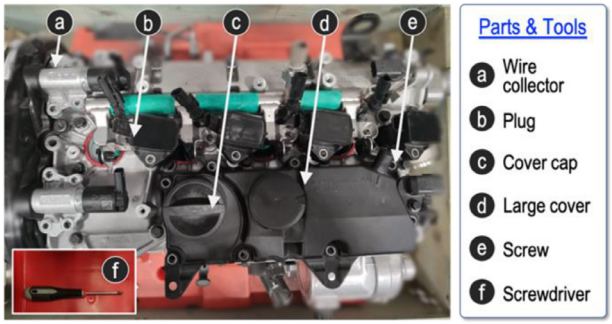

Fig. 4. Engine assembly parts and tools.

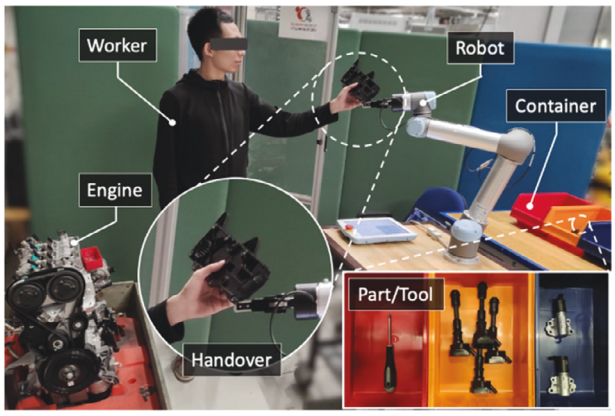

Fig. 5. Elements in engine assembly workspace.

The objective of predictive modeling of human motion trajectory is to: (1) predict the end location of a motion trajectory at each time step, and (2) evaluate the transition probability to determine if a transition is to occur such that the robot's proactive motion should be triggered. Three pose *nodes* have been defined: *handover* (n_1), *standing* (n_2) and *installation* (n_3). The entire engine assembly sequence can be represented as a transition graph (Fig. 6), where the motion trajectories start from and end at one of the nodes. For example, when the worker completes *handover* by taking the screwdriver from the robot, he may briefly pause by *standing* or move directly to *installation*. For flexibility, the duration of stay at each node is unspecified (i.e. worker can stay in any node for as long as needed) and is implemented by the "self-transition".

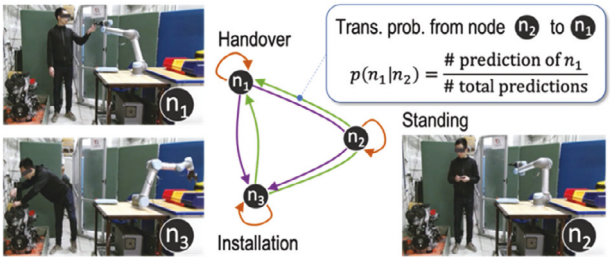

Fig. 6. Transition graph among handover, standing and installation.

To train the RNN network, motion trajectories of the worker from ten assembly operations were recorded by a Kinect sensor. Relative coordinates are used to account for the variations in the human position relative to the Kinect. For a tracked joint (e.g. wrist), its x–y–z coordinates are evaluated relative to its parent joint (e.g. elbow). The network *input* length (i.e. number of previous time steps to analyze at each frame) is set to 30 frames, corresponding to 1 s of motion. The network *output*, which represent the trajectory's end location, is sampled from all the poses during the incoming stay at that pose node. For example, when the worker is moving from *handover* to *installation*, the network output is sampled from all the poses during the incoming installation. This strategy allows the network to explore more possible locations and enhance its robustness. The dataset consists of 8206 training samples and are randomly split into the training and validation sets, with a ratio of 7:3. The mean squared error (MSE) is used as the loss function to quantify the mean deviation between the predicted and actual human body joints locations. The network parameters are determined through cross validation.

The benefit of the enhanced RNN structure with two types of additional functional units is comparatively evaluated against an RNN without the functional units and a multi-layer perceptron (MLP). As shown in Fig. 7, the enhanced RNN has the lowest error of 16.5 mm for motion trajectory end location prediction. This corresponds to an error reduction of 44% and 63% from the RNN without the functional units and the MLP, respectively.

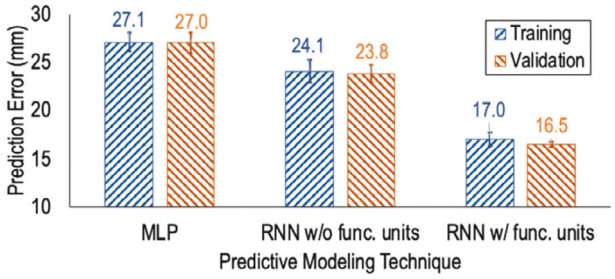

Fig. 7. Comparison of prediction error of different network structures.

The trained network is evaluated in real-time assembly operations, where the model continuously predicts the end location of the human body, as soon as new data from the Kinect sensor is provided. Through MC dropout, the distribution of predicted future locations of the individual body joint is obtained, and the mean value serves as the final predicted location. The transition probability, denoted as $p(n_i|n_j)$, $i, j = 1,2,3$ and shown in Fig. 6, determines if a transition is starting to occur or not. To minimize robot mis-triggering and ensure reliable HRC, the transition is determined to start only when the condition of $p(n_i|n_j) = 1$ is met to trigger the robot. Physically, this means that the worker will start, with 100% certainty, the transition from the action node n_i (e.g., installing) to the action node n_j (e.g., handover). A total of five predictions were made at each time step using MC dropout. The node transition probability is evaluated as the *ratio* of the number of predictions that belong to the target *node* to the total number of predictions (i.e., five). The transition is predicted to take place only if all the five results are consistent (i.e., 100% transition probability). Part/tool pickup is triggered when the *actual* human

pose is in the *installation* node, and *handover* is triggered when the transition to the handover node occurs from the *installation* or *standing* node. As the arm moves and prediction of the human hand's end location is updated, the robot adjusts the target location of its end-effector accordingly.

4. Results discussion

The collaborative assembly based on motion trajectory prediction is illustrated in Fig. 8, using the *pickup* and *handover* of a screwdriver as an example. The *actual* human pose tracked by Kinect is shown in red and its *predicted* end location in cyan, with the worker's left arm highlighted in thicker lines. The actual and target locations of the robot are in pink and yellow, respectively.

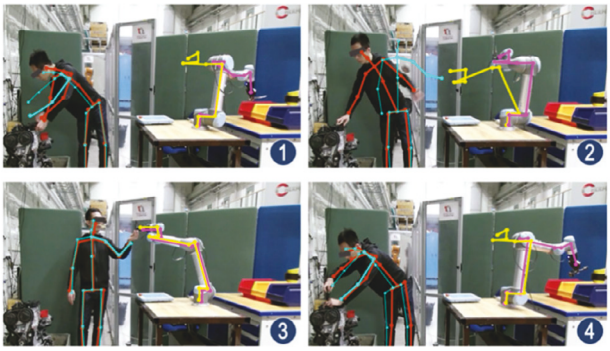

Fig. 8. Sample frames from HRC assembly sequence.

At step ①, the robot is triggered to pick up the screwdriver when the worker is detected as installing the large cover. As the worker completes the installation and starts to extend the left arm in ②, the prediction changes to the handover node. The robot is triggered, and its first target location for handover is generated. As more trajectory information is provided to the network, prediction accuracy improves, as seen in the reduced distance between the prediction and the actual hand's end location until the arm stops for handover in ③. In ④ the worker returns to the engine and the robot is triggered to pick up the next part.

The effectiveness of uncertainty handling in motion prediction is further evaluated. Fig. 9 illustrates three nearby time steps around the transition from *installation* to *handover*. All the five predicted scenarios of the worker's left arm using MC dropout are shown. In ⓐ, the worker is installing a plug and all predictions indicate that he will continue with the *installation*. In ⓑ, one of the predictions indicates that the worker is moving to handover. With remaining predictions indicating a standing pose, the uncertainty is large (as the transition probability is only 20%), therefore the robot is not triggered. In ⓒ, all predictions show that the worker is moving for *handover*, thus the robot will be triggered. It is noted that these results are consistent with the human judgement. For example, in ⓑ, the worker is retrieving the arm from the engine. At this moment, it is difficult to determine whether the movement to handover will indeed happen or the worker will stop at standing position. Correspondingly, large uncertainty is given by the model. As the arm moves further away in ⓒ, the ambiguity is reduced.

Fig. 9. Uncertainty estimation during a transition sequence.

The MC dropout provides an effective measure for uncertainty handling and mis-trigger prevention, allowing for robust HRC. In comparison, if only one prediction is made at each frame without MC dropout, the robot has shown to be mis-triggered one out of four times.

Adding the two types of functional units to the RNN structure makes the execution of the Algorithm 2 and 3 times long to finish one prediction cycle, from 0.01 s to 0.02–0.03 s. This however does not affect the real-time capability when using a Kinect with 30 Hz framerate (or 0.033 s processing time).

The developed method is extendable beyond the current experimental setup, since the RNN structure with additional functional units is independent of the experimental conditions. Regardless of the assembly actions, the input will be the sequential observations of the body joints locations. Furthermore, using body joints locations as the input makes the algorithm insensitive to variations in the appearance of the worker and the workspace.

For real assembly lines, additional factors should be considered to ensure robustness of the developed method. For example, multiple cameras should be deployed to track the worker's motion continually and avoid occultation in the workspace. In addition, parallel computing methods should be leveraged to enable the simultaneous trajectory prediction of multiple workers.

5. Conclusions

An RNN-based method has been developed for predicting human motion trajectory, with the aim of bridging the gap between human motion recognition and corresponding robot action in order to realize true HRC. A novel feature of the developed method is that it introduces two types of functional units into the RNN structure to parse the evolutionary motion pattern of the human body parts as well as their coordination for improved prediction accuracy. Furthermore, probabilistic inference based on Monte-Carlo dropout has been investigated to minimize the uncertainty-induced robot mis-trigger and enhance the reliability in interpreting the human motion. A 40% reduction in the prediction error is demonstrated with the enhanced RNN structure as compared to standard RNN. Future effort will be directed to predictive modeling that is capable of handling variations during assembly order to further advance HRC for broad acceptance and production-level implementation.

Acknowledgement

Funding provided by the US National Science Foundation under award CMMI-1830295 is sincerely appreciated.

References

[1] Krüger J, Lien TK, Verl A (2009) Cooperation of Human and Machines in Assembly Lines. *CIRP Annals* 58(2):628–646.
[2] Wang L, Gao R, Váncza J, Krüger J, Wang XV, Makris S, Chryssolouris G (2019) Symbiotic Human-Robot Collaborative Assembly. *CIRP Annals* 68(2):701–726.
[3] Urgo M, Tarabini M, Tolio T (2019) A Human Modelling and Monitoring Approach to Support the Execution of Manufacturing Operations. *CIRP Annals* 68(1):5–8.
[4] Wang P, Liu H, Wang L, Gao RX (2018) Deep Learning-Based Human Motion Recognition for Predictive Context-Aware Human-Robot Collaboration. *CIRP Annals* 67(1):17–20.
[5] Liu H, Fang T, Zhou T, Wang L (2018) Towards Robust Human-Robot Collaborative Manufacturing: Multimodal Fusion. *IEEE Access* 6:74762–74771.
[6] Kiang CT, Spowage A, Yoong CK (2015) Review of Control and Sensor System of Flexible Manipulator. *Journal of Intelligent and Robotic Systems Theory and Application* 77(1):187–213.
[7] Schmidt B, Wang L (2014) Depth Camera Based Collision Avoidance via Active Robot Control. *Journal of Manufacturing Systems* 33(4):711–718.
[8] Vongbunyong S, Kara S, Pagnucco M (2015) Learning and Revision in Cognitive Robotics Disassembly Automation. *Robotics and Computer-Integrated Manufacturin* 34:79–94.
[9] Universal robots, https://www.universal-robots.com/.
[10] Lecun Y, Bengio Y, Hinton G (2015) Deep Learning. *Nature* 521(7553):436–444.
[11] Malhi A, Yan R, Gao RX (2011) Prognosis of Defect Propagation Based on Recurrent Neural Networks. *IEEE Transactions on Instrumentation and Measurement* 60 (3):703–711.
[12] Fragkiadaki K, Levine S, Felsen P, Malik J (2015) Recurrent Network Models for Human Dynamics. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 4346–4354.
[13] Martinez J, Black MJ, Romero J (2017) On Human Motion Prediction Using Recurrent Neural Networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4674–4683.
[14] Gal Y, Ghahramani Z (2016) Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In: *Proceedings of the Thirty-thirrd International Conference on Machine Learning*, 1651–1660.