Proceedings of the ASME 2024
International Design Engineering Technical Conferences and
Computers and Information in Engineering Conference
IDETC-CIE2024
August 25-28, 2024, Washington, DC

DETC2024-143753

MULTI-TASK LEARNING FOR INTENTION AND TRAJECTORY PREDICTION IN HUMAN-ROBOT COLLABORATIVE DISASSEMBLY TASKS

Xinyao Zhang
Environmental Engineering Sciences
University of Florida, Gainesville, FL, 32611
xinyaozhang@ufl.edu

Sibo Tian
Department of Mechanical Engineering
Texas A&M University, College Station, TX, 77840
sibotian@tamu.edu

Xiao Liang
Department of Civil & Environmental Engineering
Texas A&M University
College Station, TX, 77840
xliang@tamu.edu

Minghui Zheng
Department of Mechanical Engineering
Texas A&M University
College Station, TX, 77840
mhzheng@tamu.edu

Sara Behdad*
Environmental Engineering Sciences
University of Florida
Gainesville, FL, 32611
sarabehdad@ufl.edu

ABSTRACT

Human-robot collaboration (HRC) has become an integral element of many industries, including manufacturing. A fundamental requirement for safe HRC is to understand and predict human intentions and trajectories, especially when humans and robots operate in close proximity. However, predicting both human intention and trajectory components simultaneously remains a research gap. In this paper, we have developed a multi-task learning (MTL) framework designed for HRC, which processes motion data from both human and robot trajectories. The first task predicts human trajectories, focusing on reconstructing the motion sequences. The second task employs supervised learning, specifically a Support Vector Machine (SVM), to predict human intention based on the latent representation. In addition, an unsupervised learning method, Hidden Markov Model (HMM), is utilized for human intention prediction that offers a different approach to decoding the latent features. The proposed framework uses MTL to understand human behavior in complex manufacturing environments. The novelty of the work includes the use of a latent representation to capture temporal dynamics in human motion sequences and a comparative analysis of various encoder architectures. We validate our framework through a case study focused on a HRC disassembly desktop task. The findings confirm the system's capability to accurately predict both human intentions and trajectories.

Keywords: Human intent prediction, human trajectory prediction, multi-task learning (MTL), human-robot collaboration (HRC)

1. INTRODUCTION

Human-robot collaboration (HRC) is a rapidly growing area of research and application. It plays an important role in various sectors including, but not limited to manufacturing. Understanding and predicting human intentions is a critical aspect of the successful implementation of HRC. It equips robots with the ability to interpret and respond to their human counterparts in a timely manner and promotes practical collaboration. In environments where humans and robots work in close proximity, predicting human trajectories improves safety and productivity. A major challenge in implementing HRC is the perception, prediction, and understanding of human intentions and trajectories simultaneously.

Though significant progress has been made in the domains of human intention and trajectory prediction, a gap remains in the simultaneous prediction of both aspects [1,2]. Multi-task learning (MTL), an approach where multiple related tasks are learned at the same time, presents a promising solution. Through learning to predict intentions and trajectories concurrently, robots can better understand human actions and movements, which leads to practical collaboration within the manufacturing environments.

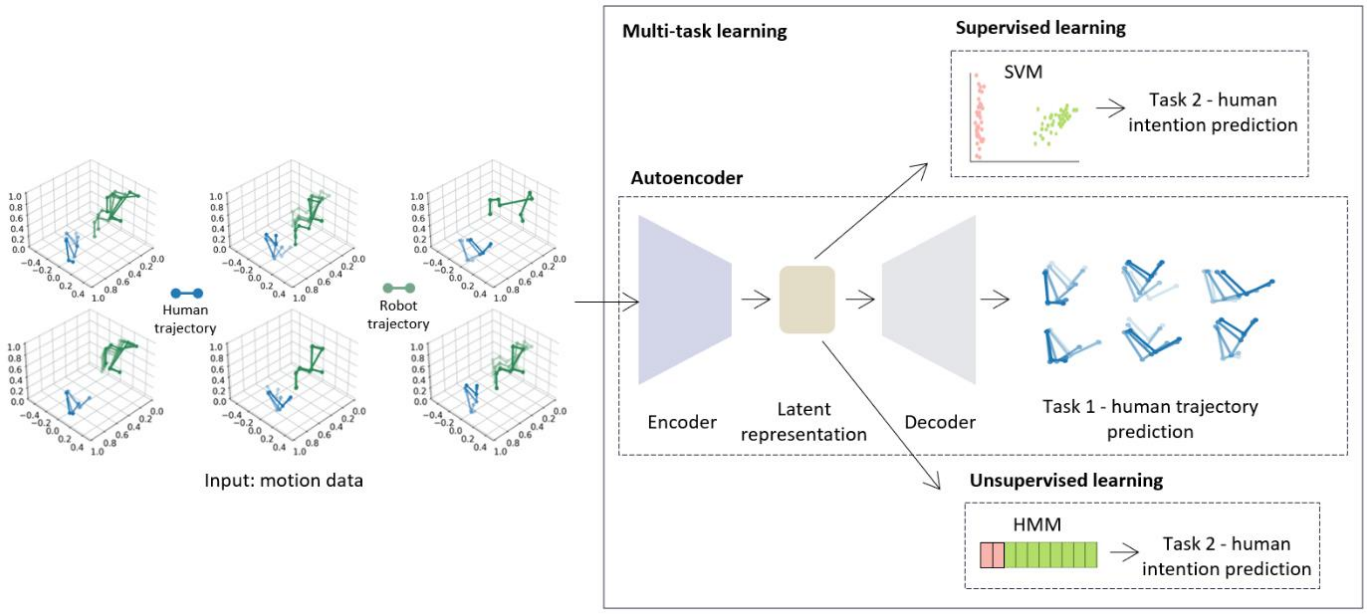

Figure 1: The proposed framework for prediction of human intent and trajectory through multi-task learning.

The objective of this paper is to develop a framework for multi-task learning and compare several encoder architectures for HRC tasks. The proposed MTL framework is illustrated in Fig. 1. This model processes motion data from both human and robot trajectories. These inputs are directed into an encoder module that analyzes the sequences to generate a latent representation of the data. From the latent space, two separate tasks are branched out. The first task employs a decoder to predict human trajectories, focusing on reconstructing the motion sequences. The second task uses supervised learning, specifically a Support Vector Machine (SVM), to predict human intention based on the latent representation. Moreover, an unsupervised learning method, Hidden Markov Model (HMM), is utilized for human intention prediction, which offers a different approach to decoding the latent features. Each of these tasks aims to decode a different aspect of the input sequences; one for the movement trajectory, and the other for the intended path.

The paper is structured as follows: Section 2 reviews relevant literature. Section 3 outlines the methodology used to create the framework. Section 4 presents a case study to evaluate the framework in human-robot collaborative disassembly tasks. Finally, section 5 concludes the paper.

2. RELATED WORK

In this section, we briefly summarize the relevant literature on HRC with respect to methods for perceiving human intentions and trajectories, the corresponding prediction methods, and the need for predicting intentions and trajectories in multi-task learning.

2.1 Human intention prediction

In human-robot teamwork, anticipating each other's actions facilitates the coordination of actions among team members. Humans possess this ability to exchange information, either directly using gestures or words, or indirectly using facial expressions or internal guesses. Equipping robots with a similar capability to collaborate with humans remains a challenge, but inferring human intentions offers a promise in addressing this challenge. Efforts have been made by researchers to develop robots capable of inferring human intentions [3]. To name several examples, Fan et al. [4] suggested identifying human intentions by preprocessing body postures and evaluating them in an HRC disassembly scenario. Margrini et al. [5] introduced the recognition of the operator's gestures to understand the human intent and to control a robot for collaborative polishing operations. In the manufacturing domain, robots should have a semantic understanding of human intent specific to the task at hand, unlike the general inference required in everyday activities.

Beyond merely inferring intentions, robots as reliable teammates should have a keen and accurate ability to predict intent [6]. This helps humans working alongside robots experience a sense of reassurance, knowing that their robotic colleagues truly understand them and there is no fear involved. Some research on predicting human intent in HRC has been published recently. The Human Digital Twin framework, which uses LSTM modules for learning spatial-temporal features, achieved an accuracy of 98.54% for human action intention recognition [4]. Moreover, employing an LSTM network allowed for inference of the operator's intention with an accuracy of 86.49% [7]. Adding a self-attention layer after LSTM layers proved to have 91% accuracy [8]. Existing studies have demonstrated that the utilization of Recurrent Neural Networks (RNNs), such as LSTM, allows for the learning of human features to predict human intent. However, these approaches often overlook the incorporation of robot features, particularly for HRC scenarios.

2.2 Human trajectory prediction

In manufacturing field environments where humans and robots work in proximity, predicting human trajectories has become a critical area of research. According to Xiao et al. [9], the analysis of disassembly trajectories helps obtain the optimal disassembly paths. This can be further extended to path planning and decision-making for robots. For example, in a shared workspace, if a robot can predict that a human worker is about to occupy a space, it can plan its path in advance to avoid that area rather than having to stop and reactively change its path. A considerable amount of research has contributed to this field [10, 11]. In addition, due to the uncertainty of product design growth and factory flexibility expectations, understanding individual human movements can help robots adjust their actions to the preferred cooperative style and work habits of individual human operators. This can increase operator satisfaction and reduce fatigue or stress associated with working near robots.

Multiple studies have demonstrated the importance of capturing human motion in existing HRC developments [12, 13]. Zhou et al. [14] conducted a study on attention mechanisms applied to human motion tracking and arm trajectory prediction. Their work was experimentally validated in an assembly task involving a collaborative robot. Incorporating the movements of the robot, Zhu et al. [15] employed a neural network approach to predict arm trajectories considering the distances between each link of the robot and each joint of the human. The integration of motion data was essential for providing safety. Hence, the use of deep learning models to predict an operator's trajectory becomes imperative. Incorporating the robot's motion into this prediction process demonstrates substantial improvements in terms of safety and predictability.

2.3 Multi-task learning

Upon reviewing the studies on human intention and trajectory prediction, it becomes evident that one specific limitation is an exclusive focus on individual aspects without simultaneous prediction. The simultaneous prediction of human intentions and trajectories improves the performance of HRC. Robots can comprehend what action a human is about to take as well as understand where human movement is likely to occur. To achieve this, MTL has emerged as an effective approach. MTL works on the principle that tasks learned simultaneously can positively influence each other and provide better generalization, which leads to improved performance compared to learning tasks alone [16]. The concept of MTL has been put into practice in HRC scenarios [17, 18]. Nonetheless, further exploration is needed to develop the prediction of human intentions and trajectories within the MTL framework, particularly in the manufacturing domain.

3. METHODOLOGY

This section introduces the general workflow for predicting human intentions and trajectories. The process begins with selecting a Bi-LSTM-based encoder-decoder architecture for handling sequential data. Four different encoder designs are evaluated for feature extraction. Also, the methods for intent classification, both supervised and unsupervised, are discussed. Finally, the objective function used for model training is explained. The following subsections provide detailed descriptions of each component.

3.1 Bi-LSTM encoder-decoder architecture

Sutskever et al. [19] first proposed the encoder-decoder architecture and applied it for neural machine translation, which was designed to convert sequences from one domain into sequences in another domain. The encoder model encodes the input sequence as a convex vector and aims to learn the information in the input. After the encoder, the decoder takes the convex vector and generates the output. The encoder or decoder is usually designed as a recurrent neural network, such as an LSTM or gated recursive unit.

The workflow of an encoder-decoder architecture consisting of Bi-LSTM networks is shown in Fig. 2. The LSTM cell comprises a current input vector x_t, the last memory cell state c_{t-1}, and the last hidden state h_{t-1}. The way each LSTM cell operates is explained mathematically as,

$$f_t = \sigma(w_{fx}x_t + w_{fh}h_{t-1} + w_{fc}h_{t-1} + b_f) \quad (1)$$
$$i_t = \sigma(w_{ix}x_t + w_{ih}h_{t-1} + w_{ic}h_{t-1} + b_i) \quad (2)$$
$$o_t = \sigma(w_{ox}x_t + w_{oh}h_{t-1} + w_{oc}h_{t-1} + b_o) \quad (3)$$
$$c_t = c_{t-1} \odot f_t + i_t \odot tanh(w_{ct}x_t + w_{ch}h_{t-1} + b_c) \quad (4)$$
$$h_t = o_t \odot tanh(c_t) \quad (5)$$

where f_t, i_t and o_t are namely the forget gate, input gate, and output gate. w and b are linear transformation matrices and biases.

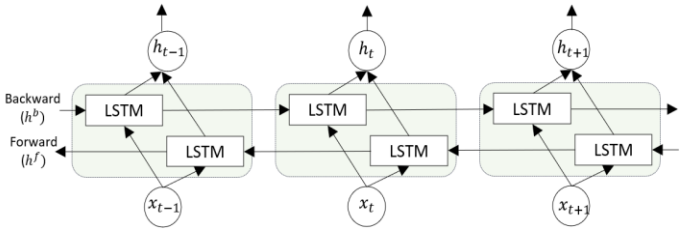

Figure 2: Architecture of the Bi-LSTM network.

Different from the standard LSTM model that processes data in a sequence focusing on the current and past information, we select a Bi-LSTM model that considers both past and future data points. This enhancement is achieved by adding an extra layer to the LSTM network. In a Bi-LSTM, there are two key layers: the forward hidden layer h_t^f and the backward hidden layer h_t^b [20]. The forward hidden layer processes the input in the natural order, that is, starting from the first element and moving forward. On the other hand, the backward hidden layer processes the input in reverse order, beginning from the last element and moving backward. The final output H_t is produced by combining the outcomes from both the forward and backward hidden layers. The implementation of the Bi-LSTM model is based on the following equations:

$$h_t^f = tanh(w_{hx}^f x_t + w_{hh}^f h_{t-1}^f + b_h^f) \quad (6)$$

$$h_t^b = tanh(w_{hx}^b x_t + w_{hh}^b h_{t-1}^b + b_h^b) \quad (7)$$

$$H_t = w_{hy}^f h_t^f + w_{hh}^b h_t^b + b_y \quad (8)$$

3.2 Different encoder designs

In the encoder-decoder architecture, the encoder converts the input sequence into a hidden representation, and then the decoder changes this representation back into an output sequence. A possible issue with this method is the risk of losing information during the process. To address this issue, we have added attention and pooling mechanisms to the encoder. Four different designs of the encoder are shown in Fig. 3.

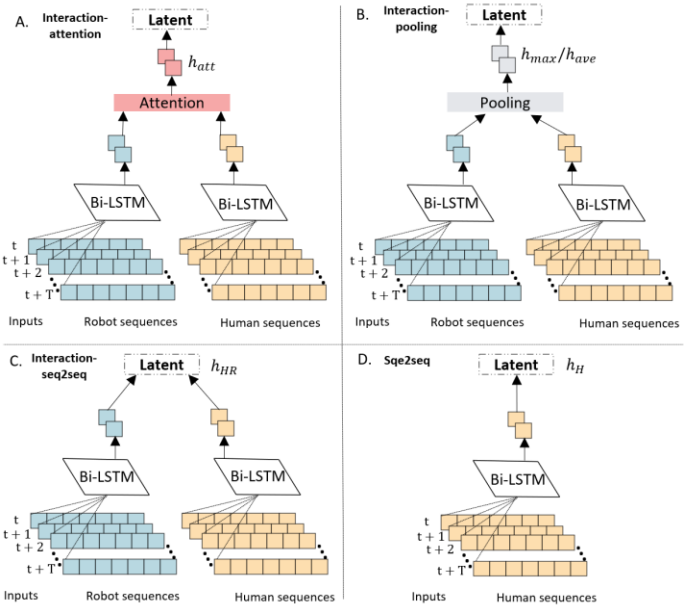

Figure 3: Different encoder architectures sharing a common decoder model.

The Interaction-attention encoder (shown in Fig. 3A) processes robot and human sequences, directing their outputs to an attention layer. An attention mechanism was introduced that allows the network to focus on the most relevant information [21]. The output sequences from the Bi-LSTM network are denoted by $H = [H_1, \ldots, H_T] \in \mathbb{R}^{NxT}$, where N represents the dimensionality of the output feature vector at each time step, and T represents the number of time steps. An attention mechanism processes these sequences to compute alignment scores, indicated by e_k^t. These scores quantify the relevance of each input element to the current output element being considered. Attention weights are then derived using a softmax function over these scores,

$$h_k^{t\prime} = \frac{exp(e_k^{t\prime})}{\sum_{t\prime=t+1}^{t+m} exp(e_k^{t\prime})} \quad (9)$$

This formulation confirms that the attention weights sum to 1, allowing the model to focus on different segments of the input sequence for each output time step. After the calculation of attention weights, the vectors are output by:

$$h_{att} = \sum_{t=1}^{T} E_t H_t \quad (10)$$

The Interaction-pooling encoder (shown in Fig. 3B) processes sequences from both robot and human and directs the outputs towards a layer specifically designed for pooling operations. To focus on the most relevant information without the need for complex attention weight calculations, the network employs both average and max pooling strategies [22].

For average pooling, the operation is defined as taking the mean within the specified dimensions of the sequence to generalize the overall trend or average effect present in the sequence. The equation for average pooling over the temporal dimension is given by:

$$h_{ave} = \frac{1}{T}\sum_{t=1}^{T} H_t \quad (11)$$

This results in a vector h_{avg} that provides a summarized representation by averaging the features in all time steps, thereby condensing the temporal information into a single vector.

In contrast, max pooling is utilized to capture the most dominant features in the sequence, by selecting the maximum value within the specified dimension for each feature. The operation is formalized through the following equation:

$$h_{max} = max(\sum_{t=1}^{T} H_t) \quad (12)$$

Here, h_{max} denotes the vector consisting of the largest feature values identified in all time steps.

The Interaction-seq2seq encoder, illustrated in Fig. 3C, processes both human and robot motion sequences. The output vectors are directly derived from the Bi-LSTM without further transformation:

$$h_{HR} = H_t \quad (13)$$

This design indicates a direct mapping from input to output, handling sequence data without additional processing layers or functions.

To establish a baseline model, the Seq2seq encoder, shown in Fig. 3D, exclusively processes human motion sequences. We aim to compare whether including robot motions can enhance the prediction of human motion, given the absence of any specialized information extraction function:

$$h_H = H_{t-human} \quad (14)$$

This formulation also represents a direct mapping of human motion from input to output.

3.3 Supervised and unsupervised intention classification

We employ SVM for supervised classification which is a discriminative classifier. SVMs are discriminative classifiers that, given labeled sequential data, identify the optimal hyperplane to separate different classes.

In addition, we select an HMM model for unsupervised recognition. An HMM consists of a set of discrete hidden states and observation sequences. The Expectation-Maximization (EM) algorithm is utilized to infer the hidden intention states and refine the model parameters based on the observed motion sequences. In the Expectation (E) step, the posterior distributions are calculated using the forward-backward algorithm. In the Maximization (M) step, parameters from the state transition matrix and the emission matrix are updated. After the convergence of the iterative process, the sequence of hidden states predicted by the HMM can be interpreted as the sequence of underlying intentions.

3.4 Objective function

In this model, Mean Squared Error (MSE) is used as the objective function to compare predicted motion positions with true labels. Cross-entropy is excluded due to the unsupervised nature of intent classification. Therefore, the model only considers trajectory prediction error to optimize the results.

4. EXPERIMENT AND RESULTS

This section presents a case study used for data collection and model evaluation. We designed a human-robot collaboration experiment focused on disassembling a desktop. In addition, we compared the performance of various encoder architectures.

4.1 Experimental design and dataset

We designed a human-robot collaborative disassembly experiment to gather data for testing the proposed framework. In this setup, the human and the robot stand face-to-face to disassemble an end-of-life desktop computer. The human operator is tasked with removing two screws located on the left and right sides of the desktop, while the robot is assigned to pick up a disassembled CD drive near the right screw, as illustrated in Fig. 4.

The experiment aimed to investigate the impact of the collaborative robot on the human worker's decision-making and corresponding movements. We tested two velocities for the robot: fast and slow. For this experiment, the robot's end-effector velocity was set to 0.5 m/s for the fast speed and 0.08 m/s for the slow speed. Each time, the robot randomly selected to move at either speed to pick up the CD drive.

When the robot moved fast, the human did not have sufficient time or space to release the right screw safely. To avoid collisions, the human operator would first remove the left screw and wait for the robot to move away before releasing the right screw. On the other hand, when the robot moved slowly, the human worker felt confident to complete the screw disassembly before the robot arrived. In the absence of safety concerns, the human would remove the right-hand screw first, following the preference of the right hand.

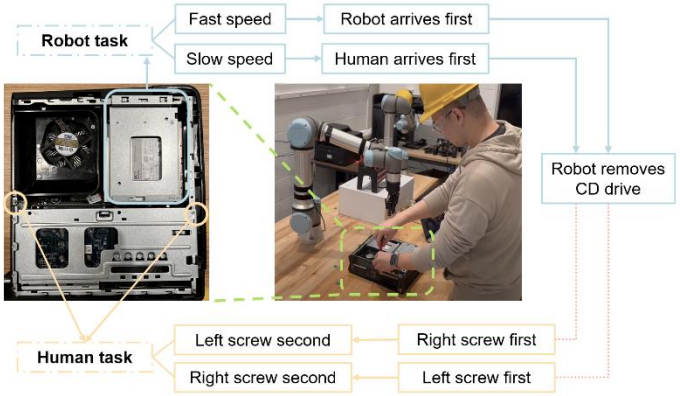

Figure 4: Experimental setup and disassembly task assignment.

We used the Vicon motion capture system to track the movement of the human operator's right arm. To determine the positions of the rotating joints, we placed two markers on each side of the shoulder, elbow, and wrist. The positions of the joints are estimated by averaging the positions of the corresponding markers. Data is recorded at a frequency of 50 Hz. We collected an equal number of trials for each robot speed setting and divided the data into training and test datasets in a 2:1 ratio.

4.2 Results on different test sets

We train the models using all collected training data under two speed modes of the manipulator. We then test the models on three different test sets: fast-speed, slow-speed, and overall. Results are introduced and compared based on these test sets. Given the limited data size, we use all motion trials from the training set to obtain pre-trained models. Evaluating these pre-trained models on diverse test sets helps us assess the performance of different feature integration methods.

We select the Mean Squared Error (MSE) and the coefficient of determination R^2 as criteria to evaluate the trajectory prediction results. For classification results, we use accuracy as the primary metric. To assess the generative performance of the models, we ran them on five predetermined random seeds. The results, presented as mean and standard deviation values, are summarized in Table 1.

First, we compare the performance on trajectory prediction. The Interaction-attention model consistently demonstrated the best performance. This is attributable to the attention mechanism's ability to capture important temporal relationships in time series data. For the classification performance, different models exhibit varying levels of prediction accuracy. This variability is reasonable, as the classification task was not incorporated into the loss function. Further, we observe that the pooling function outperformed the attention mechanism. Although the attention mechanism is robust for sequence

prediction, its advantage may be reduced in sequence classification tasks due to the limited data available.

Second, we compare the performance between encoder models. With the introduction of multiple feature integration encoders, we conduct detailed model comparisons. Integrating robot motions significantly improved the prediction of human motions when comparing Seq2seq and Interaction-Seq2seq models. This improvement is particularly valuable in human-robot collaboration settings and shows the importance of considering robot motion, even when the robot follows a predefined path. Evaluating various pooling functions reveals that max pooling is better suited to our scenario. Furthermore, when comparing the max pooling model with the attention model, we observe that the attention mechanism demonstrated superior performance in trajectory prediction.

Table 1: Prediction results on distinct test sets.

Fast-speed set	Classification		Trajectory	
	SVM	HMM	MSE	R^2
Seq2seq	0.79	0.64	0.10	0.87
Interaction-seq2seq	0.81	0.70	0.07	0.92
Interaction-pooling (Ave)	**0.83**	0.65	0.11	0.88
Interaction-pooling (Max)	0.82	0.64	0.08	0.91
Interaction-attention	0.81	**0.74**	**0.06**	**0.93**
Slow-speed set	**Classification**		**Trajectory**	
	SVM	HMM	MSE	R^2
Seq2seq	0.85	**0.74**	0.09	0.88
Interaction-seq2seq	0.85	0.65	0.09	0.88
Interaction-pooling (Ave)	**0.86**	0.64	0.10	0.86
Interaction-pooling (Max)	0.85	0.63	0.09	0.88
Interaction-attention	0.85	0.66	**0.07**	**0.91**
Overall set	**Classification**		**Trajectory**	
	SVM	HMM	MSE	R^2
Seq2seq	0.77	0.51	0.09	0.88
Interaction-seq2seq	0.84	0.53	0.08	0.90
Interaction-pooling (Ave)	**0.85**	0.52	0.10	0.87
Interaction-pooling (Max)	0.84	**0.54**	0.08	0.90
Interaction-attention	0.82	0.51	**0.06**	**0.92**

4.3 Results of trajectory and classification plots

To visualize the trajectory results, we display the predicted trajectories from the Interaction-attention model, which has the best prediction performance. This model is trained on the past 50 time steps and predicts the future 50 time steps. Human trajectories are projected onto 3D coordinates in meters, as illustrated in Fig. 5.

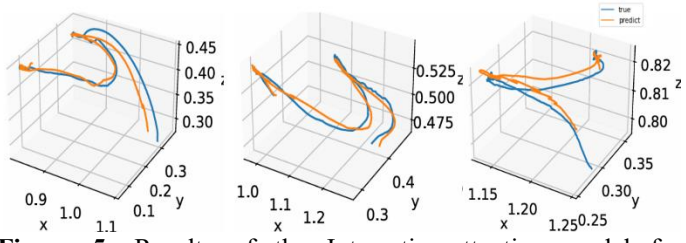

Figure 5: Results of the Interaction-attention model for visualizing human trajectories.

Fig. 6 illustrates the classification heatmaps of the Interaction-Attention model, where the classification accuracy for each intent classis listed.

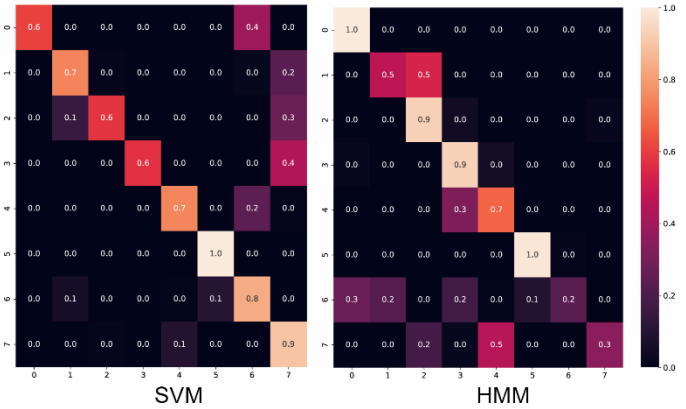

Figure 6: Results of the Interaction-attention model for human intents classification heat-maps.

4.4 Results of latent representations

Latent representations are important for evaluating whether a model can capture essential features through its learning process. By visualizing these latent spaces, we better observe the model's ability to differentiate and understand the underlying structure of the data. This is particularly important in HRC tasks where precise understanding of human intentions and trajectories is needed for safe and effective interaction. To display the latent representations, we utilize Principal Component Analysis (PCA) to reduce the original 64 latent dimensions to 3 dimensions. This dimensionality reduction technique helps project the high-dimensional data into a more interpretable form while preserving as much variance as possible.

The attention-based latent representation shows well-distributed classes and indicates that the model has successfully captured the unique features of different classes, shown in Fig. 7. Moreover, the latent spaces for both training and testing data display similar patterns, which suggests that the attention mechanism generalizes well for different data subsets. This consistency between train and test distributions validates the robustness of the attention-based approach in feature extraction.

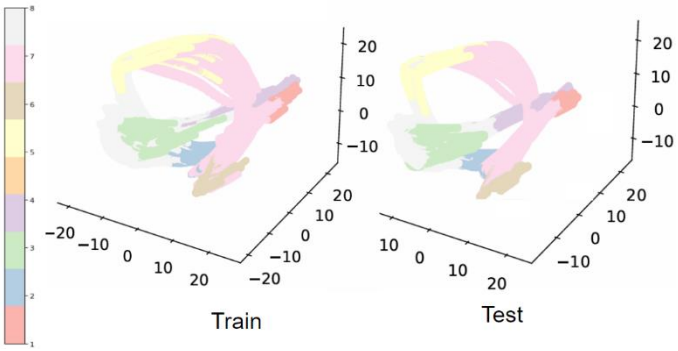

Figure 7: Performance of attention mechanism in latent space.

On the other hand, the latent representations in Fig. 8 obtained through max pooling demonstrate differences between the training and testing data. This difference indicates that max pooling may not capture the essential features as effectively as the attention mechanism. The observed variations highlight the superior capability of the attention mechanism in maintaining the integrity of feature representation in different data sets.

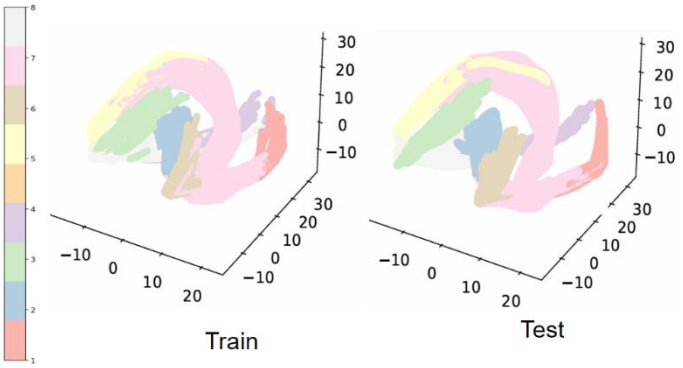

Figure 8: Performance of max polling in latent space.

4.5 Results of human joint positions

Different from trajectory plots on Fig. 5, we illustrate plots that specifically display the positions of a human's joints. This detailed visualization shows the exact movements and angles of various joints, which is essential for understanding and predicting human motion in collaborative tasks. The trajectory plots, Fig. 9 and Fig. 10, showcase both fast-speed and slow-speed test sets on the Interaction-Action model. We can evaluate the model's robustness and accuracy in predicting human joint positions under different conditions by comparing these different speed settings.

The predicted trajectories (red lines) closely follow the true trajectories (blue lines) across all joint positions. The x-axis in the plots represents time steps, and we have concatenated 8 test trials together to provide a comprehensive view of the model's performance over extended sequences. The predicted joint positions closely aligning with the true positions indicates that the Interaction-Action model effectively captures the dynamics of human movement, regardless of the speed of the action.

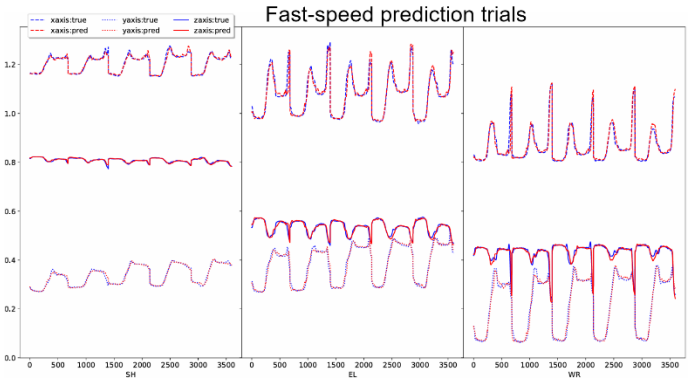

Figure 9: Trajectories of joint positions on fast-speed test set.

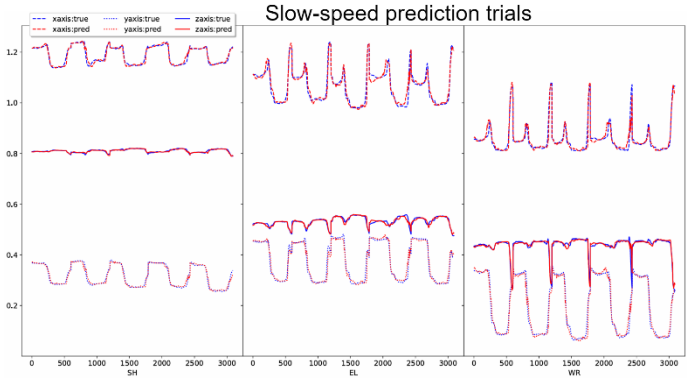

Figure 10: Trajectories of joint positions on slow-speed test set.

4.6 Results of model inference performance

This section analyzes model inference performance which assists in understanding the trade-offs between model complexity and inference speed, and helps users select the most appropriate model.

The number of parameters in each encoder module reflects the model's size and complexity. A higher number of parameters generally indicates a more complex model that may capture more complex patterns in the data. For example, the Interaction-seq2seq model in Table 2 has the highest parameter count at 651.858k.

Table 2: Prediction results on distinct test sets.

Encoder modules	Parameters (k)	Inference time (ms/batch)
Seq2seq	384.02	0.52
Interaction-seq2seq	651.86	0.83
Interaction-pooling (Ave)	417.43	0.85
Interaction-pooling (Max)	417.43	0.85
Interaction-attention	532.44	0.92

The inference time, measured in milliseconds per batch, indicates how quickly each model can predict outcomes based on new data. Lower inference times are needed for real-time applications, where speed is essential. The Seq2Seq model in Table 2 demonstrates the fastest inference time with a mean of 0.52 ms/batch and a standard deviation of 0.007 ms. On the other

hand, the Interaction-attention model, while having a considerable number of parameters (532.443k), shows a slower inference time with a mean of 0.92 ms/batch. This is because attention calculation does not require additional parameters but includes computationally intensive operations to calculate attention weights.

5. CONCLUSIONS

In this study, we aim to accurately predict both human intentions and movement paths in HRC settings. This becomes important for safety and productivity when humans and robots share workspaces. We developed a multi-task learning framework that can simultaneously predict a person's intentions and their movement trajectory. Four different encoder architectures are tested within this framework, and we explore both supervised and unsupervised methods for analyzing movement data, with a special focus on capturing the timing of these movements.

We conducted experiments where humans and robots collaboratively disassemble components to collect data and evaluate the performance of the proposed framework. The results demonstrate that the system performs well in predicting both intentions and movements. The latent representations to evaluate how well the models capture important features have been shown and detailed plots of specific human joint positions under different speed settings have been provided. In addition, we compare the inference times of different encoder designs to assess their efficiency.

The scope of the work can be extended. Specifically, the proposed framework can be applied to multi-robot collaboration scenarios, where accurate predictive trajectories among robots are necessary. Also, the proposed framework could be extended for different applications and more complex human-robot collaborative settings. Further, instead of using a Bi-LSTM based auto-encoder architecture, the evaluation can include applying different sequential methods in the encoder and decoder.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation–USA under grants # 2026276, and 2026533. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Jahanmahin, R., Masoud, S., Rickli, J., and Djuric, A., 2022, "Human-robot interactions in manufacturing: A survey of human behavior modeling," Robotics and Computer-Integrated Manufacturing, 78, p. 102404.

[2] Long, Y., jiang Du, Z., dong Wang, W., and Dong, W., 2018, "Human motion intent learning based motion assistance control for a wearable exoskeleton," Robotics and Computer-Integrated Manufacturing, 49, pp. 317–327.

[3] Bi, Z., Luo, C., Miao, Z., Zhang, B., Zhang, W., and Wang, L., 2021, "Safety assurance mechanisms of collaborative robotic systems in manufacturing," Robotics and Computer-Integrated Manufacturing, 67, p. 102022.

[4] Fan, J., Zheng, P., and Lee, C. K., 2023, "A Vision-based Human Digital Twin Modelling Approach for Adaptive Human-Robot Collaboration," Journal of Manufacturing Science and Engineering, pp. 1–11.

[5] Magrini, E., Ferraguti, F., Ronga, A. J., Pini, F., De Luca, A., and Leali, F., 2020, "Human-robot coexistence and interaction in open industrial cells," Robotics and Computer-Integrated Manufacturing, 61, p. 101846.

[6] Nahavandi, S., 2019, "Industry 5.0—A Human-Centric Solution," Sustainability, 11(16).

[7] Cacace, J., Caccavale, R., Finzi, A., and Grieco, R., 2022, "Combining human guidance and structured task execution during physical human–robot collaboration," Journal of Intelligent Manufacturing, pp. 1–15.

[8] Zhang, R., Lv, J., Li, J., Bao, J., Zheng, P., and Peng, T., 2022, "A graph-based reinforcement learning-enabled approach for adaptive human-robot collaborative assembly operations," Journal of Manufacturing Systems, 63, pp. 491–503.

[9] Xiao, J., Gao, J., Anwer, N., and Eynard, B., 2023, "Multi-agent Reinforcement Learning Method for Disassembly Sequential Task Optimization Based on Human-Robot Collaborative Disassembly in Electric Vehicle Battery Recycling," Journal of Manufacturing Science and Engineering, pp. 1–36.

[10] Nicora, M. L., Ambrosetti, R., Wiens, G. J., and Fassi, I., 2020, "Human–Robot Collaboration in Smart Manufacturing: Robot Reactive Behavior Intelligence," Journal of Manufacturing Science and Engineering, 143(3), 031009.

[11] Zhang, J., Liu, H., Chang, Q., Wang, L., and Gao, R. X., 2020, "Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly," CIRP Annals, 69(1), pp. 9–12.

[12] Li, S., Wang, R., Zheng, P., and Wang, L., 2021, "Towards proactive human–robot collaboration: A foreseeable cognitive manufacturing paradigm," Journal of Manufacturing Systems, 60, pp. 547–552.

[13] Simões, A. C., Pinto, A., Santos, J., Pinheiro, S., and Romero, D., 2022, "Designing human-robot collaboration (HRC) workspaces in industrial settings: A systematic literature review," Journal of Manufacturing Systems, 62, pp. 28–43.

[14] Zhou, H., Yang, G., Wang, B., Li, X., Wang, R., Huang, X., Wu, H., and Wang, X. V., 2023, "An attention-based

deep learning approach for inertial motion recognition and estimation in human-robot collaboration," Journal of Manufacturing Systems, 67, pp. 97–110.

[15] Zhu, Y., Chen, S., Zhang, C., Piao, Z., and Yang, G., 2023, "Development of adaptive safety constraint by predicting trajectories of closest points between human and co-robot," Journal of Intelligent Manufacturing, pp. 1–10.

[16] Caruana, R., 1997, "Multitask Learning," Machine Learning, pp. 41–75.

[17] Fan, J., Zheng, P., and Lee, C. K., 2022, "A Multi-Granularity Scene Segmentation Network for Human-Robot Collaboration Environment Perception," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2105–2110, doi: 10.1109/IROS47612.2022.9981684.

[18] Abuduweili, A., Li, S., and Liu, C., 2019, "Adaptable Human Intention and Trajectory Prediction for Human-Robot Collaboration," CoRR, abs/1909.05089.

[19] Sutskever, I., Vinyals, O., & Le, Q. V., 2014, "Sequence to sequence learning with neural networks," Advances in neural information processing systems, 27.

[20] Yousaf, K., & Nawaz, T., 2022, "A deep learning-based approach for inappropriate content detection and classification of youtube videos," IEEE Access, 10, pp. 16283-16298.

[21] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I., 2017, "Attention is all you need," Advances in neural information processing systems, 30.

[22] Kao, C. C., Sun, M., Wang, W., & Wang, C., 2020, "A comparison of pooling methods on LSTM models for rare acoustic event classification," In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 316-320.