2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Hierarchical Deep Learning for Intention Estimation of Teleoperation Manipulation in Assembly Tasks

*Mingyu Cai, *Karankumar Patel, Soshi Iba, Songpo Li

***Abstract*— In human-robot collaboration, shared control presents an opportunity to teleoperate robotic manipulation to improve the efficiency of manufacturing and assembly processes. Robots are expected to assist in executing the user's intentions. To this end, robust and prompt intention estimation is needed, relying on behavioral observations. The framework presents an intention estimation technique at hierarchical levels i.e., low-level actions and high-level tasks, by incorporating multi-scale hierarchical information in neural networks. Technically, we employ hierarchical dependency loss to boost overall accuracy. Furthermore, we propose a multi-window method that assigns proper hierarchical prediction windows of input data. An analysis of the predictive power with various inputs demonstrates the predominance of the deep hierarchical model in the sense of prediction accuracy and early intention identification. We implement the algorithm on a virtual reality (VR) setup to teleoperate robotic hands in a simulation with various assembly tasks to show the effectiveness of online estimation. Video demonstration is available at: `https://youtu.be/CMYDgcI4j1g`.**

I. INTRODUCTION

Shared autonomy to enable close human-robot collaboration is being actively investigated in industrial applications and surgical tasks [1]–[3]. Teaming up humans' dexterity and mechanic capability of robots boosts production efficiency, raising the need for robotic teleoperation. Whenever a flexible and skilled manual action is required without access to human's physical presence, teleoperation could provide a means to remedy the situation [4]. It involves a wide range of applications e.g., healthcare to safely provide medical assistance to contagious patients, industrial productions requiring sterile environments, and assistive applications restoring arm mobility to impaired users.

However, it's still challenging to operate a robot for non-experts since perception and action are in this case both mediated by technical systems, they are also possibly affected by delays. For seamless physical human-robot collaboration, the robot has to understand human performance and intentions to be able to provide effective and transparent assistance [5]–[8]. This work focuses on reliable human intention estimation for assistive motion control which is a critical component of safe and seamless robot teleoperation.

Existing works of human intention estimation investigate either grasping goals [6], [7], [9] or analyzing single short-horizon actions [5]. However, they failed to fully reason about the contextual relations between adjacent actions under an umbrella of one certain structured task, which provide potential temporal logic for understanding long-term intention and prediction. For instance, when a human-robot team is placing a screw on a wheel, there is a big probability of taking a screwdriver as the next action and the assembling target is likely to be an auto toy. Moreover, it is also necessary to know the task to generate the proper assistance to the current action. For example, the grasping constraints of the same object would be different in an usage task and for a relocation task.

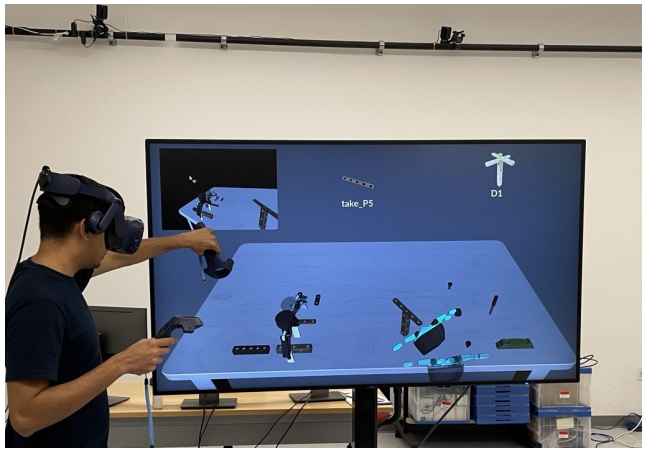


Fig. 1: Experimental setup for the data collection and model testing. The movements of human operator's head, hands, and eye gaze are tracked via HTC Vive virtual reality system. The top-left corner of screen visualizes the scene as perceived by the operator's point of view, and the background scene shows the global view of a teleoperation process. Action and task estimation results are shown in the middle and top right screen respectively.

Contribution In this work, we formulate intention estimation at hierarchical levels. In particular, the low-level intention estimation tracks fine actions for control assistance. The high-level mechanism is to predict human's long-horizon coarse tasks, which provides useful instructions of action sequences. Instead of developing separate models for each level that may cause hierarchical inconsistency, we are inspired by the hierarchical classification strategy [10] and extend it to the sequential neural network models. The novelty is to incorporate dependency information of hierarchical layers in a top-down manner, where the output of the lower level is conditioned by its upper level. We present three main contributions:

[1]Mingyu Cai is with the Department of Mechanical Engineering, University of California, Riverside, CA, USA, 92521. This work is done when he was working with Honda Research Institute. Karankumar Patel, Soshi Iba, Songpo Li are with Honda Research Institute, San Jose, CA, 95134 USA. * Both authors contributed equally to this research.

- Different from previous method [10], our hierarchical levels require different sequential lengths of data, resulting inconsistent multi-input horizons. We address this issue by proposing a multi-window strategy that forwards the input data with a different range of masks to achieve flexible hierarchical data inputs.
- Compared with the standard method, we show that the layer-dependent deep hierarchical model is capable of improving the estimation performance using inputs from either motion data or visual egocentric view data.
- A new assembly dataset was collected in a virtual reality setup with two robot hands in a simulation to manipulate objects in teleoperation. The online performance is demonstrated through 6 assembly tasks with 21 actions in total.

It's also worth pointing out that our architecture can be easily extended with the state-of-the-art estimation models for more sophisticated intentions and performance improvement.

Related Works In the context of teleoperation, advancing autonomy mainly addresses two challenges: predicting the operator's intent in performing a task and deciding how to assist the teleoperator [11], [12]. Existing literature in general describes the what-to-predict and how-to-assist problems . After inferring the operator's intentions, many works [13]–[15] integrated cooperative motion planners and learning-based policies from demonstrations. Within the concept of human intention estimation, early approaches and several recent ones formulated the user control input in driving the robotic movement as behavioral cue for inference and prediction [16], [17]. To predict a distribution over the different action targets, most of these works fused robot motion features such as end effector pose, velocity, arm joints, or whole gestures, and various types of observation on human behavior including human trajectories, gesture, gaze information giving Area-of-Interest of teleoperators, speech, facial expressions, and force-torque measurements. In this paper, we focus on intetions recognitions for high-level actions and takks.

Along the line of intention estimation, Hidden Markov Models (HMMs) have been widely used to analyze a discrete set of tasks/subtasks [17]–[19]. These works are studied on a single-layer, whereas human intention is often composed of a multilayer hierarchy. The use of hierarchical HMM representations has been investigated for multi-layer classifications [8], [16], [20]. The aforementioned works generally infer the probability distribution over intentions by dynamic programming, which may be computational expensive for online performance with rich and long sequential observations, and also increases the complexity of modeling. Neural networks (NNs) have seen increasing popularity in robotics, and sequential NN models e.g., RNN and transformer, are becoming powerful tools for human-robot situation understanding [6], [21]–[24]. These works detect user motion intent from limb dynamics and from various sensors to enforce collaboration tasks. Direct feed-forward after training makes NNs efficient in practice. The hierarchical structure of intentions in human-robot interaction has not been thoroughly explored in neural networks literature, and this study compares its accuracy within this context. A command can be interpreted as a pyramid of a goal, sub-goals, and primitives. Only a few existing works [25]–[28] designed a hierarchical network based on topological properties of graphical task representations. However, their models didn't include the top-down relation during training and required experts to pre-construct the graph structure.

II. Problem Formulation

A collaboration team of human teleoperating robots is assigned a set of m toy assembly tasks denoted as T, which aim to build desired targets e.g., airplanes, vehicles, and block buildings. The human teleoperator attempts to take a set of n actions in total denoted as A e.g., pick up a screwdriver, screw track with the left hand, pick up a toy block, etc, and actively leads the team to complete all tasks by performing actions sequences that are unknown to the robot. We define the human intention at time-step t as $H_t = (T_t, A_t)$, where $T_t \in T$ and $A_t \in A$ represent task and action attempted to perform at time t. With modern sensor equipment, the online observations history $X_{1:t} \in \mathbb{R}^{t \times F}$ is available that includes information on intention estimation e.g., human-robot motion features, videos of surrounding cameras, egocentric views, gaze, etc, where F denotes the number of input features. To achieve seamless teleoperation, it's expected to online capture the intention of the teleoperator and subsequently provide autonomous shared control as assistance.

Different from existing works, this work considers the hierarchical intention relations shown in Fig. 2. In practice, each task T_t does not include all action categories. For instance, the block-building task never involves the actions related to screws. We denote A_t^T as the set of actions that the task T_t only takes from. The problem can be formulated as: at every time-step t with the observation history $X_{1:t}$, the objective is to efficiently predict the teleoperator's intention $H_t = (T_t, A_t)$ with hierarchical relations online s.t., $A_t \in A_t^T$.

III. Method

A. Deep Hierarchical Model

Always taking $X_{1:t}$ as the input results in issues of dynamic input and numerous lengths. Assigning the proper window size for the sequential data is a common modeling technique that is applied to process datasets. We denote L as the selected window size, and our model only considers the most recent L time-steps. Consequently, the dataloader generates $X \in \mathbb{R}^{L \times F} = X_{t-L:t}$ as the input. And the ground truth intention $H_t = (T_t, A_t)$ only depends on the attempted behavior at the current time step t, which can be generated through standard annotation process through the manual segmentation and labeling of the collected dataset of actions and tasks. [29]. Annotation efficiency could also be achieved by employing a hybrid approach that combines human labeling with state-of-the-art segmentation models.

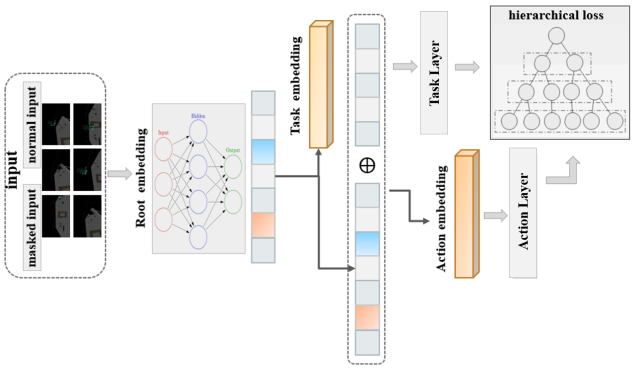


Fig. 2: The task-action hierarchical deep learning model including dependent loss functions and leaf layers conditional by the embeddings from its root layer.

The categories are organized by a tree with two hierarchical layers, where task prediction Y_T is the root layer of action inference Y_A. Let $\tilde{T}$ and $\tilde{A}$ denote the outputs of layers Y_T and Y_A at time-step t. Since the predictions rely on contextual relations of observation history, this framework applies the sequential neural network models as the backbone e.g., RNN [30], LSTM, [31] transformer [32], Slow-Fast [29] , etc. We define the applied backbone (root) neural network as $\mathcal{N}_r(X, \theta_r)$, where θ_r are the parameters to be trained. Its output can be regarded as the root latent space: $X_r = \mathcal{N}_r(X, \theta_r)$.

Given the root representation, the objective is to generate hierarchical representations for task and action layers. Since the action layer is the leaf node of the task layer, we design the neural network structure such that the action prediction is conditioned on the task inference i.e., $P(\tilde{A}_t|\tilde{T}_t))$, where $P(\cdot|\cdot)$ represents the conditioned probability. To do so, first, we construct the task and action encoders, respectively, i.e., $X_T = \mathcal{N}_T(X_r, \theta_T)$ and $X_A = \mathcal{N}_A(X_r, \theta_A)$.

Then, the task classification layer can be designed using softmax regression as

$$\tilde{y}_{Ti} = \frac{exp(W_{Ti} * X_T)}{\sum_{k=1}^{m} exp(W_{Tk} * X_T)}$$

where W_{Ti} are the parameters (weights) of ith task category. To condition the prediction of action, we first concatenate the action and task embeddings i.e., $X_{A|T} = X_A \oplus X_T$. Similarly, the action classification layer can be constructed as

$$\tilde{y}_{Ai} = \frac{exp(W_{Ai} * X_{A|T})}{\sum_{k=1}^{n} exp(W_{Ak} * X_{A|T})}$$

where W_{Ai} are the parameters (weights) of ith action category. Finally, the inference results $\tilde{T}$ and $\tilde{A}$ can be obtained by taking the $\arg\max$ of $\tilde{y}_T$ and $\tilde{y}_A$.

The classification loss function of action and task is designed through standard classification entropy loss as:

$$ELoss = -T_t \cdot \log(\tilde{T}_t) - A_t \cdot \log(\tilde{A}_t)$$

To enhance the hierarchy relations, we introduce $\mathbb{D}$, $\mathbb{I}_A$, and $\mathbb{I}_T$ to indicate whether the intention predictions of neural network model have conflict hierarchical category structure i.e., $A_t \notin A_t^T$, especially

$$\mathbb{D} = \begin{cases} 1 & \text{if } \tilde{A}_t \in A_t^T \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbb{I}_T = \begin{cases} 1 & \text{if } \tilde{T}_t = T_t \\ 0 & \text{otherwise} \end{cases}, \mathbb{I}_A = \begin{cases} 1 & \text{if } \tilde{A}_t = A_t \\ 0 & \text{otherwise} \end{cases}$$

Based on that, the hierarchical dependence loss is formulated as:

$$DLoss = -(ploss)^{\mathbb{D}\cdot\mathbb{I}_A} * (ploss)^{\mathbb{D}\cdot\mathbb{I}_T}.$$

where $ploss$ serves as a penalty that enforces the neural network to acquire structural information from the category arrangement. The value of $ploss$ can either be fixed as a constant or be linked to the prediction error. The total loss of the model is defined as the weighted summation of the classification entropy loss $Eloss$ and hierarchical dependence loss $Dloss$ i.e.,

$$Loss(\theta) = \alpha \cdot ELoss + \beta \cdot DLoss,$$

where $\alpha \in (0, 1)$, $\beta \in (0, 1)$ are tuning parameters to bias the weights of different loss functions.

B. Multi-window Strategy

In the practice of sequential models, the length of input data is crucial for classification accuracies. In our case, the action inference needs a shorter length of input sequential data compared with task prediction, whereas many deep learning models require a fixed length of input sequential data. Directly sharing the same input with the longest length of data for both task and action recognition is not ideal, since additional unnecessary information may confuse the action inference model and downgrade its performance.

To address the issue and achieve more informative inputs, we developed the multi-window method using the mask technique. In particular, we create the latent embedding space for task and action, respectively, in the model. Each window takes different inputs such that we make the unnecessary horizons of input data for action embeddings. This allows the model to discard masked information and operate only on useful data horizons at hierarchical levels.

In particular, we denote $M \in \{0, 1\}^{(L_0+L_1)}$ as the sequential mask vector generated by users, where $L_0 + L_1 = L$ and 0 indicates a time-step (index) is invisible for the model, and vice versa. Here L_0 represents the prefix data that should be masked, and L_1 represents the suffix data that should be kept the same. Let $M[i]$ denote the ith element of the mask vector. Thus, we have $M[i] = 0, \forall L_0 \geq i > 0$, and $M[j] = 1, \forall L_1 \geq j > L_0$. Given the current input $X \in \mathbb{R}^{L\times F}$, the mask process generates the representation of valid inputs $\hat{X} \in \mathbb{R}^{L\times F}$ as

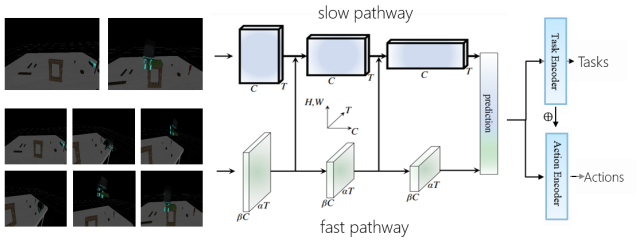


Fig. 3: Hierarchical Slow-Fast model accepts only visual inputs without feature extractions.

$$\hat{X} = Mask(X, M) = \left\{ X[i] : \begin{cases} X[i] & \text{if } M[i] = 1 \\ 0 & \text{otherwise} \end{cases} \right\} \quad (1)$$

In this framework, we choose the L of input data X as the sequential length for parent (task) layer prediction, since the one task has longer horizons including a sequence of actions. Then, L_1 is selected according to the longest duration of all actions. Finally, we forward X and $\hat{X} = Mask(X, M)$ as the input of task and action embedding models, respectively. As a result, the task and action embedding can be produced in a heterogeneous way i.e.,

$$\begin{cases} X_T = \mathcal{N}_T(\mathcal{N}_r(X, \theta_r), \theta_T) \\ X_A = \mathcal{N}_A(\mathcal{N}_r(Mask(X, M), \theta_r), \theta_A). \end{cases} \quad (2)$$

C. Vision-based Deep Hierarchical Model

With only visual input, we apply Slow-Fast model [29] that can learn useful temporal information for video recognition, as the backbone to further test the performance of the deep hierarchical model. It includes two pathways i.e., a Slow pathway, operating at a low frame rate to capture spatial semantics, and a Fast pathway operating at a high frame rate, to capture motion at fine temporal resolution. It has shown SOTA accuracy on popular benchmarks, Kinetics, Charades, and AVA. The motivation for applying such types of neural network models is that data on motion features may not always be available. In practice, it's desired to predict intentions only using perception information.

In this work, we extend the standard Slow-Fast model by integrating the developed hierarchical structure as shown in Fig. 3, where the primary model extracts temporal and spatial information as the inputs of task and action embeddings. The mask mechanism in section III-B is modified in a way that $X[i] \in \mathbb{R}^{H \times W \times C}$ of egocentric history $X \in \mathbb{R}^{L \times H \times W \times C}$ is the frame images, where H, W, C represent with, height, and number of channels. The mask process is still the same as (1) by setting the elements of the $3D$ matrix $X[i]$ as zero. Finally, we can add a multilayer perception at the end to produce task and action embeddings as (2).

D. Manipulation Assistive Control

Once we obtain the results of intention estimation, we can subsequently pass it into developed assistive control modules to provide autonomous AI support and mitigate the operational workload. There are two common existing formulations to assist the teleoperator. First, shared control is popular to correct the movement of the robot arm according to intention predictions [17], [33].

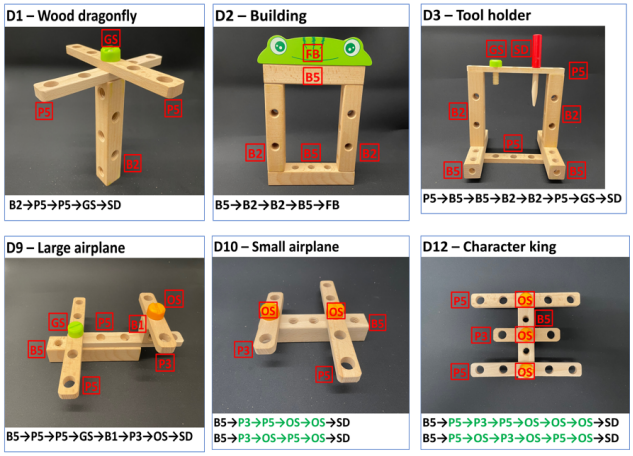


Fig. 4: Toy assembly tasks with one of instructions.

On the other hand, continuously operating the remote arm for routine tasks can be cumbersome for the teleoperator, especially in the presence of communication latency. In such a scenario, the teleoperator may relax and the system will automatically switch to autonomous control mode which the robot arm takes the intention estimation as inputs and recursively re-plans through trajectory generation [34], [35] or imitation learning policies [12], [36], to executes the task and corresponding action sequence for the next steps. The operator can take over the control back at any time and customize their desired behaviors.

When conducting remote teleoperations, the teleoperator only receives perception information from the virtual fixtures of the robots, resulting in the reality gap between the operator's simulated scenarios and the robots' actual workspace. In particular, it's challenging for humans to establish real contacts remotely. Our framework includes an AI support module that leverages the manifold information of objects to align the motion of robot end-effector with the desired contact path.

IV. Experimental Results

Experimental setup: We collected teleoperation sequences of human users performing assembly tasks by operating two robotic hands on a virtual reality setup in a simulation developed in [7], [9]. The users performed 6 assembly tasks in a virtual scene rendered via Rviz. This was displayed in the HTC Vive Pro Eye headset, featuring a 1440×1600 pixels screen per eye (2880×1600 pixels combined, 110 degrees of Field-Of-View), and a binocular Tobii eyetracker working at 120 Hz. The virtual scene consisted of a table with 10 types of toy assembly pieces.

Dataset collection: We collected data on 13 participants performing 6 assembly tasks (toys) that are shown in Fig. 4. In the figure, one of the instructions as a sequence of assembling pieces is displayed below the image. Except for inferring the assembly toys of the teleoperators, we are interested in capturing the natural order of actions in which the participants assemble the targeted toy. We label actions based on their start

and end times. There are 21 actions in total that are designed based on the movements of end-effectors i.e., picking, placing, fastening, withdrawing, and associated objects. The actions span two or three stages i.e., pre-contact when the hand (and tool) starts approaching the object, the interaction, and post-contact when the object is released. This information from the dataset is crucial for intention estimation. Our dataset includes total of 202 demonstrations of teleoperating 6 tasks. During each process, we record 6D pose of objects in workspace and two-arm end-effectors, gaze direction, and video frames of egocentric views of teleoperators as shown in Fig. 1 at 10Hz. The average demonstration duration is 1.5 minutes across the 6 tasks. To test the improvement of hierarchical designs in diverse neural network structures, we split data into two types i.e., egocentric views and the rest features as the motion features. This helps to demonstrate the predictive capability of different feature combinations.

Baselines: We integrate our model with Graph Convolutional Network (GCN), LSTM, and SLOW-FAST neural networks that are mainly compared with two baselines: (1) Independent NN: applying these neural networks for task and action predictions independently, (2). NN-HMM: combining neural network models and Hidden Markov Model [37] reasoning task and action in two stages of bottom-up manner. Note that HMM itself is not able to take visual inputs as the SLOW-FAST model. We abbreviate Hierarchical as Hie in Table I, II, and III.

Training: For each neural network model, we choose a data length of 35 frames for task reasoning and 10 frames for action prediction, corresponding to durations of 3.5 seconds and 1 second, respectively. The selection of these lengths aims to encompass sufficient information for both task and action. The number of actions and tasks is not balanced. The model is always expected to perform well in the minority class as well as the majority class for multi-label classification. Before training, we calculate the category weights of actions based on a balancing method, which adjusts weights inversely proportional to class frequencies, and then pass these weights into the optimizers at hierarchical levels to fit the model. Furthermore, we normalize the motion features to ensure they are treated equally by neural networks.

Evaluation: Accuracy is computed on a per-frame basis. The testing evaluation is real-time implementation. The Slow-Fast model's network weights are initialized from the Kinetics-400 classification models. For a more in-depth understanding of the implementation, refer to [29]. In the case of NN-HMM, the model undergoes a two-stage bottom-up training process. Initially, LSTM and Slow-Fast serve as neural network models with diverse input types for training on the action estimation layer. Subsequently, the predicted actions are utilized as input data for HMM, where the Viterbi algorithm generates tasks.

Results: Initially, GCN and LSTM serve as fundamental models, handling motion features. The baseline (1) employs them separately for action and task intention estimation, overlooking hierarchy relations. The hierarchical HMM model also utilizes motion data as input, with improved accuracies demonstrated in Table I. Hierarchical structures outperform alternative approaches. In a second step, we integrate the hierarchical structure into Slow-Fast neural networks to showcase its generalization across diverse inputs. The resulting accuracy comparison is presented in Table II, indicating enhanced performance of video models with the hierarchical structure.

TABLE I: The accuracy comparisons with data type of motion features, where Hierarchical is abbreviated as Hie.

Method | Accuracy	Data Type	Action	Task
NN-HMM	motion	92.27%	94.91%
LSTM	motion	92.27%	96.79%
Hie-LSTM	motion	**95.41%**	**98.25%**
GCN	motion	89.98%	96.15%
Hie-GCN	motion	**94.73%**	**97.10%**

TABLE II: The accuracy comparisons with egocentric data, where Hierarchical is abbreviated as Hie.

Method | Accuracy	Data Type	Action	Task
Slow-Fast	egocentric	82.81%	84.57%
Slow-Fast-HMM	egocentric	82.81%	85.94%
Hie-Slow-Fast	egocentric	**86.18%**	**87.33%**

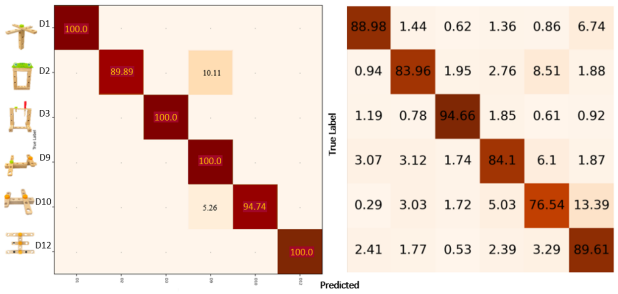
The confusion matrix in Fig.5 depicts the performance of using LSTM and Slow-Fast as backbones for the hierarchical model. Examining Fig.5a, mispredictions between tasks D9 and D10 are noticeable due to shared configurations at the beginning. Despite this, at least 50% accuracy in recognizing intentions for each action is achieved on average, dependent on the uniqueness of movements and objects.

Moreover, the significance of employing a multi-window strategy is demonstrated in section III-B. Our complete framework, denoted as Hierarchical NN-W and Hierarchical SF-W, outperforms baselines (Hierarchical NN-O and Hierarchical SF-O) without the multi-window strategy, as shown in Table III. Utilizing more informative data inputs improves accuracies, eliminating the need for the neural network to extract features. Future work will explore an auto-tuning method for optimizing window sizes.

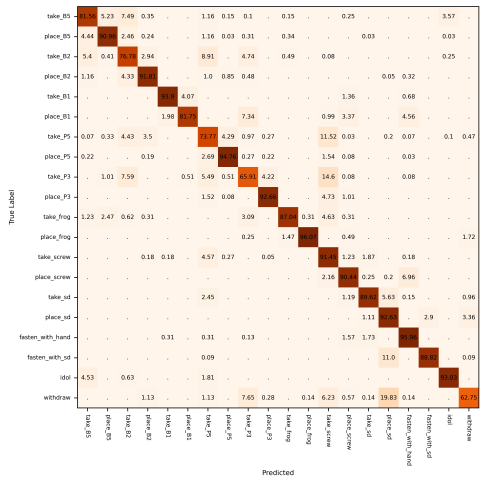
Finally, we apply the LSTM and motion data to conduct qualitative task and action prediction results of our hierarchical deep learning models on teleoperation videos recorded in ROS bags shown in Fig. 6. The intention estimation produces results efficiently with 2 Hz. The ground truth and prediction results are shown below the image sequence. Each pair of frames corresponds to the time at each second. By comparing the results of tasks $D9$ and $D10$, our

TABLE III: The results of accuracy comparisons on different data types, where Hierarchical is abbreviated as Hie.

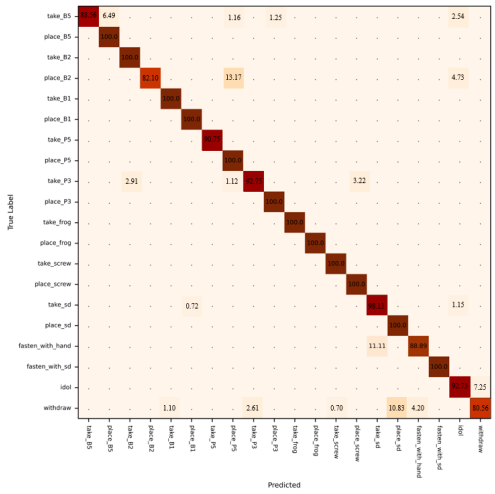
Method | Accuracy	Data Type	Action	Task
Hie-SF-O	egocentric	82.21%	84.57%
Hie-SF-W	egocentric	**86.18%**	**87.33%**
Hie-NN-O	motion	93.82%	95.89%
Hie- NN-W	motion	**95.41%**	**98.25%**



(a) Normalized confusion matrix of task prediction using hierarchical LSTM (Left) and hierarchical Slow-Fast (Right).



(b) Normalized confusion matrix of action prediction using hierarchical Slow-Fast



(c) Normalized confusion matrix of action prediction using hierarchical LSTM

Fig. 5: On the x axis the predictions, on the y the ground truth. Numbers represents the frequency with which samples of a certain class (row) were classified with the label on the corresponding

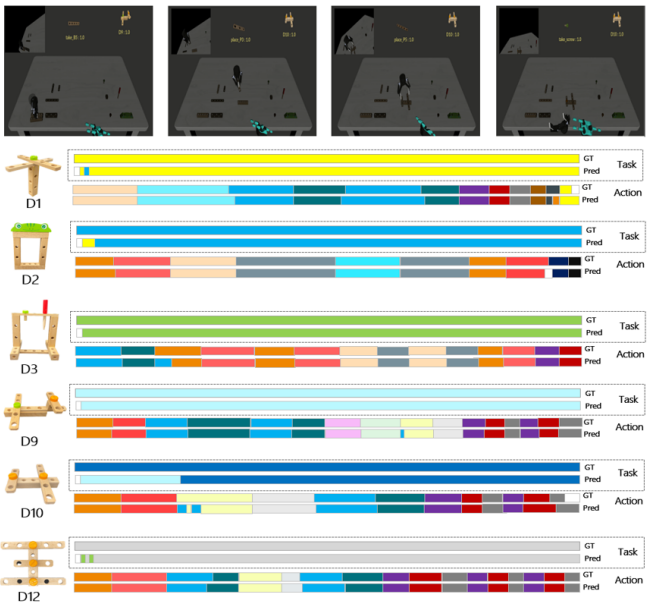


Fig. 6: We select 6 videos from each assembly task. For each video, we show the ground-truth task label, predictions results.

model can sensitively recognize tasks based on the differences in sequential actions that appeared at early stages.

V. Discussion and Conclusion

In this analysis, we find that incorporating hierarchical relations into intention estimation systems enhances prediction performance. This structure encourages top-down layers to share prediction information through hierarchical relations, expediting accurate predictions. Our online testing reveals that high-level intention abstraction increases the stability of low-level inference, reducing switches and improving robot control stability. To summarize, we propose hierarchical deep learning models that consider dependent relations among multi-layer intentions and enforce their hierarchical structure during training. Our model is adaptable for integration with existing neural networks. For hierarchical data input, we introduce a multi-window strategy to mask unnecessary information at each layer, resulting in diverse inputs and embeddings. We demonstrate enhanced inference performance with both motion and vision data compared to independent neural network models in real-time teleoperation Vive systems. In future research, we will explore the intention estimation model's robustness to teleoperation anomalies and the generalization capability of foundation models using zero-shot or few-shot learning.

VI. Acknowledgment

We would like to express our deep and sincere gratitude to Dirk Ruiken and Simon Manschitz for providing the robotic simulation environment and their kind support throughout the project. Also we would like to thank our colleagues for helping us with the data collection and valuable comments.

REFERENCES

[1] K. I. Alevizos, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Physical human–robot cooperation based on robust motion intention estimation," *Robotica*, vol. 38, no. 10, pp. 1842–1866, 2020.

[2] C. Fang, L. Peternel, A. Seth, M. Sartori, K. Mombaur, and E. Yoshida, "Human modeling in physical human-robot interaction: A brief survey," *IEEE Robotics and Automation Letters*, 2023.

[3] R. Wilcox, S. Nikolaidis, and J. Shah, "Optimization of temporal dynamics for adaptive human-robot interaction in assembly manufacturing," *Robotics*, vol. 8, no. 441, pp. 10–15, 2013.

[4] G. Li, Z. Li, and Z. Kan, "Assimilation control of a robotic exoskeleton for physical human-robot interaction," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2977–2984, 2022.

[5] P. Schydlo, M. Rakovic, L. Jamone, and J. Santos-Victor, "Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action sequences prediction," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5909–5914.

[6] 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)*. IEEE, 2018, pp. 1326–1333.

[7] A. Belardinelli, A. R. Kondapally, D. Ruiken, D. Tanneberg, and T. Watabe, "Intention estimation from gaze and motion features for human-robot shared-control object manipulation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 9806–9813.

[8] Z. Huang, Y.-J. Mun, X. Li, Y. Xie, N. Zhong, W. Liang, J. Geng, T. Chen, and K. Driggs-Campbell, "Hierarchical intention tracking for robust human-robot collaboration in industrial assembly tasks," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9821–9828.

[9] S. Manschitz and D. Ruiken, "Shared autonomy for intuitive teleoperation," *ICRA Workshop: Shared Autonomy in Physical Human-Robot Interaction: Adaptability and Trust*, May 2022.

[10] D. Gao, W. Yang, H. Zhou, Y. Wei, Y. Hu, and H. Wang, "Deep hierarchical classification for category prediction in e-commerce system," *arXiv preprint arXiv:2005.06692*, 2020.

[11] T. B. Sheridan, "Teleoperation, telerobotics and telepresence: A progress report," *Control Engineering Practice*, vol. 3, no. 2, pp. 205–214, 1995.

[12] L. Rozo, S. Calinon, D. G. Caldwell, P. Jimenez, and C. Torras, "Learning physical collaborative robot behaviors from human demonstrations," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 513–527, 2016.

[13] W. Yu, R. Alqasemi, R. Dubey, and N. Pernalete, "Telemanipulation assistance based on motion intention recognition," in *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE, 2005, pp. 1121–1126.

[14] A. D. Dragan and S. S. Srinivasa, "A policy-blending formalism for shared control," *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 790–805, 2013.

[15] K. Hauser, "Recognition, prediction, and planning for assisted teleoperation of freeform tasks," *Autonomous Robots*, vol. 35, pp. 241–254, 2013.

[16] D. Aarno and D. Kragic, "Motion intention recognition in robot assisted applications," *Robotics and Autonomous Systems*, vol. 56, no. 8, pp. 692–705, 2008.

[17] A. K. Tanwani and S. Calinon, "A generative model for intention recognition and manipulation assistance in teleoperation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 43–50.

[18] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[19] A. K. Tanwani and S. Calinon, "Learning robot manipulation tasks with task-parameterized semitied hidden semi-markov model," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 235–242, 2016.

[20] C. Zhu, Q. Cheng, and W. Sheng, "Human intention recognition in smart assisted living systems using a hierarchical hidden markov model," in *2008 IEEE International Conference on Automation Science and Engineering*. IEEE, 2008, pp. 253–258.

[21] Y. Li and S. S. Ge, "Human–robot collaboration based on motion intention estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 3, pp. 1007–1014, 2013.

[22] E. De Momi, L. Kranendonk, M. Valenti, N. Enayati, and G. Ferrigno, "A neural network-based approach for trajectory planning in robot–human handover tasks," *Frontiers in Robotics and AI*, vol. 3, p. 34, 2016.

[23] C. Lea, R. Vidal, and G. D. Hager, "Learning convolutional action primitives for fine-grained action recognition," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1642–1649.

[24] W. Lu, Z. Hu, and J. Pan, "Human-robot collaboration using variable admittance control and human intention prediction," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2020, pp. 1116–1121.

[25] J. Bandouch and M. Beetz, "Tracking humans interacting with the environment using efficient hierarchical sampling and layered observation models," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, 2009, pp. 2040–2047.

[26] B. Hayes and B. Scassellati, "Autonomously constructing hierarchical task networks for planning and human-robot collaboration," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5469–5476.

[27] S. Holtzen, Y. Zhao, T. Gao, J. B. Tenenbaum, and S.-C. Zhu, "Inferring human intent from video by sampling hierarchical plans," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1489–1496.

[28] J.-H. Han, S.-H. Choi, and J.-H. Kim, "Interactive human intention reading by learning hierarchical behavior knowledge networks for human-robot interaction," *ETRI Journal*, vol. 38, no. 6, pp. 1229–1239, 2016.

[29] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6202–6211.

[30] L. R. Medsker and L. Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, no. 64-67, p. 2, 2001.

[31] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[33] F. Abi-Farraj, T. Osa, N. P. J. Peters, G. Neumann, and P. R. Giordano, "A learning-based shared control architecture for interactive task execution," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 329–335.

[34] J. T. Feddema and O. R. Mitchell, "Vision-guided servoing with feature-based trajectory generation (for robots)," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 5, pp. 691–700, 1989.

[35] T. Marcucci, M. Petersen, D. von Wrangel, and R. Tedrake, "Motion planning around obstacles with convex optimization," *arXiv preprint arXiv:2205.04422*, 2022.

[36] L. X. Shi, A. Sharma, T. Z. Zhao, and C. Finn, "Waypoint-based imitation learning for robotic manipulation," *arXiv preprint arXiv:2307.14326*, 2023.

[37] L. Rabiner and B. Juang, "An introduction to hidden markov models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.