2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

DuCAS: a knowledge-enhanced dual-hand compositional action segmentation method for human-robot collaborative assembly

Hao Zheng*, Regina Lee, Huachang Liang, Yuqian Lu, Xun Xu

***Abstract*— Recognising and tracking human actions from videos is crucial for human-robot collaborative assembly (HRCA). However, traditional action segmentation methods suffer from limited scene adaptability, partly because they conceptualise actions as unified verb-object entities with complete semantics. To overcome this, we propose a compositional action segmentation method. Following the human-robot shared assembly taxonomy, we deconstruct an assembly action into four elements: *action verb*, *manipulated object*, *target object* and *tool*. Our approach employs individual segmentation models for each action element, and then integrates general knowledge from large language models and domain-specific knowledge from predefined rules to form semantic-complete actions. Our method's emphasis on general action elements and a modular design endows it with greater flexibility and adaptability than traditional approaches. Another attribute of our method is its capability to segment actions of each hand concurrently, facilitating more nuanced HRCA. Comparative experiments validate the superiority of our method over traditional action segmentation methods. More details can be found at `https://github.com/LISMS-AKL-NZ/DuCAS`.**

I. INTRODUCTION

Human-robot collaborative assembly (HRCA) is essential for human-centric and personalised manufacturing [1], [2]. It enhances assembly efficiency and flexibility, and alleviates the physical and cognitive load on humans through effective task allocation and prompt robotic assistance [3], [4], [5]. A prerequisite of effective HRCA is the capability of robots to understand human actions, as the actions convey critical information about human intention and task progression.

In the context of HRCA, the primary objective of action understanding is the precise recognising and tracking of human actions. To this end, it is crucial for HRCA to explore action segmentation techniques that identify and segment all action instances within a long video sequence [6]. Traditional action segmentation methods [7], which conceptualise actions as unified verb-object entities with complete semantics, have limited adaptability across various application scenarios, and no existing method addresses dual-hand assembly actions. To overcome these limitations, we propose DuCAS, a dual-hand compositional action segmentation method. Following the Human-Robot Shared Assembly Taxonomy (HR-SAT) [8], we deconstruct an assembly action into four elements: *action verb*, *manipulated object*, *target object* and *tool*. DuCAS employs four segmentation models to separately segment these elements, which are then integrated to form semantic-complete actions. DuCAS distinguishes itself from traditional action segmentation methods through offering enhanced flexibility by its modular design, and broader applicability by focusing on the segmentation of general action elements rather than the specific unified semantic entities.

However, simply combining the outputs of the four models can lead to nonsensical and illogical combinations due to the error accumulation from four models, thereby reducing the overall action segmentation performance. To mitigate this, we incorporate both general and domain-specific knowledge to systematically refine the output combinations, thereby achieving error compensation. Initially, we employ an large language model (LLM) [9] to provide general knowledge used to eliminate patently nonsensical combinations. Subsequently, rule-based domain-specific knowledge is applied to exclude combinations that are illogical for particular application contexts. This two-stage strategy reflects our "*rationality before speciality*" philosophy, akin to the Piaget's theory of human cognitive development [10]. Integrating general knowledge from an LLM ensures fundamental rationality in the final action segmentation outputs. Incorporating domain-specific knowledge boosts DuCAS's performance in specific domains, which also contributes to a paradigm for adapting this method to various applications.

This paper is organised as follows. Section 2 provides a brief review of research on human-robot collaborative assembly and action segmentation. Section 3 presents the method for the proposed DuCAS. Section 4 introduces the experimental results of our method. Lastly, Section 5 concludes this paper.

II. RELATED WORK

A. Human-robot collaborative assembly

In HRCA, humans and robots engage in close collaboration within a dynamic environment, where robots adjust their actions in real-time to complement human intentions and meet task demands. Essential to this is robots' capability of perceiving the assembly environment, which has been facilitated by advancements in computer vision technology [11], [12], [13]. Perceiving human actions, as conveyors of intentions and task dynamics, is critical for assembly environment perception. Therefore, integrating human action perception into HRCA has garnered increasing attention recently. A few recent studies have focused on reducing human

*This study was supported by the China Scholarship Council (Grant No. 202006420001).

Hao Zheng (corresponding author), Regina Lee, Huachang Liang, Yuqian Lu, Xun Xu are with the Department of Mechanical and Mechatronics Engineering, The University of Auckland, New Zealand `hzhe951@aucklanduni.ac.nz`; `klee702@aucklanduni.ac.nz` ; `hlia981@aucklanduni.ac.nz` ; `yuqian.lu@auckland.ac.nz`; `x.xu@auckland.ac.nz`

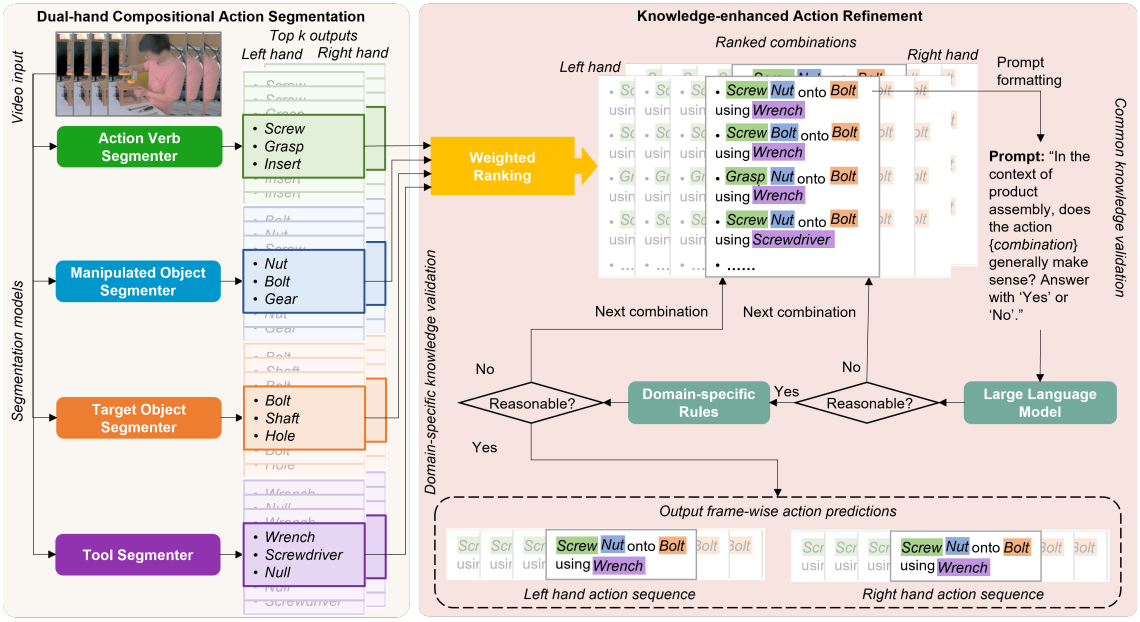

Fig. 1. Overview of DuCAS. It has dual-hand compositional action segmentation models and a two-stage knowledge-enhanced action refinement.

fatigue and enhancing assembly efficiency through improved human-robot task allocation and robotic assistive action planning, drawing on the insights gained from human action perception [3], [6], [14], [15], [16]. However, these studies only focus on a broad view of human actions, neglecting the detailed individual hand actions. This limits robots from providing attentive assistance, such as delivering a tool to the specific hand. Moreover, existing studies predominantly utilise action recognition techniques that take a video clip as input and output an action class prediction. However, in the context of frame-wise action recognition of a long video, a few erroneous frames can lead to substantial inaccuracies in the action segment sequence, diverging from the demands of real-world applications. Therefore, it is necessary to investigate action segmentation techniques, that refine frame-wise action predictions by considering the surrounding frames, for HRCA.

B. Action segmentation

Action segmentation aims at temporally locating and recognizing human action segments (constituted by consecutive frames with same action labels) in long untrimmed videos [17]. Current action segmentation approaches commonly employ temporal networks on the pre-extracted frame-by-frame motion features to capture the features' dependencies across the entire video for accurate frame-wise action classification [18], [19], [20], [17], [21]. However, these methods label an action as a unified verb-object entity with complete semantics, e.g., "*Screw Nut onto Bolt using Wrench*", encountering two primary limitations: 1. low action transferability across scenarios, and 2. insufficient exploration and utilisation of semantic relationships between action verbs and interacted objects. To address these issues, we propose a compositional action segmentation approach. This approach firstly segments action elements, e.g., *screw*, *nut*, *bolt*, and *wrench*, and then integrates them to get the semantic-complete action segmentation result.

However, simply combining the segmented action elements can introduce a wide range of unreasonable combinations. Thus, it is crucial to employ both common and domain-specific knowledge to refine these combinations. In manufacturing, the representation, modelling, and application of domain-specific knowledge have long been subjects of exploration [8], [22], [23]. Recent advancements demonstrate that LLMs can provide common knowledge which is applicable to various applications [9], [24]. For instance, [25] leveraged an LLM to refine scene graph generation, aligning them with common sense intuitions. [26] employed an LLM to introduce a commonsense prior into a Monte Carlo Tree Search algorithm to enhance large scale task planning.

III. METHOD

Fig. 1 shows the overall framework of DuCAS. Initially, the video is input into four dual-hand action element segmentation models to predict the top k actions and their confidence scores for each hand in every frame. Subsequently, these action elements are integrated to form a semantic-complete action, creating a list of combination candidates ranked by weighted sorting. Each combination is verified in a two-stage process, starting with the highest-ranked: first through general knowledge validation, then domain-specific knowledge validation. For each frame, if a combination does not pass a stage, it checks the next ranked combination. Thus, the first combination passing both stages becomes the final output. Although DuCAS has not been applied to an HRCA system in this study, we consider it a key enabling technology that enhances robots' ability to interpret human assembly actions, thereby facilitating collaboration.

A. Compositional dual-hand action element segmentation

We designed four action element segmentation models with uniform architectures, as depicted in Fig. 2, differing solely in the final layer's output dimension to align with the number of classes for each action element. Recognizing the significance of hand-object interaction features and general motion features in the scene in assembly action understanding, our model integrates graph attention networks (GAT) [27] to capture these interaction features, alongside I3D networks [28] for general motion features. The concatenated features are then processed through a temporal convolutional network (TCN) [18] to predict frame-wise action element classes.

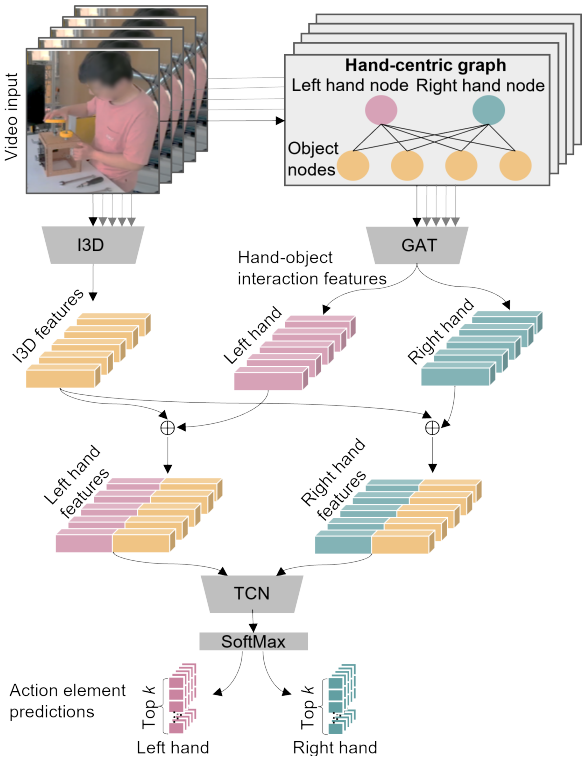

Fig. 2. Architecture of the compositional action element segmentation model.

a) ***Hand-object interaction features extraction****:* A video is modeled as a hand-centric graph $g = \{G_1, G_2, \ldots, G_T\}$, where $G_t = (V_t, E_t)$ corresponds to the t-th frame and T denotes the video length. V_t includes nodes for the left hand $v_{lh,t}$, right hand $v_{rh,t}$, and objects $V_{obj,t}$. Hand-object edges $E_t = \{e_{hand-obj,t} | e_{hand-obj,t}$ connects $v_{hand,t}$ and $v_{obj,t}$ for $hand \in \{lh, rh\}$ and each object $obj\}$. This graph structure, by focusing solely on hand-object connections, efficiently captures the dynamic interactions between each hand and objects within the scene. Node attributes, $x_{i,t} = [loc_{i,t}, emb_{i,t}]$, combine the spatial location $loc_{i,t}$ and class embedding $emb_{i,t}$ of node i at frame t.

A multi-stage GAT is used to update the node features following Equation 1, which are then concatenated along the edges E_t to capture hand-object interactions:

$$h_{i,t} = \sigma\left(\sum_{j \in N(i)} \alpha_{ij,t} W_t x_{j,t}\right) \tag{1}$$

$$H_{hand,t} = \bigoplus_{i \in N(hand)} h_{i,t}, \quad hand \in \{lh, rh\}. \tag{2}$$

where $h_{i,t}$ is the output feature of i-th node at frame t, $N(i)$ denotes the neighbor nodes of node i, α is the learnable attention score, W is the learnable weight matrix, and $H_{hand,t}$ is the concatenated feature for $hand$ at frame t.

b) ***Motion feature extraction****:* The I3D network [28] is utilised to capture the general motion features $X_{mot} = \{x_{mot,1}, x_{mot,2}, \ldots, x_{mot,T}\}$.

c) ***Temporal dynamics capturing****:* The hand-object interaction features are concatenated with the motion features to form the frame features, following:

$$F_{hand,t} = H_{hand,t} \oplus x_{mot,t}, \quad hand \in \{lh, rh\}. \tag{3}$$

Then, the frame features F are input into a TCN block to capture the long dependencies across the video. Following [18]'s design, the TCN block is structured into L layers of 1D dilated convolutions, with each layer's dilation factor increasing exponentially. After the TCN block, a SoftMax layer is applied to produce frame-wise action predictions for both hands. For each frame, the top k predictions are selected as the output, resulting in $Y_{lh} = \left\{\left[y^1_{lh,1}, \ldots, y^k_{lh,1}\right], \ldots, \left[y^1_{lh,T}, \ldots, y^k_{lh,T}\right]\right\}$ and $Y_{rh} = \left\{\left[y^1_{rh,1}, \ldots, y^k_{rh,1}\right], \ldots, \left[y^1_{rh,T}, \ldots, y^k_{rh,T}\right]\right\}$.

To enhance segment-level accuracy, we design a composite loss function that comprises of a classification loss L_{cls}, a smoothing loss L_{T-MSE}, and a segmental edit distance loss $L_{seg-edit}$:

$$L = L_{cls} + \lambda L_{T-MSE} + \mu L_{seg-edit} \tag{4}$$

$$L_{cls} = \frac{1}{T} \sum_{hand \in \{lh,rh\}} \sum_{t} -\log(y_{hand,t,c}) \tag{5}$$

$$L_{T-MSE} = \frac{1}{TC} \sum_{hand \in \{lh,rh\}} \sum_{t,c} \tilde{\Delta}^2_{hand,t,c} \tag{6}$$

$$\tilde{\Delta}_{hand,t,c} = \begin{cases} \Delta_{hand,t,c} & \text{if } \Delta_{hand,t,c} \leq \tau \\ \tau & \text{otherwise} \end{cases} \tag{7}$$

$$\Delta_{hand,t,c} = |\log y_{hand,t,c} - \log y_{hand,t-1,c}| \tag{8}$$

$$L_{seg-edit} = \frac{1}{TC} \sum_{hand \in \{lh,rh\}} \text{Levenshtein}(\bar{S}_{hand}, S_{hand}) \tag{9}$$

where $y_{hand,t,c}$ is the predicted probability for the ground truth action label c of the $hand$ at frame t. C is the total number of classes. $\bar{S}_{hand}$ and S_{hand} are the predicted and ground-truth segment sequences, respectively. Levenshtein$(\cdot)$ calculates the Levenshtein distance between two sequences.

B. Knowledge-enhanced action refinement

a) ***Weighted ranking****:* For the t-th frame, let $Y^i_{t,e}$ and $S^i_{t,e}$ denote the i-th prediction and its confidence score for the e-th action element, respectively, with $e \in \{1, 2, 3, 4\}$ and $i \in \{1, 2, \ldots, k\}$. The total number of combinations is k^k. The aggregate confidence score for the j-th combination in frame t is determined using Equation 10. Subsequently, action element combinations are ranked by descending the aggregate confidence scores to form a candidate list per frame, following Equation 11:

$$AS^j_t = \sum_{e=1}^{4} \omega_e \cdot S^j_{(t,e)} \tag{10}$$

$$\text{List}_t = \text{sort}_{desc}\{\overline{AS}^1_t, \overline{AS}^2_t, \ldots, \overline{AS}^{k^k}_t\} \tag{11}$$

where $\overline{AS}^j_t$ denotes the action element combination corresponding to the AS^j_t, and ω_e denotes the weight for action element e.

b) ***General knowledge validation****:* Firstly, each combination candidate is translated into a semantic string: *Combination* = "{*Action verb*} {*Manipulated object*} *onto* {*Target object*} *using* {*Tool*}." Then, the prompt is formed as a string: "*In the context of product assembly, does the action combination generally make sense? Answer with 'Yes' or 'No'.*" Finally, the prompt is fed into an LLM.

c) ***Domain-specific knowledge validation****:* In this phase, two straightforward rules are established based on prior knowledge:

Rule 1: *The assembly action must be meaningful for the assembled product*. This requires the prior knowledge of the possible assembly actions for the assembled product.

Rule 2: *Continuity must be maintained within each assembly action*. This rule aims to eliminate sporadic predictions that interrupt the continuity of identical predictions across successive frames.

The pseudo code for the rule-based domain-specific knowledge validation can be found in Algorithm 1. Here, a threshold ϵ is utilised to control the length of interrupting frames. A $segment$ is defined as a sequence of consecutive frames with identical action predictions.

IV. Experiments

A. Dataset

The performance of DuCAS was evaluated on a subset of HA-ViD (a human assembly video dataset) [29], which includes object bounding boxes and action labels, with the addition of four action element labels we created. This sub-dataset features 116 videos from three views (side, front and top), spanning 219 atomic action classes, 11 action verb classes, 29 manipulated object classes, 26 target object classes and 5 tool classes. The average length of videos is 1350 frames. The sub-dataset was split into a training set of 98 videos and a testing set of 18 videos. Although HA-ViD is not collected in a HRCA scenario, it offers valuable data that facilitates robots in understanding human assembly actions.

B. Implementation details

In DuCAS, we stack four GAT blocks, with each block comprising of four layers. Object nodes are constructed using the ground truth bounding box labels. The TCN block employs three dilated convolution layers with dilation factors that double progressively, and incorporates a dropout rate of 0.2 after each layer. The loss function utilises parameters τ, λ, and μ, assigned values of 4, 0.15, and 1, respectively. We use the Adam optimiser with a learning rate of 0.0001, over a total of 100 epochs. For the output of compositional action segmentation, we select the top 3 predictions as the result. Through experiments, the four weights for the action verb, manipulated object, target object, and tool are determined as 2.5, 5, 2, and 1, respectively. For the LLM, we employ ChatGPT 4.0 [9]. In Algorithm 1, we set $\epsilon = 4$.

Algorithm 1 Domain-specific knowledge validation

Input:
$List_t$: List of predicted actions for each frame t
$valid\_actions$: Set of valid actions for the given product
Output:
$refined\_sequence$: Refined action prediction sequence for the video
Initialise: An empty list $predicted\_sequence$
for frame t in video **do**
  for action i in $List_t$ **do**
    if $List_t[i]$ in $valid\_actions$ **then**
      Append $List_t[i]$ to $predicted\_sequence$
    end if
  end for
end for
Get $segments$ from $predicted\_sequence$
Copy $predicted\_sequence$ to $refined\_sequence$
for each $segment$ in $segments$ **do**
  if $previous\_segment$ = $next\_segment$ & $length(segment) < \epsilon$ & $segment \neq null$ **then**
    Update $refined\_sequence$ by changing $segment$ action predictions to $previous\_segment$ action predictions
  end if
end for

C. Comparison with the state-of-the-art

We compare DuCAS with MS-TCN [18], DTGRM [17], ASRF [19], and C2F-TCN [21], on the dataset. We report the frame-wise accuracy (Acc), segmental edit distance (Edit), and segmental F1 score at overlapping thresholds of 10%, 25%, and 50% (F1@{10, 25, 50}) in Table I. Table I demonstrates that DuCAS surpasses competing methods in performance. Additionally, DuCAS uniquely supports the simultaneous segmentation of dual-hand actions, a feature not present in other models. Other models were trained on the sub-dataset of each hand separately.

TABLE I

COMPARISON WITH THE STATE-OF-THE-ART ON A SUBSET OF HA-VID.

Method	View	Left hand atomic action					Right hand atomic action					Single model for dual-hand action
		Acc	Edit	F1@{10, 25, 50}			Acc	Edit	F1@{10, 25, 50}			
MS-TCN[18]	Side	42.2	26.17	28.66	24.39	12.2	17.37	4.98	6.64	4.65	2.66	✗
	Front	41.28	31.65	28.63	21.89	12.63	16.75	5.01	6.63	4.97	3.31	
	Top	42.79	27.45	28.11	**27.03**	**14.59**	18.07	6.63	8.47	6.51	4.56	
DTGRM[17]	Side	41.81	**30.39**	31.2	23.4	14.89	18.34	18.94	16.78	13.05	6.99	✗
	Front	44.41	32.58	30.29	22.86	9.14	27.59	25.57	**29.21**	**21.07**	10.72	
	Top	44.04	**28.34**	**28.99**	27.03	**14.59**	27.57	23.24	26.09	20.79	14.37	
ASRF[19]	Side	25.78	22.16	18.18	12.65	5.53	11.88	15.33	16.39	13.58	7.96	✗
	Front	14.47	18.47	18.7	11.43	3.12	16.79	23.01	24.14	18.51	10.46	
	Top	16.94	20.41	15.81	10.31	5.5	22.44	23.34	24.62	21.98	13.19	
C2F-TCN[21]	Side	51	9.27	12.9	11.61	5.16	20.47	10.85	9.7	7.88	4.24	✗
	Front	46.62	8.11	10.9	8.48	6.06	26.32	15.02	16.96	14.96	6.98	
	Top	**56.45**	11.13	11.35	8.51	2.84	24.8	11.62	12.32	9.97	5.87	
DuCAS(ours)	Side	**53.11**	29.73	**35.44**	**29.37**	**18.97**	**37.03**	**29.08**	**33.31**	**28.44**	**19.05**	✓
	Front	**55.79**	**33.09**	**33.19**	**27.97**	**16.27**	**36.65**	**24.34**	26.22	20.28	**11.61**	
	Top	50.15	26.53	27.14	24.66	13.73	**30.18**	**30.62**	**28.94**	**24.51**	**16.58**	

TABLE II

PERFORMANCE OF COMPOSITIONAL ACTION SEGMENTATION FOR ACTION VERB (AV), MANIPULATED OBJECT (MO), TARGET OBJECT (TO), AND TOOL (TL). THE TOP 1 PREDICTION IS SELECTED AS THE RESULT. DUE TO THE PAGE LIMIT, WE ONLY REPORT THE AVERAGE RESULT OVER THREE VIEWS.

Method	Left hand atomic action					Right hand atomic action				
	Acc	Edit	F1@{10, 25, 50}			Acc	Edit	F1@{10, 25, 50}		
AV	56.97	41.86	42.62	38.63	25.35	60.22	55.373	58.8	53.33	39.82
MO	62.84	43.32	41.49	36	24.34	46.35	38.41	36.47	30.24	19.71
TO	82.39	47.60	46.41	44.5	34.54	59.71	54.17	45.5	40.46	26.83
TL	97.42	90.25	91.98	90.94	84.77	96.68	78.81	82.25	80.55	69.84

TABLE III

PERFORMANCE OF DUCAS WITHOUT KNOWLEDGE-ENHANCED ACTION REFINEMENT (W/O AR), WITHOUT WEIGHTED RANKING MECHANISM BUT WITH FULL TWO-STEP VALIDATION (W/O WR), WITH ONLY GENERAL KNOWLEDGE VALIDATION (GK ONLY). DUE TO THE PAGE LIMIT, WE ONLY REPORT THE AVERAGE RESULT OVER THREE VIEWS.

Method	Left hand atomic action					Right hand atomic action				
	Acc	Edit	F1@{10, 25, 50}			Acc	Edit	F1@{10, 25, 50}		
DuCAS	**53.02**	29.78	**31.92**	**27.33**	**16.32**	**34.62**	28.01	29.49	24.41	15.75
w/o AR	42.88	**34.31**	31.27	26.80	15.73	20.61	27.48	29.44	24.49	16.04
w/o WR	51.32	27.89	28.83	26.55	15.54	30.69	**28.36**	**30.66**	**26.9**	**17.88**
GK only	47.72	30.62	30.79	26.77	15.15	25.21	25.38	28.78	23.60	14.37

D. Ablation study and discussion

DuCAS has two primary modules: compositional action segmentation and knowledge-enhanced action refinement. Initially, we evaluate the segmentation performance on each action elements, detailed in Table II. Subsequently, we assess the contribution of each component within the knowledge-enhanced action refinement module, with findings presented in Table III.

Table II demonstrates that segmentation performance for tools and action verbs significantly surpasses that of manipulated and target objects. This disparity may stem from the broader diversity of manipulated and target objects (see Section IV-A), which elevates the segmentation complexity. Furthermore, the segmentation of action verbs and tools predominantly relies on motion and visual cues, whereas discerning manipulated and target objects may require more intricate interactional information and reasoning processes. This variance in segmentation performance among different action elements underscores the need for tailored attention to these elements during their integration into a semantic-complete action. To address this, we introduced a weighted ranking approach (see Section III-B), the effectiveness of which is shown in Table III. However, this approach is but a preliminary trial. Future endeavours should focus on developing more nuanced methodologies that leverage both the confidence of each action element prediction and their inherent semantic relationships.

Comparing Table II with DuCAS (w/o AR) in Table III highlights a performance drop due to the error accumulation from simply combining four models' results. This underscores the necessity for leveraging knowledge to refine the action element combinations. Table III demonstrates the efficacy of the two-stage, knowledge-enhanced action refinement module in improving the method's performance, especially on Acc. Even without weighted ranking (w/o WR), comparable results were achieved because it still contains the two-stage validation, with only varied ranking lists. The

main difference lies in Acc, because the selection of optimal weights in Equation 10 was based on Acc. The integration of general knowledge validation (GK only) markedly improves outcomes compared to methods without such validation (w/o AR), highlighting the utility of general knowledge in specialised contexts. Furthermore, such integration enables the compositional action segmentation to yield reasonable results applicable to various scenarios without LLM finetuning. However, a future study can be implementing the LLMs that are customised for a target domain. The benefits derived from incorporating domain-specific knowledge emphasises its indispensable role, despite the rudimentary hand-crafted rules. We believe that exploring automated techniques for domain-specific knowledge engineering is a promising future research direction.

V. CONCLUSIONS

In this work, we propose DuCAS, a dual-hand compositional action segmentation method, for HRCA. DuCAS processes videos to output the top k predictions of each action element, including *action verb*, *manipulated object*, *target object*, and *tool*, per frame. With the action element predictions, we refine their combinations using both general and domain-specific knowledge to generate coherent assembly action sequences. Our experiments demonstrate DuCAS's superiority over existing methods. Furthermore, compared with traditional action segmentation methods, compositional action segmentation has greater flexibility and adaptability for industrial applications. Our knowledge-enhanced action refinement process in DuCAS exemplifies the application of compositional action segmentation in practical settings. We aspire that our "*rationality before specialty*" approach to incorporate general and domain-specific knowledge can inspire researchers in the era of LLMs. Future work will focus on breaking through DuCAS's limitations, including real-time segmentation capabilities, customizing LLMs for manufacturing, and developing automated techniques for domain-specific knowledge engineering.

REFERENCES

[1] Y. Lu, H. Zheng, S. Chand, W. Xia, Z. Liu, X. Xu, L. Wang, Z. Qin, and J. Bao, "Outlook on human-centric manufacturing towards industry 5.0," *Journal of Manufacturing Systems*, vol. 62, pp. 612–627, 1 2022.

[2] Q. Lv, R. Zhang, T. Liu, P. Zheng, Y. Jiang, J. Li, J. Bao, and L. Xiao, "A strategy transfer approach for intelligent human-robot collaborative assembly," *Computers & Industrial Engineering*, vol. 168, p. 108047, 6 2022.

[3] A. Raatz, S. Blankemeyer, T. Recker, D. Pischke, and P. Nyhuis, "Task scheduling method for hrc workplaces based on capabilities and execution time assumptions for robots," *CIRP Annals*, vol. 69, pp. 13–16, 2020.

[4] N. Lucci, A. Monguzzi, A. M. Zanchettin, and P. Rocco, "Workflow modelling for human–robot collaborative assembly operations," *Robotics and Computer-Integrated Manufacturing*, vol. 78, p. 102384, 12 2022.

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

[6] H. Zheng, S. Chand, A. Keshvarparast, D. Battini, and Y. Lu, "Video-based fatigue estimation for human-robot task allocation optimisation," in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*. IEEE, 8 2023, pp. 1–6.

[7] G. Ding, F. Sener, and A. Yao, "Temporal action segmentation: An analysis of modern techniques," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, pp. 1011–1030, 2 2024.

[8] R. K.-J. Lee, H. Zheng, and Y. Lu, "Human-robot shared assembly taxonomy: A step toward seamless human-robot knowledge transfer," *Robotics and Computer-Integrated Manufacturing*, vol. 86, p. 102686, 4 2024.

[9] OpenAI, "Gpt-4 technical report," 3 2023.

[10] P. Barrouillet, "Theories of cognitive development: From piaget to today," *Developmental Review*, vol. 38, pp. 1–12, 12 2015.

[11] V. Kakani, V. H. Nguyen, B. P. Kumar, H. Kim, and V. R. Pasupuleti, "A critical review on computer vision and artificial intelligence in food industry," *Journal of Agriculture and Food Research*, vol. 2, p. 100033, 12 2020.

[12] L. Zhou, L. Zhang, and N. Konz, "Computer vision techniques in manufacturing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, pp. 105–117, 1 2023.

[13] H. Zheng, G. Cheng, Y. Li, and C. Liu, "A fault diagnosis method for planetary gear under multi-operating conditions based on adaptive extended bag-of-words model," *Measurement*, vol. 156, p. 107593, 5 2020.

[14] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, "Transferable two-stream convolutional neural network for human action recognition," *Journal of Manufacturing Systems*, vol. 56, pp. 605–614, 7 2020.

[15] M. Terreran, L. Barcellona, and S. Ghidoni, "A general skeleton-based action and gesture recognition framework for human–robot collaboration," *Robotics and Autonomous Systems*, vol. 170, p. 104523, 12 2023.

[16] J. Koch, L. Büsch, M. Gomse, and T. Schüppstuhl, "A methods-time-measurement based approach to enable action recognition for multi-variant assembly in human-robot collaboration," *Procedia CIRP*, vol. 106, pp. 233–238, 2022.

[17] D. Wang, D. Hu, X. Li, and D. Dou, "Temporal relational modeling with self-supervision for action segmentation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 2729–2737, 5 2021.

[18] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation." IEEE, 6 2019, pp. 3570–3579.

[19] Y. Ishikawa, S. Kasai, Y. Aoki, and H. Kataoka, "Alleviating over-segmentation errors by detecting action boundaries." IEEE, 1 2021, pp. 2321–2330.

[20] Z. Wang, Z. Gao, L. Wang, Z. Li, and G. Wu, "Boundary-aware cascade networks for temporal action segmentation," 2020, pp. 34–51.

[21] D. Singhania, R. Rahaman, and A. Yao, "C2f-tcn: A framework for semi- and fully-supervised temporal action segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–18, 2023.

[22] S. Lemaignan, A. Siadat, J.-Y. Dantan, and A. Semenenko, "Mason: A proposal for an ontology of manufacturing domain." IEEE, 2006, pp. 195–200.

[23] H. Zhang, U. Roy, and Y.-T. T. Lee, "Enriching analytics models with domain knowledge for smart manufacturing data analysis," *International Journal of Production Research*, vol. 58, pp. 6399–6415, 10 2020.

[24] H. Touvron and et al., "Llama 2: Open foundation and fine-tuned chat models," 7 2023.

[25] B. Jiang, Z. Zhuang, and C. J. Taylor, "Enhancing scene graph generation with hierarchical relationships and commonsense knowledge," 11 2023.

[26] Z. Zhao, W. S. Lee, and D. Hsu, "Large language models as commonsense knowledge for large-scale task planning," 5 2023.

[27] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 10 2017.

[28] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset." IEEE, 7 2017, pp. 4724–4733.

[29] H. Zheng, R. Lee, and Y. Lu, "HA-vid: A human assembly video dataset for comprehensive assembly knowledge understanding," in *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. [Online]. Available: https://openreview.net/forum?id=DILUIcDmU9