2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Multi-modal Affordance Planner with Temporal-Context Action Policy for Long-Horizon Bimanual Robot Manipulation

Ji-Heon Oh[1], Danbi Jung[1], Ismael Espinoza[1], Yong-Hyeok Choi[1], YoungOuk Kim[2], Dongin Shin[2], JongSul Moon[2], Wonha Kim[3], Tae-Seong Kim*[1].

***Abstract*— Bimanual robot manipulation for long-horizon (LH) tasks is crucial for the practical use of humanoids, but it struggles with robust planning and generalization. Approaches based on Task and Motion Planning (TAMP), transformers, and Large Language Models (LLMs) suffer from critical limitations, including costly human demonstrations, task planner hallucination, and unsatisfactory generalization performance. To address these challenges, this paper introduces the Multi-modal Affordance Planner with Temporal-Context Action Policy (MAP-TCA), a novel hierarchical framework that learns and performs diverse bimanual long-horizon (LH) tasks by generating action plans from MAP. The MAP-TCA consists of a planner based on Bimanual Robot Manipulation Retrieval-Augmented Generation (Bi-RAG)-enhanced Large-Language Model (LLM) and a low-level Temporal Context Action Policy (TCA). With multimodal inputs including vision, language, and affordance for primitive action demonstration, Bi-RAG generates a Primitive Action (PA)-specific embedded space. Then, MAP generates LH plans, LH demonstrations, and reward functions within the PA-specific embedded space, thereby mitigating hallucinations and reducing training cost. The generated plan, demos, and rewards then guide TCA, which learns the LH tasks via behavior cloning (BC) and online fine-tuning. We demonstrate that the proposed MAP-TCA achieves an average success rate of 86.75%, comparable to a baseline model, TCA, which is trained extensively on direct human demonstrations and manually designed rewards. Our work presents a scalable and generalizable solution for complex bimanual LH manipulation, significantly reducing the dependency on human supervision**

I. INTRODUCTION

Bimanual robot manipulation is essential for humanoid robots to perform complex tasks in practice for life-care and medical-care services [1]–[3]. These services frequently involve executing long-horizon (LH) tasks that comprise multiple short-horizon primitive actions (PAs), such as grasping, relocating, and opening doors [1]–[3]. The inherent complexity of these tasks presents a twofold challenge. Firstly, LH contexts drastically complicate the planning process, while diverse environmental conditions create a complex action-observation space. Secondly, Task and Motion Planning (TAMP) represents a foundational hierarchical approach for these problems, typically using a high-level module to decompose an LH task into a sequence of PAs and a low-level module to execute them. For instance, TAMP methods such as Bimanual Keypose-conditioned Consistency Policy (BiKC) [4], Hierarchical Deep Relational Imitation Learning (HDR-IL) [5], extended tree search for TAMP (eTAMP) [6], Learning and Abstraction with Guidance (LEAGUE) [7], and Relay Policy Learning (RPL) [8] employ high-level planners to logically decompose complex LH tasks, such as assembling and cleaning, that demand precise geometric and logical reasoning. Despite their successes, these hierarchical methods share critical limitations. One primary limitation is their heavy reliance on human demonstrations, which involve costly, labor-intensive processes for acquiring human data and hand-crafting reward functions, thereby fundamentally constraining generalization to novel scenarios [4]–[8]. In addition, simultaneously training planners with action policies often leads to convergence issues and suboptimal outcomes [8]. Finally, TAMP frameworks exhibit execution failures due to missing temporal context, as their low-level modules, which typically rely on Multi-Layer Perceptron (MLP) policies or classical dynamics, fail to robustly execute actions that require a deep understanding of the LH task's temporal context.

Recently, Large Language Models (LLMs) have emerged as an alternative to conventional high-level planners in robotics. By leveraging vast pre-trained knowledge, LLMs excel at decomposing LH tasks from natural language descriptions. For instance, frameworks such as Language-model-based Bimanual Orchestration (LABOR) [9], Plan-Seq-Learn [10], TidyBot [11], and LLM with Multi-Agent Planning (LLM+MAP) [12] have achieved successful execution on various robot tasks by leveraging LLMs to decompose LH tasks into a sequence of PAs. In addition, EUREKA [13] and Video2Reward [14] reduced the need for human intervention in robot learning by using LLMs to autonomously generate reward functions that incorporate the PA subgoals of LH tasks. These approaches effectively overcome the limitations of traditional TAMP, as the pre-trained LLM planner eliminates the need for human intervention and avoids the convergence issues inherent in training a hierarchical policy from scratch. Despite these advancements, applying LLMs directly to bimanual robot tasks reveals significant shortcomings, stemming from a

[1]Ji-Heon Oh, Danbi Jung, Ismael Espinoza, Yong-Hyeok Choi, and Tae-Seong Kim are with the Department of Electronics and Information Convergence Engineering, Kyung Hee University, Yongin 17104, Republic of Korea. email: {dhwlgjs3, dbinqsu, inespinoza24, hope5090, tskim}@khu.ac.kr. Ji-Heon Oh and Danbi Jung are co-first authors.

*Corresponding author

[2]YoungOuk Kim, Dongin Shin, and JongSul Moon are with the Korea Electronics Technology Institute (KETI), 25 Saenari-ro, Yatap-dong, Bundang-gu, Seongnam-si, Gyeonggi-do, Republic of Korea. email: {kimyo, di_shin, moonjongsul}@keti.re.kr.

[3]Wonha Kim is with the Department of Electronics Engineering, Kyung Hee University, Yongin 17104, Republic of Korea email:wonha@khu.ac.kr.

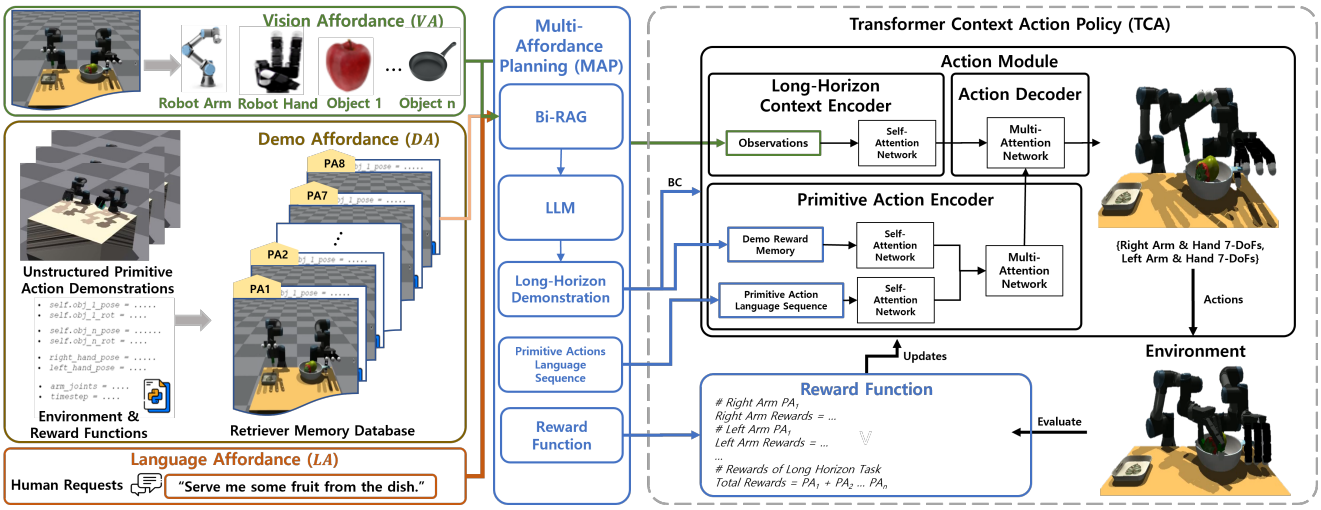


Fig. 1. Framework of MAP-TCA: Bi-RAG enhances LLM in MAP with task-specific contexts, guiding TCA through BC and fine-tuning to learn and perform bimanual LH tasks.

lack of physical grounding. This core issue manifests in two problems. First, LLMs are prone to hallucinations, generating plans that are physically or contextually infeasible because they lack the embodied domain-specific knowledge, leading the model to misunderstand the robot's bimanual kinematics or misjudge the success criteria for subtasks. Consequently, LLMs may produce incorrect task decomposition or assign impossible actions to the arms. Second, to compensate for this knowledge gap, adapting an LLM planner to new scenarios requires extensive PA-specific fine-tuning with expert demonstrations. This costly process not only hinders the generalization and scalability of LLM-based solutions but also reintroduces the very dependency on human data that LLMs are meant to overcome. Therefore, a novel framework is needed to harness the planning capabilities of LLMs for bimanual LH tasks without costly fine-tuning, while simultaneously mitigating the risk of hallucinations.

Concurrently, transformer-based agents have emerged as an alternative to conventional low-level action modules in LH manipulation to address execution failures, such as Temporal Context Transformer RL (TC-TRL) [15], Inter-Arm Coordinated Transformer Encoder (IACE) [16], Perceiver-actor2 (PerAct2) [17], and Action Chunk Transformer (ACT) [18]. By processing long sequences of visual observations, their attention mechanisms can comprehend the complex spatio-temporal context inherent in bimanual LH tasks, effectively overcoming the issue of missing temporal context where MLP-based policies fail. This capability has enabled transformer agents to achieve high success rates in some complex tasks learned directly from visual demonstrations [15]–[18]. Despite these advancements, these approaches introduce their own set of critical challenges. First, their success is predicated on a heavy dependence on LH human demonstrations, requiring massive datasets of expert trajectories to learn effectively. Second, due to a lack of logical guidance for the task's sub-goals, the agent is prone to physical hallucinations, which cause it to misinterpret ambiguous visual cues and fail to execute actions correctly. Furthermore, this lack of a structured plan leads to a severe credit assignment problem, making it nearly impossible to diagnose which part of a PA caused a failure. This forces the policy into a brute-force learning process that is highly sample-inefficient, hindering its application to a wide range of LH tasks. Therefore, a novel framework is needed to ground in multi-sensory information, using a high-level planner to prevent the physical hallucinations inherent to a vision-centric agent.

In this paper, we present Multi-modal Affordance Planner with Temporal Context Action Policy (MAP-TCA), a hierarchical framework designed to overcome the critical limitations of prior approaches in bimanual LH tasks. The MAP-TCA is inspired by human cognitive strategies of multi-sensory cross-validation and contextual memory integration to generate robust, feasible plans with minimal human supervision. Multi-sensory cross-validation could prevent hallucination related to situation awareness by validating different sensory modalities [19]. Context memory integration could also prevent planning hallucinations by ensuring that the proposed action sequence is coherent within the LH temporal context [20]. The framework comprises two modules that utilize these two cognitive strategies. The high-level Multi-modal Affordance Planner (MAP) employs a Llama3 [21] LLM model augmented with a Bimanual Robot Retrieval-Augmented Generation (Bi-RAG). The role of Bi-RAG is to retrieve and structure a PA-specific and multi-sensory latent space by associating PAs with their corresponding vision, language, and demonstration affordances. The MAP module leverages this multi-modal latent space to generate LH planner demonstrations, LH plans, and sparse reward functions. By leveraging multi-sensory information from Bi-RAG, the MAP module is grounded in the current environment, enabling it to generate physically feasible action plans and thereby mitigating the risk of hallucinations. Also, the plan-

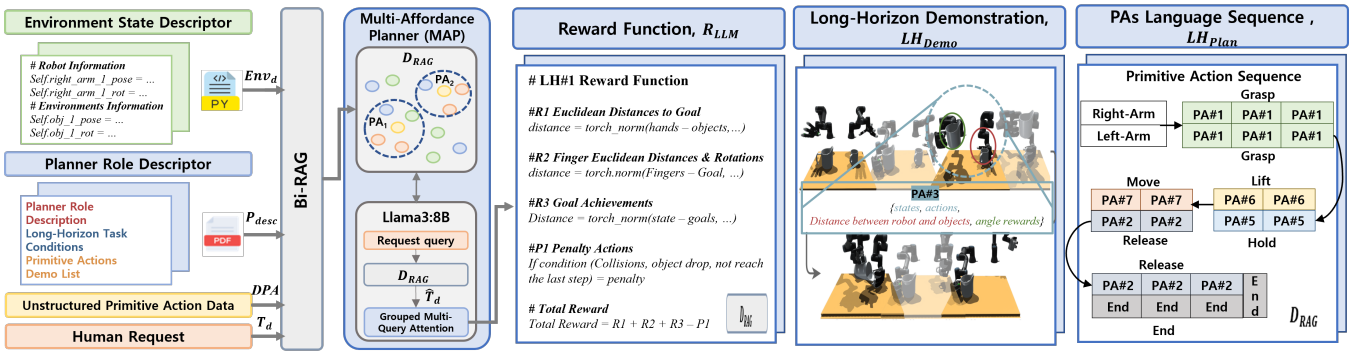


Fig. 2. Structure of the Bi-RAG module and MAP. The MAP generates R_{LLM}, LH_{demo}, and LH_{plan} based on environment state descriptions, human requests, planner role descriptor, and unstructured primitive action database.

ner's outputs eliminate the need for costly LH human demonstrations and hand-crafted reward functions, thus directly addressing the data dependency problem. The low-level module, Temporal Context Action policy (TCA), employs an attention-based transformer architecture to generate actions for the LH tasks, guided by the LH planner demonstrations and reward functions produced by the MAP module within an offline-online RL framework. The TCA is composed of an encoder-decoder transformer architecture designed to understand both LH and PA contexts. The LH context encoder leverages vision affordance to maintain situational awareness of the overall task. The PA context encoder in TCA evaluates the current PA's progress by comparing it against patterns derived from the LH planner demonstration and the vision affordance. Finally, the decoder integrates multi-affordance context using a cross-attention mechanism to generate final, context-aware actions for the bimanual robot. The proposed MAP-TCA demonstrated an average success rate of 86.75%. Critically, this performance is achieved solely using MAP, with no dependence on direct human LH demonstrations. By synergistically combining LLM-based affordance reasoning with transformer-based temporal validation, our MAP-TCA provides a principled and efficient.

II. Methods

Fig. 1 shows our proposed MAP-TCA framework, which consists of three main stages. First, the **Bimanual robot Retrieval-Augmented Generation (Bi-RAG)** retrieves contextual affordances from multi-modal information to generate PA-specific textual prompts that ground the subsequent planning phases. The **Multi-modal Affordance Planner (MAP)** leverages PA-specific prompts to generate an LH planner demonstration LH_{demo}, a language plan LH_{plan}, and a sparse reward function $RLLM$. Finally, a Temporal Context Action Policy (TCA) generates action sequences for a bimanual robot based on LH_{demo}, LH_{plan}, and R_{LLM} via offline-online RL training.

A. Bimanual Robot Retrieval-Augmented Generation

The Bi-RAG utilizes language affordances LA, vision affordances VA, and PA demonstration affordances DPA, to generate a PA-specific retrieval database, D_{RAG} and augmented text descriptions $\hat{T}_d$. The LA contains the LH task intention requested by a person and consists of a human request, T_d, and a planner role descriptor P_{desc}. The P_{desc} guides the feasible output for MAP by defining the maximum LH task duration, t_{max}, and the language description of ten elementary PAs described in Table I. The $VA = \{o_t^{robot}, o_t^{objects} \dots\}$ is the current physical state of the robot environment, including 44-DoFs bimanual robot data and 6-DoFs object affordance, which contains an environment code descriptor Env_d. The DPA provides the state-action-reward sequences and the sparse reward function for each PA.

The Bi-RAG utilizes a dense retrieval mechanism with a pre-trained text embedding model, Sentence-BERT [22], to embed multi-affordance inputs into D_{RAG}. The D_{RAG} stores task-specific information to each PA as a vector, $v(i,q) = \{v_1, v_2, \dots, v_n\}$ [23] where $v(i,q)$ is composed by an index, i and query, q supporting efficient semantic retrieval across n retrieval steps for each PA. The Bi-RAG encodes T_d into a query vector, C_in. It then retrieves the most semantically relevant context, C_{retr}, from D_{RAG}. The C_{retr} is accomplished by identifying the stored $v(i,q)$ that maximizes cosine similarity with C_in, as formalized in **(1)**:

$$C_{retr} = argmax(cos_{sim}(v(i,q), C_{in})) \quad (1)$$

Finally, the Bi-RAG module concatenates T_d and C_{retr} to generate $\hat{T}_d$ which contains the PA-specific information.

B. Multi-modal Affordance Planner

The MAP generates R_{LLM},LH_{demo}, and LH_{plan} using a Llama3-8B model [21] based on $\hat{T}_d$. As shown in Fig. 2, LH_{demo} consists of state-action sequence pairs, $(s_t, \hat{a}_t)$ of PAs and reward-related information, including the Euclidean distance reward function, $R_{dist}(p_i, p_g)$ between the robot hand's position, p_i and the optimal target position, p_g during performing LH_{demo}. The LH_{plan}, depicted in Fig. 2, is a language representation comprising a sequence of PAs assigned to each arm of a bimanual robot. The MAP also generates R_{LLM} on $\hat{T}_d$ within the context of the current LH task and

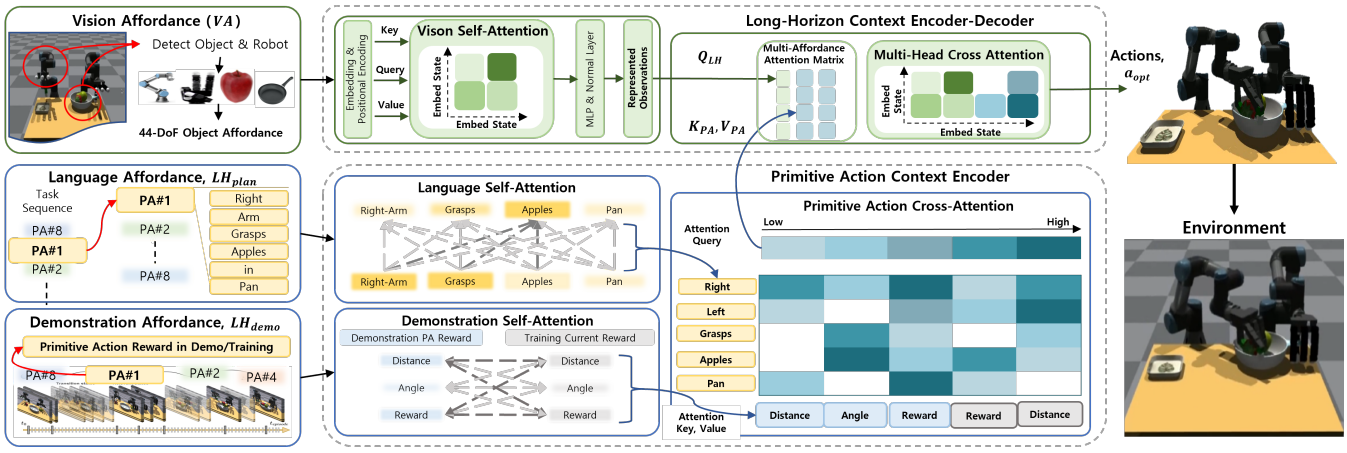


Fig. 3. Detailed framework of Temporal Context Action (TCA) policy.

D_{RAG}. The $R_{LLM}(o_t)$ defined in **(2)** is capable of guiding TCA based on the current observation, o_t, of the LH task.

$$R_{\text{LLM}}(o_t) = \sum_{i=1}^{n} \left[R_{\text{dist}}(p_i, p_g)^i + R_{\text{align}}(R_{\text{obj}}, R_g)^i - P^i_{\text{pen}} \right] \quad (2)$$

where $R_{align}(R_{obj}, R_g)^i$ is the rotational alignment reward function, which measures the orientation of the manipulated object, R_{obj}, to achieve a goal orientation, R_g. The P^i_{pen} represents penalties for undesirable and inappropriate actions, such as collisions or kinematic limitations. To ensure the successful execution of the overall LH task, the MAP module defines the precise success conditions for each PA, such as p_g, R_g, and P_{pen} by constraints in P_{desc} and Env_d. By allowing the planner to adjust parameters based on the multi-sensory affordances of the reward function, our approach generates consistent, stable reward functions.

C. Temporal-Context Action Policy

The proposed TCA handles bimanual robot actions for various LH tasks based on LH_{plan}, LH_{demo}, VA, and R_{LLM}. As depicted in Fig. 3, the TCA comprises three primary modules: first, an LH context encoder embeds VA from LH tasks, thereby learning relevant temporal context. Second, a primitive-context encoder processes LH_{plan} and LH_{demo} to learn the relevant PA progress. Third, an action decoder generates an optimal action, a_{opt}, based on the LH and primitive contexts.

1) Long-Horizon Context Encoder: The LH context encoder processes VA to generate the LH context embedding matrix, Q_{LH}. The LH context encoder begins by embedding VA into a sequence of tokens using a linear transformation layer. To preserve the temporal ordering and spatial relationships inherent in the LH task, an absolute positional encoding function [24] is then applied to inject positional information into each token. From this, a sequence of position-aware tokens is generated along with the key, K_LH, query Q_{LH}, and value V_{LH} matrices. These matrices are then processed by a multi-head self-attention mechanism to learn temporal dependencies within the task sequence. Finally, the encoder uses the general dot-product attention as a scoring function to represent the correlation between the token observations as in **(3)**:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (3)$$

where d_k represents the length of the observation vector. This attention mechanism embeds the represented observations, $\hat{o}^i_t$ and Q_{LH}, by identifying the LH temporal context between the robot and the object in VA. The $\hat{o}^i_t$ is embedded by assigning the closest attention to o_t during LH tasks. The LH context encoder uses a last-layer critic network to optimize the local state-value function $V_\theta(\hat{o}^i_t, t)$. This critic network captures the value of $\hat{o}^i_t$ n the LH context, ensuring a stable training.

2) Primitive Context Encoder: The primitive context encoder processes LH_{plan} and LH_{demo}. The LH_{plan} and LH_{demo} are fed into a self-attention module to generate the PA contextual embeddings, K_{PA} and V_{PA}, respectively. The LH_{plan} encoder embeds PA descriptions by capturing the internal relationships among instructional words. Similarly, the LH_{demo} self-attention layer embeds the relationship between $R_{LLM}(o_t)$, and $R_{LLM}(s_t)$ in the PA context. The subsequent cross-attention layer generates K_{PA} and V_{PA} pairs, contextualized for PAs by fusing LA and DPA. This mechanism effectively aligns the language-based instructions for PAs with the relevant spatial and reward status in the PA context derived from LH_{plan} and LH_{demo}.

3) Action Decoder: The action decoder generates a_{opt} by synergistically integrating Q_{LH}, K_{PA}, and V_{PA} through a multi-affordance cross-attention mechanism. This design allows the action decoder to ground its predictions in the proven expert trajectories while retaining the flexibility to dynamically select the most relevant skills based on the LH and PA context. Consequently, the decoder produces a_{opt} that are not only physically feasible and consistent with LH_{plan} but also highly adaptive to the current situation. The resulting output from this cross-attention module is then passed through a final layer to produce the policy distribution, $\pi_\phi(a_{opt}|o_t)$ over the agent's action space.

4) Temporal Context Action Policy Training: The TCA is trained via a two-phase offline-online methodology. During offline training, the policy is initialized using Behavior Cloning (BC) [25] on LH_{demo}, thereby establishing a robust foundation that imitates feasible expert plans. During the online training, TCA is fine-tuned using the Demo Augmented Proximal Policy Optimization (DA-PPO) algorithm [15] to maximize $R_{LLM}(o_t)$. The training objective minimizes a modified clipped PPO loss $L(\theta,\phi)$. The encoder is trained by minimizing $L_E(\theta)$ as shown in **(4)**, where ϕ represents the encoder parameters T, the total time steps, γ, and N, the discount factor and parallel environments, respectively.

$$L_E(\theta) = \frac{1}{T}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\left[R_{\text{LLM}} + \gamma V_\theta(\hat{o}^i_{t+1},t) - V_\theta(\hat{o}^i_t,t)\right]^2 \quad (4)$$

The decoder is trained by minimizing $L_D(\phi)$, which consists of the clipped PPO loss and demonstration loss in [15], as shown in **Eq.(5)**.

$$L_D(\phi) = L(\theta,\phi) - \frac{1}{T} + \lambda \sum_{(s,a)\in\rho_D}\left[\pi_\phi(a \mid s) - \pi^{LH}_\phi(\hat{a} \mid s)\right]^2 \quad (5)$$

where λ represents the discount factor $\pi_\phi(a \mid s)$ the current decoder policy and $\pi^{LH}_\phi(\hat{a} \mid s)$ the demo augmented policy. ρ_D is the set of LH planner demonstrations. Finally, the total loss for MAP-TCA is $L = L_E(\theta) + L_D(\phi)$.

III. Experimental Results

A. Experimental Setup

A set of four LH manipulation tasks summarized in Table I was performed to evaluate MAP-TCA. Each of the LH tasks was composed of at least eight PAs selected from Table I. For example, in the case of executing LH#3, the left arm performs a sequence of PAs comprising grasp-lift-move-hold-push, while the right arm follows a sequence of grasp-lift-tilt-hold. In addition, LH#1 and LH#2 tasks were tested using unseen objects shown in Table I to validate the generalization for various objects. The objects used in LH#1 are fruit, while those in LH#2 are common household items.

Our experimental platform, used in both simulation and real hardware settings, consists of a bimanual robot with two UR3 arms and Allegro hands. The bimanual robot has 44 degrees of freedom (DoFs), where 6 DoFs correspond to each robot arm and 16 DoFs to each dexterous hand. In simulation, the training environments were set up under the NVIDIA Isaac Gym Physics engine [26]. The DPA and LH human demonstrations were collected with a custom teleoperation module. Training was conducted on 500 parallel environments across 16 threads by leveraging a single RTX 3070 GPU with 32GB of RAM. We employed two TRL baselines to compare MAP-TCA's performance. The first baseline, Demo-Augmented Multi-Agent Transformer (DA-MAT) [15], is an extension of the Multi-Agent Transformer [27] that integrates imitation learning and a demo-gradient

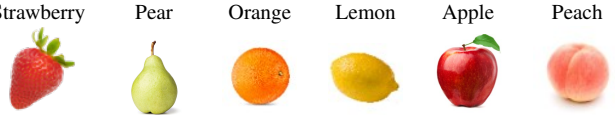
TABLE I
DESCRIPTION OF PAs & LH TASKS

PA#1	Grasp Object	**PA#5**	Hold Object
PA#2	Release Object	**PA#6**	Lift Object
PA#3	Pull Object	**PA#7**	Move Object
PA#4	Push Object	**PA#8**	Tilt Object

LH#1 "Relocate two objects in the bowl."
LH#2 "Pick up two objects, relocate them into the drawer, and then push the drawer door."
LH#3 "Pick up a kettle and a cup, pour water into the cup, and push it to the person."
LH#4 "Relocate two objects in the tray, and push serve."

LH#1 objects list

Strawberry Pear Orange Lemon Apple Peach



LH#2 objects list

Soap Cleanser Bottle Box SPAM Brick Toy

loss using LH human demonstrations without language information. The second model, Temporal Context Action policy (TCA), utilizes complete LH human demonstrations of each LH task and human-crafted reward functions. To evaluate all models, the average success rate of PAs is used as the evaluation metric. To assess the feasibility of our model on the hardware robot, we used LH#4, which includes PAs from both LH#1 and LH#2. We employed a real-to-sim-to-real methodology, which involves training the agent in simulation and subsequently fine-tuning in real physical settings [28]. First, we created a digital twin of the bimanual robot using the URDF model to accurately mirror its kinematics and inertial parameters. After training the policy for LH#4, it was transferred to the physical robot, which is controlled via a ROS2 Humble bridge. The hardware observation vector was constructed by obtaining joint angles and hand poses from the ROS2 drivers, while the object poses were estimated by FoundationPose [29]. To bridge the sim-to-real gap, we fine-tuned the policy with domain randomization, varying physical parameters such as friction, mass, and actuation delays to enhance the policy's robustness.

B. Performance Evaluation Results

The performance of MAP-TCA and the two baseline models was evaluated across the four LH tasks (i.e., LH#1≈#4), with success rates reported in Table II. The proposed MAP-TCA achieved an average success rate of 86.75% across the LH tasks, comparable to TCA trained on human demonstrations, which achieved 87.75%. The significance of this result is that TCA relies on extensive human demonstrations and handcrafted reward functions for each LH task. In contrast, MAP-TCA achieved comparable performance using only a compact set of PAs demos from DPA to generate

TABLE II

QUANTITATIVE COMPARISON RESULTS OF LH TASKS. THE PA SUCCESS RATES OF THE LH TASK SEQUENCES ARE REPORTED: SUCCESS RATES OF THE RIGHT ARM (BLUE), LEFT ARM (GREEN), AND BOTH ARMS (ORANGE).

Methods	LH#1				LH#2				LH#3			
	Grasp fruit	Lift fruit	Move fruit	Release fruit	Grasp objects	Lift objects	Release objects	Push drawer	Grasp objects	Lift objects	Tilt kettle	Hold cup
DA-MAT	21/25	19/25	17/25	14/25	18/25	16/25	13/25	10/25	17/25	16/25	12/25	15/25
TCA	25/25	24/25	21/25	20/25	23/25	24/25	22/25	19/25	24/25	23/25	20/25	20/25
MAP-TCA	25/25	24/25	20/25	19/25	24/25	22/25	23/25	18/25	24/25	23/25	21/25	20/25
Methods	LH#4				LH#1 with Unseen Objects				LH#2 with Unseen Objects			
	Grasping fruit	Lifting fruit	Grasp plate	Move fruit	Grasp fruit	Lift fruit	Move fruit	Release fruit	Grasp objects	Lift objects	Release objects	Push drawer
DA-MAT	20/25	10/25	23/25	6/25	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail
TCA	24/25	23/25	24/25	15/25	Fail	Fail	Fail	Fail	Fail	Fail	Fail	Fail
MAP-TCA	24/25	22/25	24/25	14/25	22/25	17/25	13/25	12/25	20/25	19/25	14/25	14/25

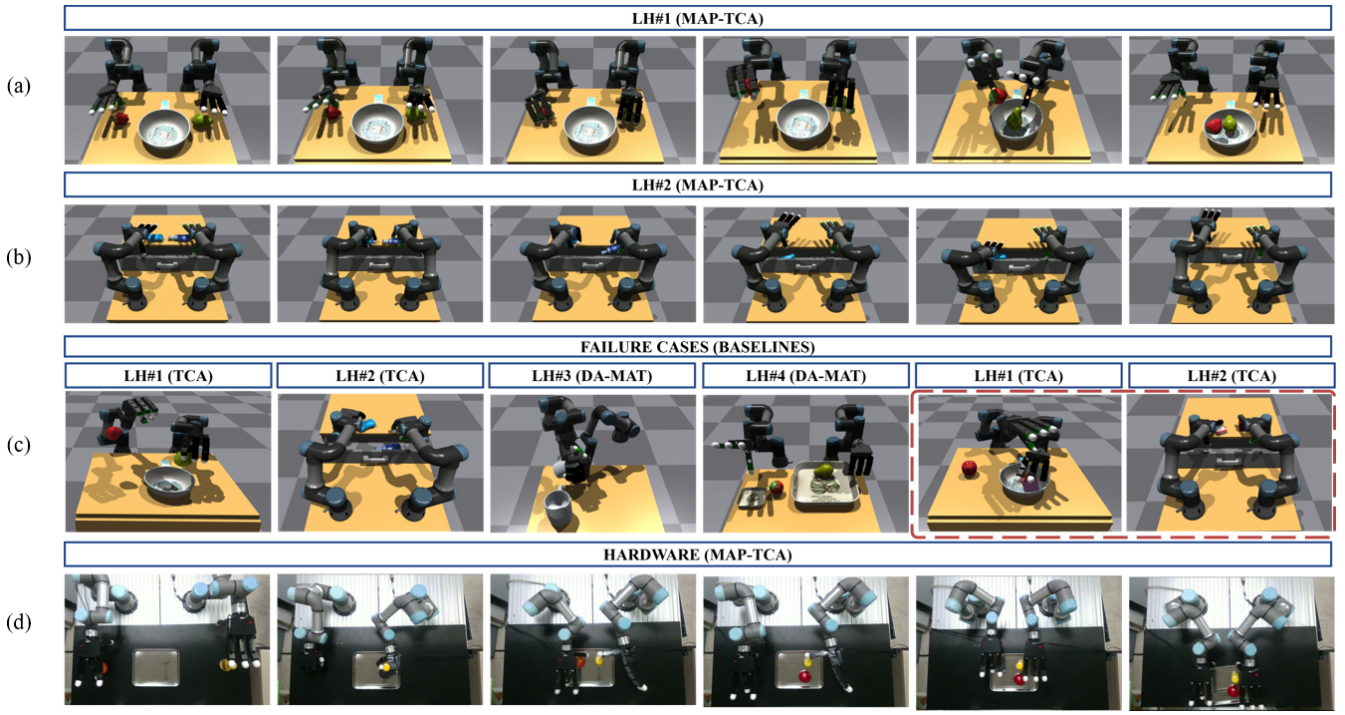


Fig. 4. (a) and (b) time-series frames of performing LH#1 and #2, respectively, by MAP-TCA, (c) failure case by TCA and DA-MAT for LH#1≈#4, and (d) performing LH#4 by MAP-TCA with the hardware robot.

LH demonstrations and rewards, drastically reducing human effort. The DA-MAT achieved a success rate below 60%, struggling to meet PA goals and maintain stable grasps.

To evaluate the generalized performance of MAP-TCA, the models were further tested on LH#1 and #2, which involve previously unseen objects, as shown in Table II. In these scenarios, the MAP-TCA achieved an average success rate of 65.5%. In contrast, both DA-MAT and TCA baselines failed to perform the LH tasks. Due to their non-adaptive reward functions, neither TCA nor DA-MAT could handle object variations, leading to grasping failures. These results demonstrate that the multi-affordance-based LLM planner can generate appropriate reward functions for novel environments and unseen objects, possibly replacing the need for handcrafted reward functions. Fig. 4 (a) and (b) show the time-series frames of executing the LH tasks by MAP-TCA, and (c) the failure cases by DA-MAT and TCA. The failure cases by DA-MAT highlight the issue of physical hallucination. The DA-MAT received a maximum reward, yet the tasks were not successfully completed in LH#3 and #4. The vision-centric DA-MAT policy misinterpreted the static visual cues, leading to failure in the final stages of the LH tasks. Also, the TCA fails to grasp unseen objects in LH#1 and #2. In contrast, the MAP-TCA completed the LH tasks

correctly. These results demonstrate that a multi-affordance approach and reward function from MAP is more robust and generalizable, and less prone to physical hallucinations. Fig. 4 (d) shows the hardware robot's performance of LH#4 with the migrated MAP-TCA policy. The LH#4 task requires a synchronous and asynchronous coordination of both arms, such as "Grasping and relocating the objects" from LH#1 and "Pushing a door or tray" from LH#2. The results demonstrate the feasibility of transferring the learned policy to the hardware robot system.

IV. Conclusion

In this paper, we present MAP-TCA, a novel hierarchical framework that integrates MAP, enhanced by Bi-RAG as a planner, and TCA as an actor, to overcome the challenges of demonstration dependency, planning hallucination, and robust execution in bimanual LH tasks. Our core contribution is the Bi-RAG-enhanced MAP module, which generates structured LH_{demo}, LH_{plan} and R_{LLM} from multi-modal affordances. This process effectively grounds the LLM planner, eliminating both the risk of hallucination and the need for hand-crafted rewards or human LH demonstrations. The LH_{demo},LH_{plan} and R_{LLM} enable TCA to generate bimanual robot actions for LH tasks robustly through a combination of offline and online training, allowing it to understand the context of PAs and LH tasks. Our experimental results demonstrate the effectiveness of MAP-TCA, achieving a success rate of 86.75% across the four LH tasks. The results reveal comparable performance to TCA's 87.75%, achieved under full human supervision. Furthermore, the proposed MAP-TCA demonstrates strong generalization, outperforming the two baseline models by 55.4% in success rate on tasks involving unseen objects. The presented MAP-TCA could provide a scalable, automated solution for complex bimanual manipulation tasks on humanoid robots.

ACKNOWLEDGMENT

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. RS-2024-00509257, Global AI Frontier Lab), the Information Technology Research Center (ITRC) grant funded by the Korean government (Ministry of Science and ICT) (IITP-2025 RS-2024-00438239), the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program (RS-2023-00232141), Development of a life assist mobile robot that understands people's daily behavior, and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (NRF-2023R1A2C100358511).

References

[1] Y. Xie, X. Zhao, Y. Jiang, Y. Wu, and H. Yu, "Flexible control and trajectory planning of medical two-arm surgical robot," vol. 18, publisher: Frontiers.

[2] H. Jeong, H. Lee, C. Kim, and S. Shin, "A survey of robot intelligence with large language models," vol. 14, no. 19, p. 8868, publisher: Multidisciplinary Digital Publishing Institute.

[3] S. Kataoka, Y. Chung, S. K. S. Ghasemipour, P. Sanketi, S. S. Gu, and I. Mordatch, "Bi-manual block assembly via sim-to-real reinforcement learning."

[4] D. Yu, H. Xu, Y. Chen, Y. Ren, and J. Pan, "BiKC: Keypose-conditioned consistency policy for bimanual robotic manipulation."

[5] F. Xie, A. Chowdhury, M. C. De Paolis Kaluza, L. Zhao, L. Wong, and R. Yu, "Deep imitation learning for bimanual robotic manipulation," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., pp. 2327–2337.

[6] T. Ren, G. Chalvatzaki, and J. Peters, "Extended tree search for robot task and motion planning," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12 048–12 055, ISSN: 2153-0866.

[7] S. Cheng and D. Xu, "LEAGUE: Guided skill learning and abstraction for long-horizon manipulation," vol. 8, no. 10, pp. 6451–6458.

[8] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman, "Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning."

[9] K. Chu, X. Zhao, C. Weber, M. Li, W. Lu, and S. Wermter, "Large language models for orchestrating bimanual robots," in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pp. 328–334, ISSN: 2164-0580.

[10] M. Dalal, T. Chiruvolu, D. Chaplot, and R. Salakhutdinov, "Plan-seq-learn: Language model guided RL for solving long horizon robotics tasks."

[11] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "TidyBot: personalized robot assistance with large language models," vol. 47, no. 8, pp. 1087–1102.

[12] K. Chu, X. Zhao, C. Weber, and S. Wermter, "LLM+MAP: Bimanual robot task planning using large language models and planning domain definition language."

[13] Y. J. Ma, W. Liang, G. Wang, D.-A. Huang, O. Bastani, D. Jayaraman, Y. Zhu, L. Fan, and A. Anandkumar, "Eureka: Human-level reward design via coding large language models."

[14] R. Zeng, D. Zhou, Q. Liang, J. Liu, H. Li, C. Huang, J. Li, X. Hu, and F. Sun, "Video2reward: Generating reward function from videos for legged robot behavior learning."

[15] J.-H. Oh, I. Espinoza, D. Jung, and T.-S. Kim, "Bimanual long-horizon manipulation via temporal-context transformer RL," vol. 9, no. 12, pp. 10 898–10 905.

[16] A. Lee, I. Chuang, L.-Y. Chen, and I. Soltani, "InterACT: Inter-dependency aware action chunking with hierarchical attention transformers for bimanual manipulation."

[17] M. Grotz, M. Shridhar, Y.-W. Chao, T. Asfour, and D. Fox, "PerAct2: Benchmarking and learning for robotic bimanual manipulation tasks."

[18] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware."

[19] B. E. Stein and T. R. Stanford, "Multisensory integration: current issues from the perspective of the single neuron," vol. 9, no. 4, pp. 255–266, publisher: Nature Publishing Group.

[20] C. Ranganath and M. Ritchey, "Two cortical systems for memory-guided behaviour," vol. 13, no. 10, pp. 713–726, publisher: Nature Publishing Group.

[21] A. Grattafiori *et al.*, "The llama 3 herd of models."

[22] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Association for Computational Linguistics, pp. 3982–3992.

[23] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "THE FAISS LIBRARY," pp. 1–17.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc.

[25] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations."

[26] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance GPU-based physics simulation for robot learning."

[27] M. Wen, J. Kuba, R. Lin, W. Zhang, Y. Wen, J. Wang, and Y. Yang, "Multi-agent reinforcement learning is a sequence modeling problem," vol. 35, pp. 16 509–16 521.
[28] M. Torne, A. Simeonov, Z. Li, A. Chan, T. Chen, A. Gupta, and P. Agrawal, "Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation."
[29] B. Wen, W. Yang, J. Kautz, and S. Birchfield, "FoundationPose: Unified 6d pose estimation and tracking of novel objects," pp. 17 868–17 879.