DGV: Fusing Dynamic Graphs and Vision-Language Models for Collaborative Dual-Arm Task Planning

Yapeng Pang[1], **Junjie Xu**[1], **Zhidong Qiao**[2], **Peng Du**[3], **Xinyu Zhang**[1]

[1] East China Normal University
[2] Harbin Institute of Technology
[3] Zhejiang University

Abstract

Dual-arm collaborative manipulation in dynamic, unstructured environments is profoundly challenging, requiring real-time handling of high-dimensional physical constraints alongside dynamic scene understanding and adaptation to high-level natural language instructions. To address these challenges, we propose the Dynamic Graph Vision-Language Model (DGV), a novel dynamic task planning framework that seamlessly integrates GNNs and VLMs. It first leverages a pre-trained VLM to integrate perceptual and semantic processing, accurately extracting object states and complex manipulation intents from the environment. This extracted information is then encoded into a dynamic spatio-temporal graph that models the robot's kinematic structure, environmental object relations, and temporal dependencies within a single, unified representation. We propose a real-time local subgraph update mechanism, which is designed to cope with rapid environmental changes. This mechanism ensures immediate action adjustments and efficient replanning based on fresh visual feedback, dramatically improving dynamic adaptability. Utilizing the updated graph structure, DGV performs efficient reasoning to generate continuous, stable, and robust dual-arm collaborative motion sequences. Our experimental results across both simulation and real-world robot platforms demonstrate that DGV achieves a task success rate nearly 20% higher than current state-of-the-art methods, while exhibiting superior performance in dynamic adaptability and robustness.

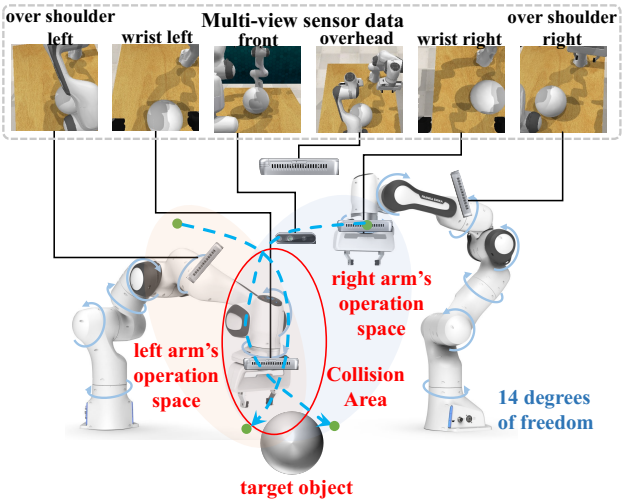

Figure 1: Two challenges when using the 14-DOF Franka Emika robots for collaborative tasks are: (1) High-DOF coupled planning, where intricate non-linear kinematics necessitate collision avoidance, particularly within the high-risk space indicated by a red circle (inter-arm and arm-environment collisions). (2) Uncertain multi-view perception, where multi-view camera data is compromised by real-world factors like occlusion and semantic ambiguity, severely degrading the robustness of dynamic task planning.

Introduction

Robust and efficient task planning is key to achieving advanced dual-arm manipulation, but it faces two primary challenges in dynamic and unstructured environments, as illustrated in Figure 1. First, the high-dimensional nature and complex physical constraints of dual-arm systems induce non-linear coupling and combinatorial complexity. This causes an explosion of the solution space, severely limiting the efficiency and convergence of traditional planning algorithms (Ratliff et al. 2009; Berenson, Srinivasa, and Kuffner 2011; Cohen, Chitta, and Likhachev 2012; Zacharias, Borst, and Hirzinger 2006; Kajita et al. 2003). The second challenge arises from the interplay of dynamic environmental changes, sensor perception errors, and high-level task semantic uncertainty in practical scenarios. These combined factors collectively prevent conventional planning methods from meeting critical demands for real-time performance, robustness, and effective high-level semantic understanding (Huang et al. 2023; Driess et al. 2023; Qureshi et al. 2020; Kaelbling and Lozano-Pérez 2013; Biswas and Veloso 2012; Pronobis and Rao 2017).

Dual-arm task planning research primarily focuses on high-dimensional motion spaces and non-linear constraints. Optimization methods like CHOMP (Ratliff et al. 2009) and TrajOpt (Schulman et al. 2014) generate smooth trajectories via gradient/convex optimization, while Kinematics-based Model Predictive Control (MPC) (Jiang et al. 2016; Dehio, Wang, and Kheddar 2022) excels in dynamic environments.

Concurrently, deep learning approaches, such as Diffusion Policy (Chi et al. 2025) and ACT (Zhao et al. 2023) (using imitation learning), generate robust actions and achieve manipulation. Furthermore, Reinforcement Learning (RL) based algorithms like CQL (Kumar et al. 2020) explore policy spaces. Despite these advancements, these methods generally lack a structured representation of complex collaborative relationships. Their reliance on prior kinematic models severely limits adaptability to dynamic perturbations and unstructured changes.

To overcome challenges related to task object changes and perceptual uncertainty, recent research has integrated advanced perception and high-level semantic capabilities into planning. Early methods, like Visual Foresight (Ebert et al. 2018) and MPNet (Qureshi et al. 2020), used video prediction and image perception to improve policy robustness against geometric perturbations. The advent of Vision-Language Models (VLMs) marked a major breakthrough: Palm-E (Driess et al. 2023) enabled zero-shot semantic understanding, VoxPoser (Huang et al. 2023) mapped semantics to 3D value maps, and VIMA (Jiang et al. 2022) enhanced multi-scene generalization through multimodal prompting. While these VLM-based methods excel in semantic understanding, they fundamentally lack structured modeling of complex physical relationships and dynamic environmental changes. Crucially, the lack of online feedback and error correction prevents these methods from simultaneously satisfying the dynamic consistency and real-time adaptability required for dual-arm collaboration.

While VLMs provide semantic grounding, they lack structured modeling of physical dependencies. Therefore, a structured paradigm, such as Graph Neural Networks (GNNs), is critically necessary to model complex collaborative constraints and achieve dynamic adaptability. Conversely, GNNs model structure lacks semantic understanding. To address the challenges of structural coupling and dynamic semantic perception, we propose the Dynamic Graph Vision-Language Model (DGV), a dynamic task planning framework that fuses GNNs and VLMs based on a Behavioral Cloning (BC) paradigm. This method utilizes a unified dynamic spatio-temporal graph as its core representation. This graph systematically encodes high-dimensional geometric constraints, the physical coupling of dual-arm collaboration, and high-level task semantic information into its nodes and edges, thereby achieving structured, explainable, and adjustable collaborative planning. Our contributions are summarized as follows.

- **Semantic-Fusion for Dual-Arm Collaboration.** We use a VLM to jointly parse natural language instructions and multi-view perception results, recognizing target objects, task intentions, and environmental constraints to provide a high-level prior for graph construction.
- **Semantically Augmented State Graph.** We construct a unified spatio-temporal graph that combines the robot's structure and sensory information, expressing joint states, physical connections, and task logic to support task planning under dual-arm collaboration.
- **Online Local Subgraph Update for Dynamic Perturbations.** We design an online graph adjustment mechanism based on VLM feedback, which dynamically updates perturbed nodes and their neighbors using attention mechanisms, thereby improving action sequence reconstruction capability and execution robustness.
- **Dynamic Graph-Guided Action Sequence Inference.** We leverage Spatial GCN and Temporal Attention to encode structural constraints, followed by a Gaussian Mixture Model (GMM) decoder to achieve continuous, feasible, and highly collaborative action sequence generation.

Related Work

In this section, we briefly review the related work on semantic fusion, spatio-temporal graph modeling, adaptive graph updates, and action sequence inference.

Semantic Fusion for Dual-Arm Collaboration

VLMs enhance task and environmental semantic understanding via joint image-language modeling. While models like CLIP (Radford et al. 2021), BLIP (Li et al. 2022), and LLaVA (Liu et al. 2023) excel in zero-shot cross-modal alignment, they focus on end-to-end decision-making, lacking the structured modeling required for dual-arm collaborative planning. SayCan (Ahn et al. 2022) maps language to behavior via a scoring mechanism, and RT-2 (Zitkovich et al. 2023) introduced the multimodal policy transduction framework. As these methods generally lack the structured modeling capability needed for complex collaborative tasks, we address this by fusing VLM-extracted semantic priors with GNNs for robust action sequence optimization.

Spatio-Temporal Graph Modeling for Dual-Arm

Traditional planning often fails to model structural dependencies in multi-joint systems. GNNs overcome this by encoding joint topology (Yan, Xiong, and Lin 2018), cross-view perception (Zhou et al. 2022), distributed exploration (Zhang et al. 2022), and robot-object interaction (Lin et al. 2022) to support effective action generation. Building upon this, we integrate structural perception with VLM semantic information to construct a unified, semantically augmented collaborative state spatio-temporal graph, which provides the structured, semantic-driven modeling foundation for dynamic dual-arm coordinated action sequence optimization.

Adaptive Graph Update for Dynamic Disturbance

Dynamic environmental disturbances challenge real-time task planning, rendering static graphs inadequate. While Dynamic GNNs like EvolveGCN (Pareja et al. 2020), DyRep (Trivedi et al. 2019), and InstantGNN (Zheng et al. 2022) support real-time node and edge updates for dynamic adaptation, we introduce a VLM-driven local subgraph adjustment mechanism. By using attention to reconstruct the adjacency of perturbed regions, our method boosts online action sequence reconstruction and execution robustness.

Action Sequence Inference via Dynamic Graphs

Dual-arm planning requires continuous, constrained action sequences. Unlike traditional optimization or end-to-end

learning, GNNs offer superior adaptability and interpretability in high-dimensional planning (Khan et al. 2020; Zang et al. 2023). Inspired by these successful applications, we utilize the real-time updated collaborative state graph and an attention mechanism to infer dual-arm action increment sequences. This mechanism simultaneously satisfies VLM semantic understanding, structural constraints, and physical feasibility, ensuring action smoothness and robustness.

Problem Definition

The dynamic task planning problem for dual-arm collaborative manipulation is defined as the real-time generation of high-dimensional, continuous dual-arm action sequences that satisfy multiple constraints within a dynamic and unstructured environment, based on multimodal inputs. Specifically, at any time t, the system receives a multimodal input $\mathcal{I}_t$, which comprises three essential components: the high-level natural language instruction l, which defines the task's semantic objective; the multi-view RGB-D observation set O_t (provided by six RGB-D cameras, as illustrated in Figure 1) for real-time environment perception; and the dual-arm system's current joint state s_t. The joint state is formally defined as $s_t = (q_0, q_1)$, where $q_i \in \mathbb{R}^n$ denotes the configuration vector of the i-th arm, and n is the joint degrees of freedom (DOF) of a single manipulator. This process formulates task planning as a constrained optimization problem, where the system must decompose high-level semantic intentions into sequential, physically feasible motion primitives while maintaining dynamic coordination. The overall objective is to generate a sequence of continuous actions $\mathcal{A} = \{a_t\}_{t=0}^{T}$ that drives the dual arms to collaboratively execute the task, subject to constraints on task semantics, physical reachability, and collision avoidance (safety).

Methodology

The overall framework of our proposed method, illustrated in Figure 2, is composed of four mutually collaborative core modules: Dual-Arm collaborative Semantic Fusion Modeling, Semantic collaborative Graph Modeling, Dynamic Local Subgraph Update Module, and Dynamic Graph Action Generation Module. These four modules synergistically collaborate to constitute an intelligent task planning system that is endowed with both adaptability and semantic closed-loop capability in dynamic environments.

Dual-Arm collaborative Semantic Fusion Modeling

The validity and robustness of dual-arm collaborative manipulation are critically dependent on accurately understanding high-level task semantics and environmental constraints. To address this, this module designs a VLM-based semantic fusion mechanism that uniformly parses natural language instructions and multi-view perception inputs to extract high-level semantic priors, which subsequently guide graph construction and dynamic action sequence inference.

Multimodal Input Representation. This module primarily relies on two input streams:

- Natural language task instruction $l \in \mathcal{L}$, serves as the high-level task description (e.g., "use the right arm to lift the red bowl"), explicitly defining the manipulation objective and semantic constraints.
- Multi-view observation set $\mathcal{O} = \{(I_i, D_i)\}_{i=1}^{6}$, provided by six RGB-D cameras. The RGB image $I_i \in \mathbb{R}^{H\times W\times 3}$ captures the scene appearance, while the depth map $D_i \in \mathbb{R}^{H\times W}$ provides accurate 3D geometric information.

High-Level Semantic Prior Extraction via VLM. We employ a fine-tuned PaliGemma (Beyer et al. 2024) model for joint modeling of $\mathcal{O}$ and l. This VLM, composed of a SigLIP-So400m image encoder and a Gemma-2B text decoder, possesses strong cross-modal alignment capability. PaliGemma is responsible for object detection and semantic recognition, 3D pose estimation, affordance inference, and semantic relevance assessment.

High-Level Semantic Prior Output. The VLM ultimately outputs the set of environmental objects $\mathcal{V} = \{\mathcal{V}^1 \cup \mathcal{V}^2$, representing of K obstacles and J target objects. For each object $v \in \mathcal{V}$, we extract its semantic prior features $\boldsymbol{F}_v$, encompassing: Geometric Information: the object's center pose $\boldsymbol{p}_v \in \mathbb{R}^7$ (position and quaternion orientation) and bounding box $\boldsymbol{f}_v^1 \in \mathbb{R}^6$. Manipulation Semantics: the affordance label $\boldsymbol{f}_v^2$ (e.g., graspable, liftable, pushable, etc.), and the target indicator $\boldsymbol{f}_v^3 \in \{0, 1\}$. Semantic Alignment Score $\alpha_v \in [0, 1]$, representing the object's relevance to instruction l. This compact set of semantic embeddings effectively aligns the language instruction with visual perception, providing a precise, task-level semantic foundation for subsequent structured graph construction and dynamic control.

Semantic Collaborative Graph Modeling

To address the challenges of structural coupling and nonlinear coordination constraints in the dual-arm system's high-dimensional state space, we propose the Semantically Augmented Dynamic Spatio-Temporal Graph ($\mathcal{G}_{ST}$) modeling mechanism. $\mathcal{G}_{ST}$ is designed to uniformly encode dual-arm states, physical structure priors, and high-level semantic information. This provides a structured, explainable, and collaborative state representation for subsequent GNN inference and task planning. We formally define the spatio-temporal graph as $\mathcal{G}_{ST} = \langle \mathcal{V}_{ST}, \mathcal{E}_{ST} \rangle$, where $\mathcal{V}_{ST}$ is the set of nodes and $\mathcal{E}_{ST}$ is the set of edges.

Construction of the Node Set $\mathcal{V}_{ST}$. The node set $\mathcal{V}_{ST}$ is the union of spatial node sets across all time steps $\mathcal{V}_{ST} = \bigcup_{t=0}^{T} \mathcal{V}_S^t, t \in \{0, \ldots, T\}$. The spatial node set $\mathcal{V}_S^t$ at each time step t comprises two types of nodes. The first type are Joint Nodes $v_i^{\text{joint}} \in \mathcal{V}_S^t$, whose feature vector $\boldsymbol{f}_i^4 \in \mathbb{R}^6$ consists of the joint end-effector pose $[x, y, z, \boldsymbol{Roll}, \boldsymbol{Pitch}, \boldsymbol{Yaw}]$ and includes a binary identifier $\boldsymbol{f}_i^5 \in \{0, 1\}$ to distinguish between the left and right manipulators. The second type are Environment/Object Nodes $v_k^{\text{env}} \in \mathcal{V}_S^t$, constructed from the semantic priors $\boldsymbol{F}_v$ extracted in Section 4.1, which encode the object's geometric pose, affordance, and semantic alignment score.

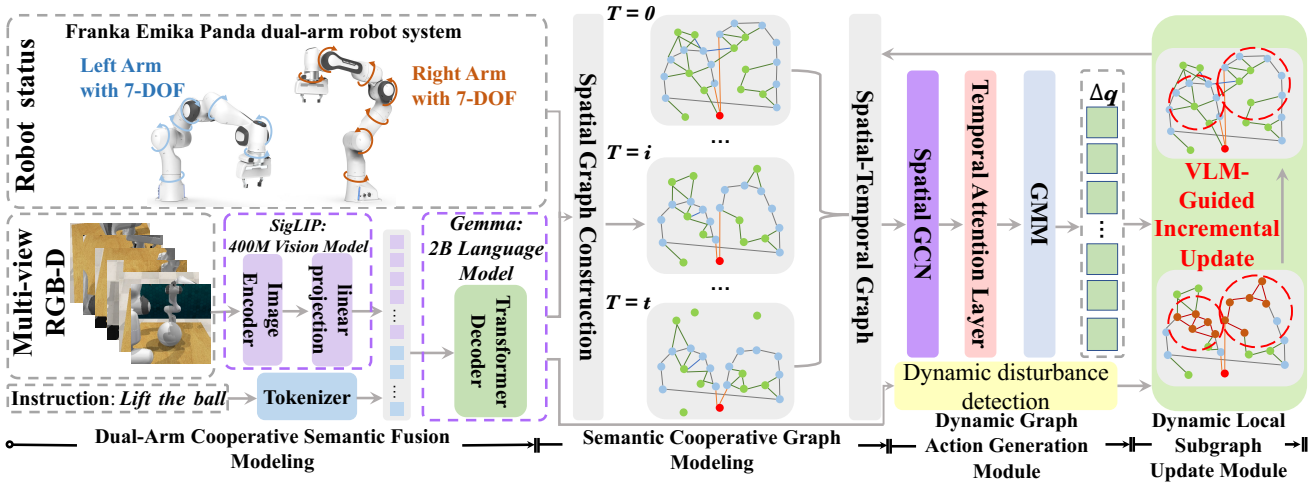

Figure 2: Overview of the Dynamic Graph-VLM (DGV) Framework. DGV integrates multimodal inputs (multi-view RGB-D images and natural language instructions) with dual-arm joint states. The PaliGemma model first performs semantic alignment between image and language, identifying target objects and task constraints. This extracted information forms the basis for constructing a multi-source fusion spatial graph, which evolves into a spatio-temporal graph as planning progresses over time. In each planning step, a Dynamic Graph Neural Network (DGNN) performs robust action sequence inference and generates action increment sequences. During execution, the system continuously monitors for dynamic perturbations (semantic or environmental changes). A Local Subgraph Update Mechanism then performs incremental structural adjustments on the affected graph regions, enabling online action sequence optimization and closed-loop control.

Construction of the Edge Set $\mathcal{E}_{ST}$. The edge set $\mathcal{E}_{ST}$ consists of spatial edges $\mathcal{E}_S^t$ and temporal edges $\mathcal{E}_{\text{temporal}}$, which encode constraints, relationships, and sequential dependencies among the nodes. Spatial Edges $\mathcal{E}_S^t$ exist between nodes at the same time step t, modeling static constraints and semantic relationships. These are categorized as: Joint-Joint Edges, based on the kinematic topology of the robot links, modeling high-dimensional coupling constraints; Joint-Env. Edges, based on collision checking and operational distance, modeling safety and reachability constraints; and Env.-Env. Edges, based on semantic relationships and physical proximity. Temporal Edges $\mathcal{E}_{\text{temporal}}$ connect the same node v_i between adjacent time steps t and $t+1$, capturing sequential dependencies and temporal smoothness of the action sequence.

Construction and Maintenance of the Dynamic Spatio-Temporal Graph. The construction of $\mathcal{G}_{ST}$ utilizes a sliding time window for online maintenance, ensuring the graph structure remains real-time. The initial graph $\mathcal{G}_0$ is constructed based on the initial joint state s_0 and environment nodes. At each subsequent time step $t+1$, new joint nodes are constructed using the next joint state s_{t+1} outputted by the action sequence inference module. Based on a local environmental stability assumption, environment nodes from the previous time step are inherited as current environment nodes. This allows for the incremental evolution of the graph through the construction of new spatial and temporal edges. By enforcing a fixed window length W, the structure corresponding to the oldest time step is removed, thereby guaranteeing computational efficiency and real-time performance.

Dynamic Local Subgraph Update Module

In dynamic collaborative manipulation, environmental perturbations (e.g., target displacement, object addition/removal) can invalidate previously planned action sequences. To address this, we design a vision-semantic driven online local subgraph update mechanism. This mechanism utilizes VLM perceptual feedback and an attention mechanism to dynamically identify and correct perturbed graph nodes and their neighbors, enabling incremental action sequence reconstruction. For real-time performance, adjustments are restricted to the predicted spatial graph $\mathcal{G}_S^{t+1}$ of the next time step. Because physical disturbances typically affect only a local subset of objects or robot joints, full graph recomputation is unnecessary. This effectively balances resource consumption and system robustness.

Perturbation Detection based on Multimodal Difference Metric. During task execution, the system continuously monitors the environment via VLM processing of real-time visual input $\mathcal{O}_t$. To quantify the impact of environmental changes on task-relevant states, we define a semantic and geometric difference metric δ_v for each environment node $v \in \mathcal{V}^{\text{env}}$. This metric synthesizes changes in geometry, size, manipulation semantics, and VLM confidence over adjacent time steps t and $t-1$:

$$\delta_v = \left\|\boldsymbol{p}_t - \boldsymbol{p}_{t-1}\right\|_2 + \left\|\boldsymbol{f}_t^1 - \boldsymbol{f}_{t-1}^1\right\|_2 + \left\|\boldsymbol{f}_t^2 - \boldsymbol{f}_{t-1}^2\right\|_2 + \left\|\boldsymbol{f}_t^3 - \boldsymbol{f}_{t-1}^3\right\|_2 + \left\|\alpha_t - \alpha_{t-1}\right\|_2 \tag{1}$$

The term $\left\|\boldsymbol{f}_t^2 - \boldsymbol{f}_{t-1}^2\right\|_2$ specifically measures the displacement in the multi-dimensional affordance score space.

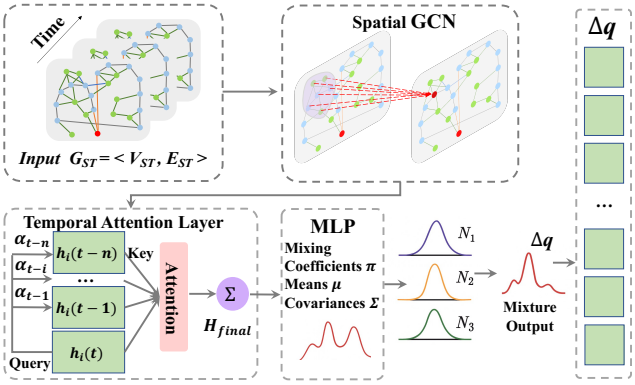

Figure 3: Dynamic Graph-Guided Action Sequence Inference Module. The module receives the semantically augmented dynamic spatio-temporal graph $\mathcal{G}_{ST}$ as input. The encoder first transforms the graph structure into node and edge embeddings. These embeddings are then fed into a Spatial GCN for structured feature aggregation, capturing the dual-arm system's structural coupling constraints and environmental semantic priors. Following spatial modeling, a Temporal Attention layer captures the action sequence's sequential dependencies and temporal coherence. The resulting spatio-temporal joint embedding is passed to the decoder, which employs a Gaussian Mixture Model (GMM) to predict the multimodal action distribution. This prediction is subsequently refined by an MLP to output the final action increment sequence $\Delta\Pi$ used to drive the manipulators.

If $\delta_v > \tau$ (a preset threshold), v is identified as a perturbed node and added to the set $\mathcal{V}^{\text{perturbed}}$.

Attention Mechanism-Guided Local Feature Fusion. For each perturbed node $v_p \in \mathcal{V}^{\text{perturbed}}$, we define a local subgraph $\mathcal{G}_{v_p}^{\text{sub}} = \langle \mathcal{V}_{v_p}^{\text{sub}}, \mathcal{E}_{v_p}^{\text{sub}} \rangle$, where the node set $\mathcal{V}_{v_p}^{\text{sub}} = \{v_p\} \cup \mathcal{N}(v_p)$ includes v_p and its first-order neighbors $\mathcal{N}(v_p)$. We introduce a feature fusion strategy based on the Graph Attention Network (GAT) (Veličković et al. 2017) framework, distinctively incorporating the latest VLM perceptual feedback. We first obtain the candidate new feature v_p^{new} provided by the VLM. Through the GAT mechanism, this feature is weighted and fused with the contextual information from the neighbors $\mathcal{N}(v_p)$, generating an enhanced node representation h_p^{new}. The core of this fusion process lies in the calculation of the attention weight α_{pj}, which dynamically assesses the relevance of the new feature to the neighborhood context, ensuring consistency between the new information and the local structure. Finally, all updated local subgraphs $\mathcal{G}_{v_p}^{\text{sub}}$ are aggregated back into the global spatio-temporal graph $\mathcal{G}_{ST}$, realizing the incremental online reconstruction of the dynamic graph structure.

Dynamic Graph Action Generation Module

After the online local subgraph update detailed in Section 4.3, we obtain the dynamic graph representation $\mathcal{G}_{ST}$, which reflects the latest environmental state. This module aims to generate a smooth and robust dual-arm action increment sequence $\Delta\Pi$ based on $\mathcal{G}_{ST}$ to realize dynamic closed-loop control. As shown in Figure 3, we adopt a hierarchical (spatial-temporal) architecture for joint modeling and employ a Gaussian Mixture Model (GMM) decoder to support multimodal prediction of the action distribution.

Spatial Graph Convolutional Layer (Spatial GCN). We first employ a Graph Convolutional Network (GCN) (Yan, Xiong, and Lin 2018) to model spatial dependencies among the node features in $\mathcal{G}_{ST}$. The GCN aggregates the features of node $\boldsymbol{h}_j^{(l)}$ and its neighbors $\mathcal{N}(i)$, encoding the physical coupling of the dual-arm links, collision constraints, and environmental semantic associations. Through multi-layer spatial aggregation, node features are transformed into spatially embedded representations $\mathcal{H}_{\text{spatial}}$, enhanced with local structure and semantics.

Temporal Dependency Modeling and Contextual Enhancement. After obtaining the spatially enhanced embeddings $\mathcal{H}_{\text{spatial}}$, we introduce a module based on the temporal graph attention mechanism (Yan, Xiong, and Lin 2018) to capture sequential action sequence dependencies. This module uses a self-attention mechanism to perform weighted aggregation over the historical states of the same node along the time axis. It computes the relevance α_τ between the current state $\boldsymbol{h}_i(t)$ and the historical action sequence features $\mathcal{H}_{1:t-1}$, selectively focusing on critical information from past actions. This mechanism effectively improves the temporal coherence and smoothness of the predicted action sequence, yielding the final spatio-temporal joint embedding $\mathcal{H}_{\text{final}}$.

GMM Decoding and Action Increment Prediction. Following the fusion of spatial and temporal features, we obtain the final spatio-temporally enhanced representation $\mathcal{H}_{\text{final}}$. To accurately capture the multimodality of action selection in dynamic environments, we employ a Gaussian Mixture Model (GMM) as the decoder. The conditional distribution of the action increment $\Delta\Pi_t$, $P(\Delta\Pi_t|\mathcal{H}_{\text{final}})$, is modeled as a mixture of M Gaussian components:

$$P(\Delta\Pi_t|\mathcal{H}_{\text{final}}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\mu_m, \Sigma_m) \tag{2}$$

where π_m is the mixing coefficient, and μ_m and Σ_m are the mean and covariance of the m-th component, respectively. The model is trained using maximum likelihood estimation (MLE). During inference, the continuous, smooth, and executable dual-arm action increment sequence $\Delta\Pi$ is generated via sampling.

Trajectory Continuity and Dynamic Adaptability. Training introduces an L_2 regularization term to constrain successive action increments, ensuring trajectory continuity and executability. Using the dynamic graph update described in Section 4.3, this module instantly recalculates action increments based on the corrected graph, realizing closed-loop control and online adaptation to dynamic environments.

Experiments and Evaluation

We evaluate the proposed DGV framework across various dual-arm collaborative tasks to rigorously test its robustness, adaptability to dynamic changes, and generalization capability. Our validation encompasses both simulated and physical robot platforms.

Dataset and Evaluation Settings

Simulation Environment and Dataset. The simulation environment, in CoppeliaSim, uses the 7 DOF Franka Emika Panda dual-arm model, replicating physics and multi-view vision for efficient data collection and comprehensive validation. Our dataset is constructed using the RLBench2 benchmark (Grotz et al. 2024) within CoppeliaSim, with 13 challenging dual-arm collaborative manipulation tasks (e.g., box pushing, lift the ball). To rigorously evaluate DGV's dynamic adaptability, the dataset includes real-time disturbances: random target pose shifts, dynamic obstacle changes (appearance/disappearance), and on-the-fly replanning after failed grasps. The system operates at 20 Hz, with target displacement perturbations ranging from 5–15 cm applied during evaluation. Approximately 200 action sequences were collected per task for training and validation.

Hardware Platform and Real-World Dataset. We also validate our method on platforms, utilizing the Unitree G1 dual-arm robot ($\approx$ 28 DOF in total, including two 7 DOF arms and Dex3-1 end-effectors). Perception is handled by four Intel RealSense D435 RGB-D cameras. We collected 400 real execution action sequences on the Unitree G1 for two representative tasks (ball lifting and block passing). The system maintains a control frequency of 10 Hz on the physical platform. This dataset, created via manual keyframe annotation and high-precision visual tracking, is used to evaluate the model's execution robustness and its ability to generalize across diverse, real-world deployment scenarios.

Evaluation Metrics. Policy performance is quantified using the Task Success Rate (TSR). The success criteria for each task are defined by geometric error thresholds applied to the final goal state (e.g., object pose or gripper configuration). Figure 4 shows examples of our experimental setups.

Comparisons with SOTA Methods

Baselines. To comprehensively validate the effectiveness and superiority of the proposed DGV framework, we select representative policy learning methods based on Transformer architectures, Diffusion Models, and Large Vision-Language Models (VLMs) as baselines for comparative evaluation: **(1) Action Chunking Transformer (ACT)** (Zhao et al. 2023), a Transformer-based architecture that uses action chunking for efficient long-horizon action sequence planning; **(2) PetACT2** (Grotz et al. 2024), an improved variant of ACT, featuring enhanced representation and execution mechanisms for superior action sequence precision and generalization; **(3) Robotic Diffusion Transformer (RDT)** (Liu et al. 2024), which combines the probabilistic modeling power of Diffusion Models with Transformer sequence learning for multimodal action sequence generation; **(4) Diffusion Policy (DP)** (Chi et al. 2025), a classic diffusion policy model that directly models the action distribution via a denoising process, excelling in multimodal action sequence expression; **(5) 3D Diffusion Policy (3DP)** (Ze et al. 2024), an extension of DP that integrates 3D spatial perception modeling to enhance robustness under complex geometric constraints; **(6)** π_0 (Black et al. 2024), a large Vision-Language-Action (VLA) model achieving zero-shot generalization across various manipulation tasks through extensive pre-training; **(7)** π_0**-Fast** (Pertsch et al. 2025), a lightweight variant of π_0, optimized to improve inference speed for real-time task execution; and **(8) Robotwin** (Mu et al. 2024), a multimodal robotics learning framework focused on efficient action sequence generation and fast adaptation under language constraints.

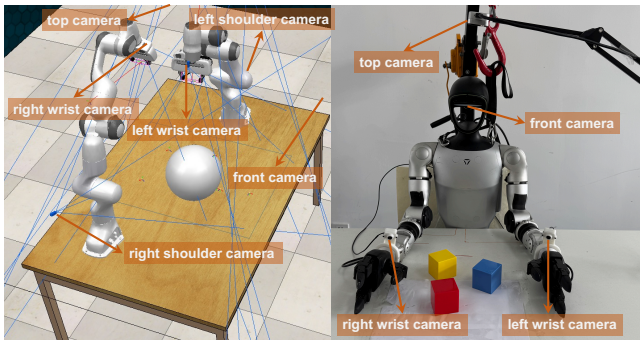

Figure 4: Experimental Environment. The figure compares the evaluation environments: the left image shows the simulated ball lifting task using the Franka Emika Panda model, and the right image illustrates the real-world execution environment utilizing the Unitree G1 manipulator system.

Simulation Results. As shown in Table 1, trained with only 200 demonstration samples, the DGV framework significantly outperforms all baseline methods across every task, achieving an overall success rate of $\mathbf{72.1}\%$. This represents an average performance gain of nearly $\mathbf{20}\%$ over the second-best method (π_0). In complex collaborative tasks like Lift Ball and Handover Item, DGV's success rate surpasses the runner-up baselines, demonstrating a critical advantage where conventional Imitation Learning (IL) methods (ACT, DP) largely fail. This validates the superior efficacy of the semantically augmented collaborative state graph in modeling dual-arm coupling constraints and achieving high-precision pose control. Regarding dynamic adaptability, DGV's success rate on the highly dynamic Sweep Dustpan task significantly exceeds that of π_0-fast and 3DP. This strongly confirms that the Local Subgraph Update mechanism effectively integrates VLM perceptual feedback to achieve robust action sequence correction and closed-loop control in highly dynamic settings.

Real-World Results. To assess the model's cross-domain robustness, all policies were tested on the Unitree G1 real-world robot platform, with results shown in Table 2. The DGV framework exhibits a significant advantage across both tasks, achieving an overall success rate of $\mathbf{73.7}\%$, which

Method	Push Box	Lift Ball	Handover Item	Pick Laptop	Sweep to Dustpan	Overall
ACT	49.8 ± 7.2	37.2 ± 1.8	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	17.4 ± 1.2
PetACT2	75.7 ± 7.6	51.2 ± 3.4	15.4 ± 0.8	38.4 ± 4.0	7.2 ± 2.3	37.8 ± 2.1
RDT	80.2 ± 3.2	51.4 ± 6.2	16.9 ± 3.8	41.8 ± 4.1	11.5 ± 0.8	40.4 ± 1.4
DP	60.8 ± 7.6	43.5 ± 1.7	0.0 ± 0.0	1.3 ± 0.6	0.0 ± 4.0	21.1 ± 1.6
3DP	60.1 ± 3.2	61.5 ± 2.8	0.0 ± 0.0	8.4 ± 3.1	0.8 ± 1.1	26.7 ± 2.0
π_0	81.2 ± 4.3	61.4 ± 8.2	32.5 ± 1.2	47.4 ± 2.2	45.3 ± 3.1	53.6 ± 1.7
π_0-fast	80.3 ± 1.8	61.6 ± 4.3	31.6 ± 1.6	45.1 ± 2.8	40.3 ± 5.4	51.8 ± 1.8
Robotwin	56.7 ± 6.2	36.3 ± 4.1	31.2 ± 1.2	47.3 ± 2.4	33.3 ± 6.8	41.0 ± 2.2
DGV (Ours)	$\mathbf{84.1 \pm 2.3}$	$\mathbf{96.4 \pm 1.2}$	$\mathbf{74.4 \pm 2.4}$	$\mathbf{51.3 \pm 4.1}$	$\mathbf{54.1 \pm 2.3}$	$\mathbf{72.1 \pm 1.8}$

Table 1: Simulation Task Performance Comparison.Task Success Rate (%) of the DGV framework against various baselines (in the previous section) policies using 200 demonstration samples in the RLBench simulation environment. All policies are evaluated over 5 independent runs with 50 test episodes each. The best performance is highlighted in bold.

Method	Lift Ball	Handover	Overall
ACT	35.8 ± 8.3	0.0 ± 0.0	17.9 ± 2.6
PetACT2	50.7 ± 7.6	8.8 ± 3.2	29.8 ± 3.6
DP	40.3 ± 7.6	0.0 ± 0.0	20.2 ± 2.8
3DP	60.1 ± 3.2	0.0 ± 0.0	30.1 ± 2.0
π_0	61.3 ± 5.8	31.0 ± 2.8	46.2 ± 4.4
DGV (Ours)	$\mathbf{86.5 \pm 1.4}$	$\mathbf{60.8 \pm 6.8}$	$\mathbf{73.7 \pm 4.3}$

Table 2: Real-World Task Performance on Unitree G1. Task Success Rate (%) comparison of the DGV framework against various baselines using 400 demonstration samples on the Unitree G1 real-world robot platform. All policies are evaluated over 5 independent runs with 50 test episodes each. The best performance is highlighted in bold.

setting	Handover-S	Handover-R	Overall
w/o STG	35.5 ± 2.6	28.9 ± 4.2	32.2 ± 2.5
w/o DSG	62.6 ± 1.4	51.7 ± 5.6	57.2 ± 4.3
w/o JOC	50.4 ± 3.8	43.3 ± 4.3	46.9 ± 6.3
w/o DA	68.8 ± 2.3	58.2 ± 1.4	63.5 ± 3.8
Full DGV	$\mathbf{74.4 \pm 2.4}$	$\mathbf{60.8 \pm 6.8}$	$\mathbf{67.6 \pm 4.6}$

Table 3: Ablation of DGV Modules. Success Rate (%) results for key module ablations of the DGV framework, evaluated on the Handover task in both simulation (Handover-S) and real-world (Handover-R) environments. Settings are defined as: w/o STG (without Spatio-Temporal Graph Modeling); w/o DSG (without Dynamic Subgraph Adjustment); w/o JOC (without Joint-Object Constraints); w/o DA (replaced with a single-modality decoder). All tests were evaluated over 5 independent runs with 50 episodes each.

represents a $\mathbf{27.5}\%$ improvement over the next-best method (π_0). Conventional Behavioral Cloning (ACT) and Diffusion Policies (DP, 3DP) completely fail on the Handover task and show limited performance on Lift Ball. This highlights that their lack of explicit structural inductive bias makes them ineffective in handling the multi-agent interaction and complex temporal constraints in real-world environments. DGV achieves $\mathbf{86.5}\%$ success on Lift Ball and $\mathbf{60.8}\%$ on Handover. This substantial performance gain is primarily attributed to the semantically augmented dynamic graph structure modeling, which effectively integrates multimodal semantic priors and real-time state feedback. This mechanism enables the model to adjust the dual arms' spatio-temporal coordination during inference, ensuring stable and robust task execution despite visual noise and control errors of the physical environment.

Ablation Studies

To quantify the contribution of each key component within the DGV framework, we performed a series of ablation studies on the challenging Handover task in both simulation (Sim) and real-world (Real) environments. The specific ablation settings are detailed in Table 3.

Ablation results demonstrate the most significant performance drop upon removing the spatio-temporal graph structure (w/o STG). This highlights that temporal dimension modeling is crucial for capturing complex sequential dependencies in dual-arm interaction, as its absence severely degrades the temporal coherence and inter-limb coordination of action sequence planning. Removing the dynamic subgraph adjustment mechanism (w/o DSG) also results in a substantial performance decline, confirming the critical role of the VLM-driven dynamic graph update in dynamically adapting to real-time environmental changes; its absence leads to policy rigidity and noticeable feedback lag. Furthermore, the removal of Joint-Object Constraint modeling (w/o JOC) causes a moderate performance decrease, emphasizing that explicitly modeling precise interactions between the joint-level topological structure and the environment is indispensable for efficient information flow and accurate motion control during complex action generation. Finally, replacing the GMM with a single-modality decoder (w/o DA) results in a slight performance reduction, confirming the GMM's clear advantage in expressing probabilistic uncertainty and predicting multimodal actions for complex collaborative motions.

To visually validate DGV's real-time robustness in dynamic environments, we conducted a comprehensive qual-

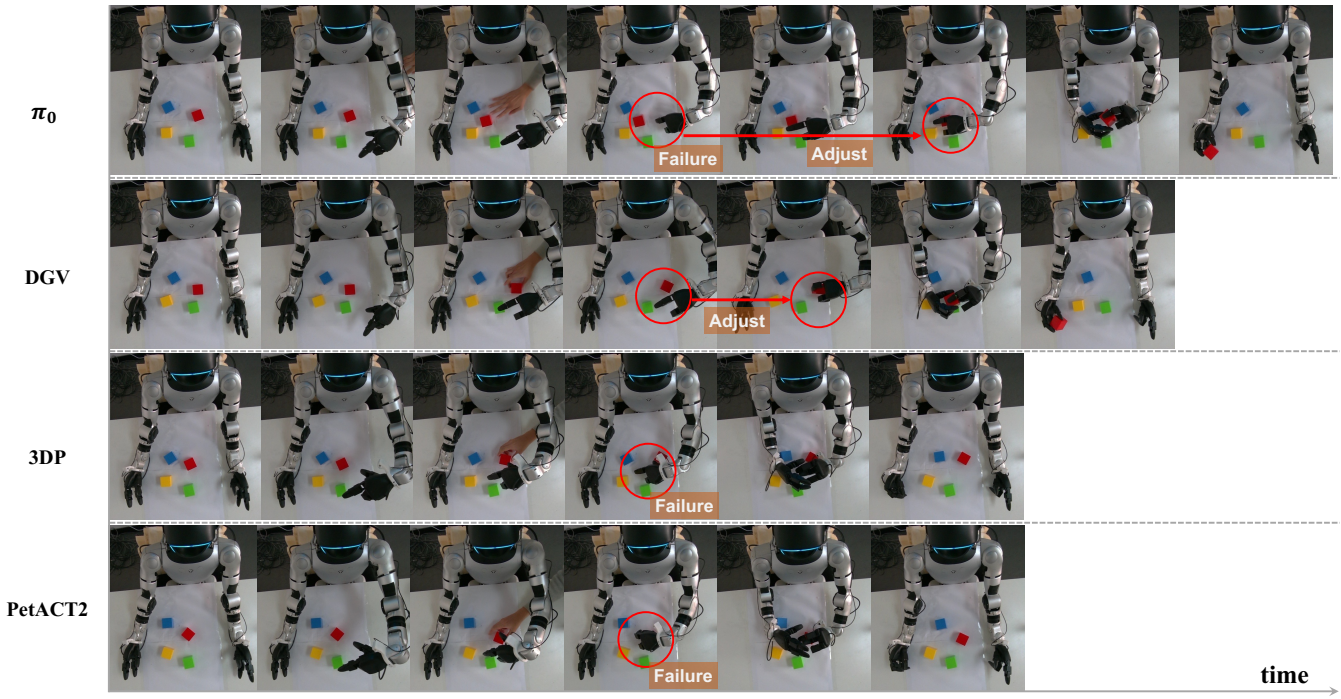

Figure 5: Qualitative Comparison of DGV and Baseline Methods under Real-World Dynamic Perturbations. This experiment demonstrates a real-time displacement perturbation introduced to the target object during the Handover task. The timeline progresses from left to right. DGV successfully corrects its action sequence and completes the task using the dynamic subgraph adjustment mechanism, showing immediate adaptation. In contrast, 3DP and PetACT2 exhibit significant planning rigidity and action drift upon encountering the perturbation, leading to immediate task failure (marked 'Failure' in red circles). π_0 experiences an initial grasping failure (Failure) at the original planned position but subsequently successfully re-plans a new action sequence and completes the task; however, this process takes significantly longer than DGV.

itative analysis of the Handover task on the physical robot platform against PetACT2, 3DP, and π_0 (Figure 5). During testing, we manually introduced a stochastic real-time displacement perturbation to the target object, forcing the policies to perform rapid online correction under pressure. Traditional Imitation Learning (PetACT2) and diffusion-based policies (3DP) exhibited significant action sequence rigidity and irreversible action drift when faced with dynamic changes, resulting in immediate grasp failure. This suggests these methods lack effective internal state update mechanisms necessary to adapt to rapid structural changes in the task space. Although the VLA model π_0 demonstrated some capacity for adjustment to the perturbation, its correction speed was slow and computationally intensive, requiring significantly more time steps than DGV to complete the task, leading to low execution efficiency. In contrast, DGV generated action sequences that adapted to the scene changes in a real-time and smooth manner, rapidly adjusting the planned action sequence to successfully complete the grasp through a high-frequency closed-loop control feedback loop.

Conclusion and Future Work

This study presents DGV, a robust dual-arm planning framework that synergistically integrates multimodal visual understanding with structured spatio-temporal graph representations to capture complex environmental dependencies. By leveraging dynamic VLM-driven local subgraph updates, the system enables robots to achieve stable, reliable execution even amidst rapid and unpredictable environmental perturbations. This dynamic graph effectively bridges high-level semantic planning and low-level physical control, efficiently modeling intricate inter-arm coordination and stringent kinematic constraints within a unified architecture. Extensive experiments conducted across diverse simulation and real-world settings demonstrate that DGV significantly outperforms state-of-the-art baselines, markedly improving success rates and operational robustness.

Despite these strengths, DGV currently has limitations in autonomously decomposing complex, long-horizon manipulation tasks into executable sub-goals. Future research will explore more sophisticated hierarchical planning architectures and extend the framework to open-world scenarios requiring enhanced zero-shot generalization across unseen objects. Furthermore, we aim to incorporate predictive world models to explicitly anticipate future scene evolution. Such advancements will empower the system to transition from reactive corrections to fully proactive behavior, generating forward-looking action sequences that dynamically anticipate environmental changes before they occur.

References

Ahn, M.; Brohan, A.; Brown, N.; Chebotar, Y.; Cortes, O.; David, B.; Finn, C.; Fu, C.; Gopalakrishnan, K.; Hausman, K.; et al. 2022. Do as I Can, Not as I Say: Grounding Language in Robotic Affordances. *arXiv preprint:2204.01691*.

Berenson, D.; Srinivasa, S.; and Kuffner, J. 2011. Task Space Regions: A Framework for Pose-Constrained Manipulation Planning. *The International Journal of Robotics Research*, 30(12): 1435–1460.

Beyer, L.; Steiner, A.; Pinto, A. S.; Kolesnikov, A.; Wang, X.; Salz, D.; Neumann, M.; Alabdulmohsin, I.; Tschannen, M.; Bugliarello, E.; et al. 2024. PaliGemma: A Versatile 3B VLM for Transfer. *arXiv preprint:2407.07726*.

Biswas, J.; and Veloso, M. 2012. Depth Camera Based Indoor Mobile Robot Localization and Navigation. In *IEEE International Conference on Robotics and Automation*, 1697–1702.

Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; et al. 2024. $\pi 0$: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint:2410.24164*.

Chi, C.; Xu, Z.; Feng, S.; Cousineau, E.; Du, Y.; Burchfiel, B.; Tedrake, R.; and Song, S. 2025. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. *The International Journal of Robotics Research*, 44(10-11): 1684–1704.

Cohen, B.; Chitta, S.; and Likhachev, M. 2012. Search-Based Planning for Dual-Arm Manipulation with Upright Orientation Constraints. In *IEEE International Conference on Robotics and Automation*, 3784–3790.

Dehio, N.; Wang, Y.; and Kheddar, A. 2022. Dual-Arm Box Grabbing with Impact-Aware MPC Utilizing Soft Deformable End-Effector Pads. *IEEE Robotics and Automation Letters*, 7(2): 5647–5654.

Driess, D.; Xia, F.; Sajjadi, M. S.; Lynch, C.; Chowdhery, A.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; et al. 2023. PALM-E: An Embodied Multimodal Language Model. *arXiv preprint:2303.03378*.

Ebert, F.; Finn, C.; Dasari, S.; Xie, A.; Lee, A.; and Levine, S. 2018. Visual Foresight: Model-Based Deep Reinforcement Learning for Vision-Based Robotic Control. *arXiv preprint:1812.00568*.

Grotz, M.; Shridhar, M.; Chao, Y.-W.; Asfour, T.; and Fox, D. 2024. PerAct2: Benchmarking and Learning for Robotic Bimanual Manipulation Tasks. In *CoRL Workshop on Whole-body Control and Bimanual Manipulation: Applications in Humanoids and Beyond*.

Huang, W.; Wang, C.; Zhang, R.; Li, Y.; Wu, J.; and Fei-Fei, L. 2023. VoxPoser: Composable 3D Value Maps for Robotic Manipulation with Language Models. *arXiv preprint:2307.05973*.

Jiang, M.; Fan, M.-Q.; Li, A.-M.; Rong, X.-W.; Kong, H.; and Song, R. 2016. Coordination Control of Dual-Arm Robot Based on Modeled Predictive Control. In *IEEE International Conference on Real-time Computing and Robotics*, 495–499.

Jiang, Y.; Gupta, A.; Zhang, Z.; Wang, G.; Dou, Y.; Chen, Y.; Fei-Fei, L.; Anandkumar, A.; Zhu, Y.; and Fan, L. 2022. VIMA: General Robot Manipulation with Multimodal Prompts. *arXiv preprint:2210.03094*, 2(3): 6.

Kaelbling, L. P.; and Lozano-Pérez, T. 2013. Integrated Task and Motion Planning in Belief Space. *The International Journal of Robotics Research*, 32(9-10): 1194–1227.

Kajita, S.; Kanehiro, F.; Kaneko, K.; Fujiwara, K.; Harada, K.; Yokoi, K.; and Hirukawa, H. 2003. Resolved Momentum Control: Humanoid Motion Planning Based on the Linear and Angular Momentum. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, 1644–1650.

Khan, A.; Ribeiro, A.; Kumar, V.; and Francis, A. G. 2020. Graph Neural Networks for Motion Planning. *arXiv preprint:2006.06248*.

Kumar, A.; Zhou, A.; Tucker, G.; and Levine, S. 2020. Conservative Q-Learning for Offline Reinforcement Learning. *Advances in Neural Information Processing Systems*, 33: 1179–1191.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. BLIP: Bootstrapping Language-Image Pre-Training for Unified Vision-Language Understanding and Generation. In *International Conference on Machine Learning*, 12888–12900.

Lin, Y.; Wang, A. S.; Undersander, E.; and Rai, A. 2022. Efficient and Interpretable Robot Manipulation with Graph Neural Networks. *IEEE Robotics and Automation Letters*, 7(2): 2740–2747.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual Instruction Tuning. *Advances in Neural Information Processing Systems*, 36: 34892–34916.

Liu, S.; Wu, L.; Li, B.; Tan, H.; Chen, H.; Wang, Z.; Xu, K.; Su, H.; and Zhu, J. 2024. RDT-1B: A Diffusion Foundation Model for Bimanual Manipulation. *arXiv preprint:2410.07864*.

Mu, Y.; Chen, T.; Peng, S.; Chen, Z.; Gao, Z.; Zou, Y.; Lin, L.; Xie, Z.; and Luo, P. 2024. Robotwin: Dual-Arm Robot Benchmark with Generative Digital Twins. In *European Conference on Computer Vision*, 264–273. Springer.

Pareja, A.; Domeniconi, G.; Chen, J.; Ma, T.; Suzumura, T.; Kanezashi, H.; Kaler, T.; Schardl, T.; and Leiserson, C. 2020. EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs. In *AAAI Conference on Artificial Intelligence*, volume 34, 5363–5370.

Pertsch, K.; Stachowicz, K.; Ichter, B.; Driess, D.; Nair, S.; Vuong, Q.; Mees, O.; Finn, C.; and Levine, S. 2025. FAST: Efficient Action Tokenization for Vision-Language-Action Models. *arXiv preprint:2501.09747*.

Pronobis, A.; and Rao, R. P. 2017. Learning Deep Generative Spatial Models for Mobile Robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 755–762.

Qureshi, A. H.; Miao, Y.; Simeonov, A.; and Yip, M. C. 2020. Motion Planning Networks: Bridging the Gap Between Learning-Based and Classical Motion Planners. *IEEE Transactions on Robotics*, 37(1): 48–66.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning Transferable Visual Models from Natural Language Supervision. In *International Conference on Machine Learning*, 8748–8763.

Ratliff, N.; Zucker, M.; Bagnell, J. A.; and Srinivasa, S. 2009. CHOMP: Gradient Optimization Techniques for Efficient Motion Planning. In *IEEE International Conference on Robotics and Automation*, 489–494.

Schulman, J.; Duan, Y.; Ho, J.; Lee, A.; Awwal, I.; Bradlow, H.; Pan, J.; Patil, S.; Goldberg, K.; and Abbeel, P. 2014. Motion Planning with Sequential Convex Optimization and Convex Collision Checking. *The International Journal of Robotics Research*, 33(9): 1251–1270.

Trivedi, R.; Farajtabar, M.; Biswal, P.; and Zha, H. 2019. DyRep: Learning Representations Over Dynamic Graphs. In *International Conference on Learning Representations*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph Attention Networks. *arXiv preprint:1710.10903*.

Yan, S.; Xiong, Y.; and Lin, D. 2018. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. In *AAAI Conference on Artificial Intelligence*, volume 32.

Zacharias, F.; Borst, C.; and Hirzinger, G. 2006. Bridging the Gap Between Task Planning and Path Planning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4490–4495.

Zang, X.; Yin, M.; Xiao, J.; Zonouz, S.; and Yuan, B. 2023. GraphMP: Graph Neural Network-Based Motion Planning with Efficient Graph Search. *Advances in Neural Information Processing Systems*, 36: 3131–3142.

Ze, Y.; Zhang, G.; Zhang, K.; Hu, C.; Wang, M.; and Xu, H. 2024. 3D Diffusion Policy: Generalizable Visuomotor Policy Learning via Simple 3D Representations. *arXiv preprint:2403.03954*.

Zhang, H.; Cheng, J.; Zhang, L.; Li, Y.; and Zhang, W. 2022. H2GNN: Hierarchical-Hops Graph Neural Networks for Multi-Robot Exploration in Unknown Environments. *IEEE Robotics and Automation Letters*, 7(2): 3435–3442.

Zhao, T. Z.; Kumar, V.; Levine, S.; and Finn, C. 2023. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. *arXiv preprint:2304.13705*.

Zheng, Y.; Wang, H.; Wei, Z.; Liu, J.; and Wang, S. 2022. Instant Graph Neural Networks for Dynamic Graphs. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2605–2615.

Zhou, Y.; Xiao, J.; Zhou, Y.; and Loianno, G. 2022. Multi-Robot Collaborative Perception with Graph Neural Networks. *IEEE Robotics and Automation Letters*, 7(2): 2289–2296.

Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlhart, P.; Welker, S.; Wahid, A.; et al. 2023. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In *Conference on Robot Learning*, 2165–2183.