ZAP: Zero-Shot Assembly Planning with Vision-Language Models

Linpeng Peng[1]*, Yanbo Wang[1,2]*, Chuanjie Lv[1], Wencan Jiang[1], Liming Xu[1], Jianbiao Mei[1], Xinyue Yao[1], and Yong Liu[1]†

[1] Zhejiang University, Hangzhou, China
[2] City University of Hong Kong, Hong Kong, China
penglinpeng@zju.edu.cn, 3220105837@zju.edu.cn, yongliu@iipc.zju.edu.cn
https://github.com/ZJU-PLP/ZAP
*Equal contribution. †Corresponding author.

Abstract. Achieving general-purpose robotic assembly is a longstanding goal in AI. Early data-driven methods, particularly those based on Reinforcement Learning (RL), made progress but are fundamentally limited by poor sample efficiency and struggle to generalize to unseen objects in zero-shot scenarios. To overcome these issues, more structured approaches have emerged. Classical planners like ASAP explicitly enforce geometric feasibility but require precise, pre-existing geometric models, which can hinder deployment under real-world perception noise. Conversely, recent Vision-Language Model (VLM)-based systems like Manual2Skill excel at semantic understanding but are critically dependent on pre-authored manuals, restricting their autonomy in unstructured environments. To address both planning-time model dependency and manual dependency, we introduce ZAP, a novel framework for zero-shot VLM-guided assembly planning with geometry-based simulation verification. ZAP constructs an internal structural representation from perception without relying on task-specific manuals, pre-authored assembly sequences, or hard-coded assembly logic. Our framework features two core modules: a VLM-Parser that analyzes segmentation-based multi-view RGB-D observations to infer an "implicit manual" and compile it into structured part dossiers, and a VLM-Planner that performs Chain-of-Thought (CoT) reasoning over these dossiers to produce physically plausible, robot-executable assembly sequences. We validate candidate plans using a geometry-based digital-twin verifier and further examine representative executions on a robotic arm. Our experiments show that ZAP remains effective in manual-free scenarios where methods like Manual2Skill are less directly applicable. Furthermore, by reasoning directly from visual perception, ZAP can be more flexible than geometry-dependent classical planners like ASAP in the evaluated settings. The framework generates and executes plans for complex items such as multi-part LEGO models and household furniture in our evaluation setting, providing evidence toward more autonomous assembly without task-specific manuals.

Keywords: Vision · Language · Reasoning · Robotics

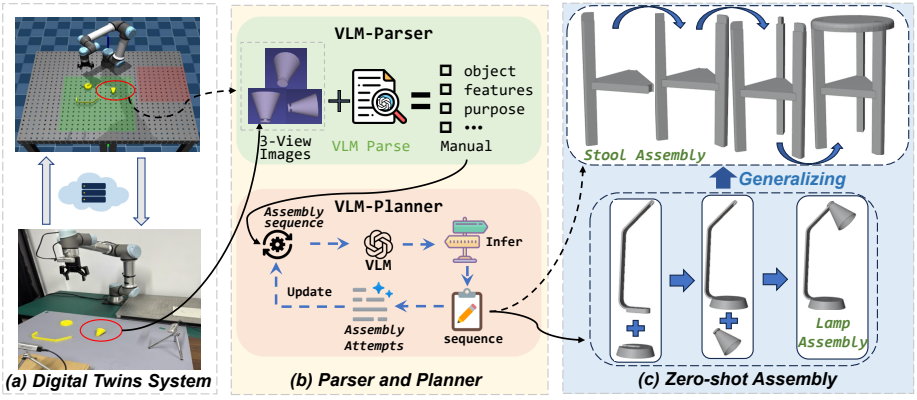

Fig. 1: Illustrating the core concept of ZAP: a VLM-guided agent transforms multi-view RGB-D observations into an assembly plan through internal reasoning, while a digital twin is used for geometry-based feasibility verification.

1 Introduction

Autonomous robotic assembly requires precise visual-spatial reasoning over many unstructured parts [4]. In practice, the problem couples three requirements: detecting fine-grained geometric interfaces, selecting physically stable assembly orders, and executing contact-rich manipulation [13, 40]. Precision-driven settings such as in-orbit assembly can rely on strict control pipelines [21]. By contrast, general assembly in open environments remains difficult. Recent CAD-based and disassembly-based planners provide useful geometric formulations [26, 32], but they assume clean 3D priors. This assumption breaks under sensor noise and unseen object topologies, especially for furniture-like tasks where manuals and part semantics are critical [10, 30].

Conventional data-driven approaches, especially RL, map pixels to assembly actions [7, 37]. They work well in constrained setups [25, 42] and have improved with automated correction modules [28]. However, they still face a generalization bottleneck in zero-shot conditions with unseen geometries. This limitation has motivated increased interest in foundation models [39]. These models encode broad physical and procedural priors and have shown utility in open-vocabulary 3D perception [36], physics-aware grasping [9, 33], and high-level planning [2, 8]. Yet systems such as Manual2Skill [27] still assume the presence of human-authored manuals. In manual-free environments, an agent must instead infer an "implicit manual" from perception and physical common sense. In this paper, planning-time model dependency refers to using precise geometric priors as primary planning inputs, while manual dependency refers to reliance on pre-authored instructions or assembly sequences.

To bridge this gap, we introduce **ZAP** (**Z**ero-shot **A**ssembly **P**lanning), a framework that reformulates robotic assembly as a perception-to-logic mapping problem. ZAP synthesizes an internal structural representation from segmentation-

based multi-view RGB-D observations without using task-specific manuals, pre-authored assembly sequences, or hard-coded assembly logic. Object meshes, when available in the benchmark simulator, are reserved for digital-twin feasibility verification rather than used as primary inputs to the VLM planner.

- **VLM-Parser for Structural Abstraction**: Rather than simple object detection, our parser implements a multi-view structural abstraction strategy. By fusing orthogonal viewpoints and leveraging VLM-based visual understanding, it transforms automatically segmented part-centric RGB-D observations into structured "Part Dossiers"—capturing functional roles, auxiliary visual attributes, and axial interface features without requiring learned task-specific 3D encoders [9, 18, 25, 27, 42].
- **VLM-Planner as Heuristic Simulation**: The planning process is framed as an iterative mental rehearsal. Leveraging Chain-of-Thought (CoT) reasoning [31, 34], the VLM-Planner evaluates candidate actions based on visual stability priors, producing physically plausible assembly sequences through advanced heuristic exploration [3, 6, 24, 29]. An **Iterative Decision Aggregator (IDA)** is employed to backtrack and recover from invalid assembly states.Candidate transitions are subsequently checked by a geometry-based digital-twin verifier, while meshes are not exposed as primary inputs to the VLM planner.

Crucially, to validate our framework in a vision-centric context, we introduce **ASAP-G**, a novel multimodal benchmark that extends the ASAP dataset [25] with high-fidelity semantic annotations. ASAP-G is constructed from locally rendered multi-view RGB-D observations and custom structured annotations, and the resulting observation-text pairs are not used for task-specific training or fine-tuning of the VLM. Our experiments indicate favorable scaling behavior; notably, ZAP attains an 88.67% mean planning success rate and a 10.00-point margin on hard assemblies (Figure 4). Sim-to-real transfer on a UR5e platform, integrated with digital-twin validation [12, 20, 23], provides empirical evidence that these plans can transfer across the evaluated household and industrial objects.

2 Related Work

2.1 Geometric and Learning-based Assembly Planning

Classical assembly planning is built on deterministic geometric primitives, including early fine-assembly frameworks [22] and decomposition-based planners [15]. These methods are effective in structured settings and can provide optimal sequences under their assumptions. Their main limitation is the model-dependency bottleneck: they require accurate CAD models that are often unavailable in real environments. Recent multi-robot extensions such as APEX-MR [14] improve coordination, but they still rely on explicit specifications and high-fidelity geometry.

To address these limitations, researchers have explored Learning from Demonstration (LfD) [43] and Reinforcement Learning (RL) [37, 38]. These paradigms learn from sensory data, but they often require many interactions and generalize poorly to unseen topologies. Hybrid pipelines with digital twins or AR-based human guidance improve transfer [35]. Still, they depend on substantial task-specific training or manual intervention. ZAP targets this gap by performing zero-shot structural inference directly from multi-view perception.

2.2 Foundation Models for Embodied Reasoning

Large Language Models (LLMs) and Vision-Language Models (VLMs) have accelerated progress toward general-purpose embodied agents. Early systems such as SayCan [1] grounded language in robot affordances, and later models such as PaLM-E [5] coupled language with sensor streams for decision-making. More recent work uses LLMs for code generation [17] and design-for-assembly optimization [8]. VLM-MSGraph [16] further improves structured understanding by building hierarchical scene graphs from multimodal observations.

These methods are strong at instruction interpretation [19] and hierarchical scene decomposition, but most remain manual-following pipelines that assume explicit blueprints or manuals [27]. In parallel, modular VLM systems have explored zero-shot tool-use through multimodal affordance modeling [41]. ZAP instead treats assembly as representation generation from observations: rather than consuming pre-authored instructions, it infers an "implicit manual" from visual evidence and model priors. This shift from manual-following to manual-inference is a step toward more general robotic assembly in unstructured settings.

3 Methodology

3.1 Problem Formulation

We formulate robotic assembly as a multimodal sequential decision-making problem. Given multi-view RGB-D observations, the goal is to recover a physically valid and logically consistent assembly trajectory $\mathcal{S}$ without using task-specific CAD models or manuals as primary planning inputs.

Let $\mathcal{P} = \{p_1, p_2, \ldots, p_K\}$ be a set of K unassembled parts [13]. We consider a controlled preparation region where all parts are placed with sufficient spacing. Three calibrated RGB-D cameras capture the preparation region from approximately orthogonal viewpoints:

$$\mathcal{O} = \{O^{front}, O^{side}, O^{top}\},$$

where each view contains all visible parts. Since the cameras are positioned with proper height differences and the parts are spatially separated, severe inter-part occlusion is largely avoided.

Based on these preparation-region observations, we apply automatic instance segmentation and part-wise cropping to obtain part-centric RGB-D three-view observations for each part:

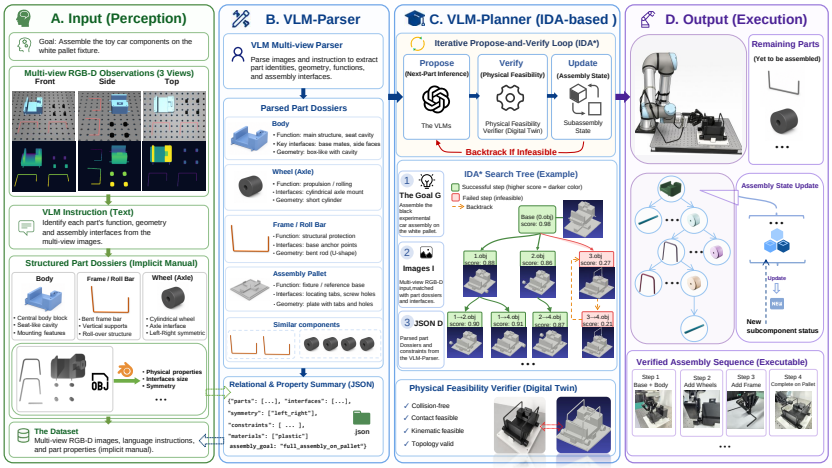

Fig. 2: Overview of our VLM-guided iterative assembly planning framework. Given multi-view RGB-D observations and a language-level assembly goal, the VLM-Parser extracts part identities, functional cues, geometric properties, and assembly interfaces, producing structured part dossiers and a relational JSON summary. The VLM-Planner then performs an IDA-based propose–verify–update search over the current subassembly state. Each candidate next-part action is checked by a digital-twin verifier for physical feasibility, and infeasible branches are rejected through backtracking. The final output is a verified, executable assembly sequence for robotic execution.

$$\mathcal{I}_k = \{I_k^{front}, I_k^{side}, I_k^{top}\}, \quad k = 1, 2, \ldots, K.$$

The complete visual input for assembly reasoning is then defined as:

$$\mathcal{I} = \{\mathcal{I}_k\}_{k=1}^{K}.$$

Each view contains an RGB crop and an aligned depth map; VLM reasoning primarily uses RGB evidence for semantic planning, while depth is used during robotic grasping to recover part-level grasp poses.

ZAP predicts an assembly plan $\mathcal{S}$ as a sequence of $K-1$ transitions:

$$\mathcal{S} = (s_1, s_2, \ldots, s_{K-1}) \tag{1}$$

Each action s_t at step t involves the integration of a single mobile part $p_{\text{mov}} \in \mathcal{P} \setminus C_{t-1}$ into the evolving sub-assembly C_{t-1}:

$$s_t : (p_{\text{mov}}, C_{t-1}) \to C_t = C_{t-1} \cup \{p_{\text{mov}}\}, \quad C_0 = \{p_{\text{base}}\} \tag{2}$$

where p_{base} is the selected foundation part. A plan $\mathcal{S}$ is physically valid if every transition s_t satisfies the fundamental Feasibility and Completeness constraints:

– **Feasibility:** Each action must be kinematically plausible, ensuring a collision-free path for p_{mov} relative to C_{t-1}.

– **Completeness:** The terminal state must encompass all initial parts, i.e., $C_{K-1} = \mathcal{P}$.

Our framework thus aims to learn a mapping $f_{\text{ZAP}} : \mathcal{I} \to \mathcal{S}$ that navigates this complex combinatorial space , where $\mathcal{I}$ is obtained from the preparation-region observations $\mathcal{O}$ through segmentation-based part-centric extraction.

3.2 ASAP-G: Physical-Semantic Alignment Benchmark

To support grounded visual-spatial reasoning, we introduce **ASAP-G**, a multimodal benchmark for assembly planning. Compared with ASAP, it adds calibrated multi-view imagery and semantic part dossiers to better align geometry with functional semantics. The multi-view observations are locally rendered and processed with automatic instance segmentation to obtain part-centric RGB-D crops.

ASAP-G contains 150 assemblies selected by stratified sampling over topology and category. We choose this scale to enable exhaustive human verification beyond automatic labeling. Each assembly includes 3–30 parts and diverse joint geometries that are challenging for zero-shot planning.

ASAP-G uses a dual-stage annotation pipeline. We first generate metadata drafts with a VLM agent. Because these drafts may contain semantic errors (e.g., wrong counts of repeated structures), researchers manually verify and revise each entry with engineering constraints. In practice, approximately 6.5% of generated part annotations required manual correction, mainly for repeated or symmetric structures such as identical screws or legs.

The resulting locally rendered image-text pairs and custom structured annotations are not used for task-specific VLM training or fine-tuning.

For sim-to-real reliability, we apply physical manifold filtering. All meshes are checked for watertightness and initial inter-part collisions are removed. The final benchmark is split into *Simple*, *Medium*, and *Hard* tiers with a 7:5:3 ratio, corresponding to 3–5, 6–10, and 11–30 parts, respectively.

3.3 VLM-Parser: Projective-to-Axial Structural Abstraction

The VLM-Parser performs a pixels-to-topology mapping. It converts the automatically extracted part-centric observations $\mathcal{I} = \{\mathcal{I}_k\}_{k=1}^{K}$ into **Structured Part Dossiers** (SPD), denoted by $D = \{D_1, \ldots, D_K\}$. This representation turns part-centric RGB-D input into a planning-ready structural description for zero-shot reasoning [5,29]. Here, the implicit manual denotes a natural-language summary of part functions and relations inferred from observations, while the structured dossiers provide a machine-readable schema of part attributes and interfaces for downstream planning.

The parser follows an orthogonal projection prior: for manufactured components, front/top/side views provide sufficient cues for the 3D envelope [25]. To reduce geometric hallucination, we use sparse structural encoding and map 2D projections to six axial directions. Each dossier D_i for part p_i has three components:

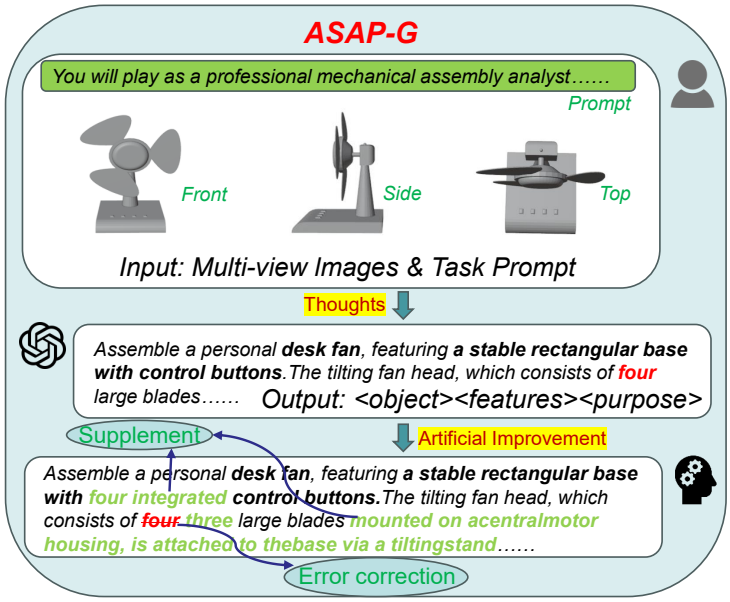

Fig. 3: ASAP-G dataset generation process. The pipeline generates semantic descriptions for 3D assemblies through VLM-assisted drafting and manual verification.

- **Functional Affordance** (`sem_sum`): Encapsulates VLM-inferred semantic attributes, including the object's nomenclature, auxiliary visual attributes such as color and material, and its latent assembly role relative to the global objective G.
- **Metric Geometric Primitives** (`phys_prop`): Incorporates deterministic spatial measurements derived from cross-view consistency, including axis-aligned bounding volumes and 3D centroids inferred from the orthogonal planes.
- **Topological Interface Manifolds** (`interfaces`): Implements a projective-to-axial mapping to identify mating features (e.g., "concentric hole", "axial plug") localized at the six primary extremities. This module reconciles the qualitative features identified in 2D projections with programmatically validated metric dimensions in 3D space.

This constrained representation acts as an inductive bias and emphasizes assembly-relevant geometry over texture details. The parser uses two streams: a high-level *semantic stream* from VLM reasoning and a low-level *deterministic geometric stream* from calibrated tools. The parser primarily relies on RGB visual evidence for semantic topology and interface reasoning, while aligned depth maps are retained for downstream grasp-pose estimation during robotic execution rather than serving as primary inputs to VLM planning.

By enforcing cross-view consistency, ZAP builds a geometrically grounded state representation. This representation serves as an "implicit assembly manual" for downstream planning.

3.4 VLM-Planner: Heuristic Search via Mental Rehearsal

The VLM-Planner treats assembly planning as heuristic state-space search with physical priors. Instead of autoregressive rollout, it uses Chain-of-Thought (CoT) reasoning to run short "mental rehearsals" before each action [31,39]. This process removes infeasible branches early and improves search efficiency in long-horizon combinatorial tasks [16, 34].

The core module is the **Iterative Decision Aggregator (IDA)**. It runs a recursive *propose-and-verify* loop to align VLM proposals with physical constraints. At each step k, IDA performs two stages:

1. **Heuristic Proposal Stage:** The VLM acts as a prior-informed proposer. It evaluates the set of available parts P_{avail} using a multi-objective scoring function $\sigma(p, C_{k-1})$. This function quantitatively weighs: (i) structural stability, the visual center-of-mass alignment relative to the support base; (ii) functional affordance, the logical role of the part in the global assembly goal G; and (iii) assembly precedence, the geometric accessibility for future components. CoT reasoning is employed to externalize the physical rationale, acting as a semantic heuristic to narrow the search frontier [26, 42].
2. **Deterministic Verification Stage:** Each high-confidence proposal is subjected to a physical consistency check against the digital-twin-based feasibility verifier G_{contact}. This verifier uses object meshes only to instantiate the simulation environment and to perform discrete collision checking and joint-alignment validation; these meshes are not exposed as primary inputs to the VLM planner. By treating G_{contact} as an environmental feedback signal rather than an oracle input, we ensure that only kinematically and topologically valid connections are integrated into the sequence [20, 25].

To ensure robustness against the perceptual ambiguities of zero-shot settings, we integrate a systematic backtracking mechanism. When the planner encounters a dead-end—defined as a state where no remaining parts satisfy the digital-twin feasibility check—it reverts to the last stable sub-assembly state and re-evaluates alternative heuristic branches. This iterative refinement, formalized in Algorithm 1, enables ZAP to discover valid assembly trajectories for complex objects where greedy search or manual-dependent policies typically fail [10, 11]. By grounding high-level reasoning in deterministic physical feedback, our planner bridges the gap between neural-symbolic intuition and rigid-body mechanics.

4 Experiments

This section evaluates our ZAP framework. We detail our experimental setup, present quantitative comparisons against state-of-the-art methods, and provide

Algorithm 1 IDA: Iterative Decision Aggregation with Backtracking

Require: Set of parts $\mathcal{P} = \{p_1, p_2, \ldots, p_K\}$, base part p_{base}, feasibility verifier G_{contact}.
Ensure: Valid assembly sequence $\mathcal{S} = (s_1, s_2, \ldots, s_{K-1})$, or FAILURE.

procedure IDA-PLANNER($\mathcal{P}, p_{\text{base}}, G_{\text{contact}}$)
  $\mathcal{S} \leftarrow$ InitializeEmptySequence()
  $C \leftarrow \{p_{\text{base}}\}$
  $\mathcal{H} \leftarrow$ InitializeHistoryLog() ▷ Stores the search tree and states
  while $|\mathcal{S}| < K - 1$ **do** ▷ $K - 1$ transitions assemble K parts
    $P_{\text{avail}} \leftarrow \mathcal{P} \setminus C$
    $p_{\text{mov}} \leftarrow$ ProposeAndVerify($\mathcal{S}, C, P_{\text{avail}}, G_{\text{contact}}$)
    if $p_{\text{mov}} \neq$ NULL **then**
      $\mathcal{S}$.append((p_{mov}, C))
      $C \leftarrow C \cup \{p_{\text{mov}}\}$
      $\mathcal{H}$.recordStep($\mathcal{S}, C$)
    else
      $(\mathcal{S}, C) \leftarrow$ Backtrack($\mathcal{H}$) ▷ Reverts to last stable sub-assembly
      if $\mathcal{S} =$ FAILURE **then**
        return FAILURE
      end if
    end if
  end while
  return $\mathcal{S}$
end procedure

ablation studies. Crucially, generated plans are validated by the digital-twin verifier, while physical-robot execution is reported only for representative physical demonstrations to assess sim-to-real feasibility.

In this section, we empirically evaluate the ZAP framework's capacity for zero-shot spatial reasoning and physical-semantic alignment. Here, physical-semantic alignment denotes consistency between semantic affordance predictions in the structured dossiers and kinematic/contact feasibility checks in the digital twin. We focus on three core objectives: (i) quantifying planning robustness across hierarchical complexity, (ii) validating the necessity of the dual-stream perceptual-logical representation, and (iii) assessing the fidelity of sim-to-real transfer within our digital twin system.

4.1 Experimental Protocol

ASAP-G Benchmark and Evaluation. To demonstrate the statistical significance of our framework, we conduct a comprehensive evaluation across the entire ASAP-G benchmark, which comprises 150 unique assemblies stratified by topological complexity. Unlike prior works that evaluate on limited subsets, we report the Planning Success Rate (PSR) on all 150 test cases to ensure a rigorous assessment of zero-shot generalization across diverse categories, including industrial components, furniture, and complex toys.

For our physical demonstrations and sim-to-real validation, we further select a representative subset of challenging assemblies (e.g., the toy car and stool) that manifest diverse geometric constraints and contact interactions. This two-tiered evaluation protocol—combining large-scale automated planning with representative physical execution—provides a robust measure of both the logical correctness and the physical viability of the ZAP framework.

Evaluation Metrics. Our primary metric is the Planning Success Rate (PSR), defined as the percentage of generated sequences that pass the digital-twin feasibility verifier and produce the target assembly topology. For assemblies with multiple valid orders, any sequence that satisfies these feasibility and connectivity criteria is counted as successful.

Hardware and Sensing. The system utilizes three extrinsic-calibrated Intel RealSense D435 cameras to capture orthogonal RGB-D views. RGB streams provide part-centric observations for VLM parsing, while aligned depth maps are used at grasp time to estimate part-level grasp poses and support metric consistency.

4.2 Experimental Setup

Our framework leverages `Gemini` and `GPT` model families for the VLM-Parser and VLM-Planner modules, with API calls configured for deterministic outputs (temperature=0.1). Our experiments are conducted on a workstation with an AMD Ryzen 9 7950X CPU and 64GB of RAM, utilizing an NVIDIA RTX 3090 GPU primarily for accelerating physics simulation and rendering. Planning performance is evaluated in a high-fidelity digital twin system, while physical demonstrations use the corresponding real-world setup. The simulation, built in MuJoCo 3.3.4, and the physical setup both feature a UR5e robotic arm, a Robotiq-85 gripper, and a custom 3D-printed assembly fixture. To acquire RGB-D observations, the physical setup is equipped with a multi-view system of three Intel RealSense D435 cameras; in our physical demonstrations, the aligned depth maps are used for execution-time grasp-pose recovery. To assess generalization across varying complexities, we test our system on three 3D-printed objects: a simple desk lamp, a moderately complex stool, and a challenging multi-part toy car.

4.3 Quantitative Success Rate Comparison

We evaluate ZAP against established geometric planners (ASAP-Heuristics/Learning) and a stochastic baseline to quantify zero-shot transferability and structural reasoning efficacy. Table 1 summarizes the Planning Success Rate (PSR) stratified by difficulty and semantic category.

In-distribution Bias and Zero-shot Boundaries. Crucially, our evaluation protocol employs a subset of the original ASAP corpus for the ASAP-G benchmark [25]. This configuration gives the baseline methods a relative in-distribution advantage, as their performance reflects access to full geometric priors and pretrained expertise on the ASAP training split [25]. In contrast, ZAP operates in a zero-shot setting, defined by three evaluation boundaries: (i) **Distribution Isolation**: no samples from the ASAP-G dataset were used for prompt-tuning or model alignment [13]; (ii) **Topology Blindness**: the agent has no prior access to ground-truth assembly graphs or target connectivity matrices; and (iii) **Zero-Tuning Policy**: ZAP utilizes a frozen VLM backbone with a fixed temperature and prompt template across all test instances to ensure evaluation fairness [13]. Despite this systemic disadvantage, ZAP achieves a mean PSR of 88.67%, demonstrating a robust capacity for cross-domain generalization [13,36].

Scaling Robustness against Combinatorial Explosion. As illustrated in Figure 4, we observe a distinct complexity-robustness trade-off. In simple scenarios (3–5 components), classical geometric solvers perform near-optimally due to the constrained search space. However, as task complexity scales, these methods suffer from performance decay due to the combinatorial explosion of potential assembly sequences. In the Hard tier, ZAP maintains a PSR of 86.67%, exceeding the ASAP and Random Permutation baselines by 10.00 points. This stability substantiates our hypothesis that integrating physical-semantic common sense allows the agent to effectively prune infeasible branches and prioritize topologically coherent trajectories.

Categorical and Difficulty Analysis. ZAP demonstrates a noticeable performance margin in the Hard assembly tier (+10.00% PSR) and the Toy category (+19.05%). These results indicate that while traditional planners struggle with the non-standardized geometries of toy assemblies, the VLM-Parser's axial interface abstraction effectively captures the functional affordances necessary for successful planning. We also observe that the Random Permutation baseline with physical verification achieves a high baseline PSR (84.67%), reinforcing the premise that our deterministic verification layer (Section 3.4) is a critical safeguard against kinematically impossible proposals. Under the same verification budget, however, Random Permutation degrades on the Hard tier, whereas ZAP maintains a higher PSR by using VLM-guided semantic pruning to narrow the combinatorial search space.

4.4 Ablation Studies

To quantitatively disentangle the contributions of our structural representation and iterative reasoning framework, we conduct systematic ablation studies across hierarchical difficulty tiers. These experiments evaluate the performance degradation relative to the full ZAP framework when key architectural components are decoupled.

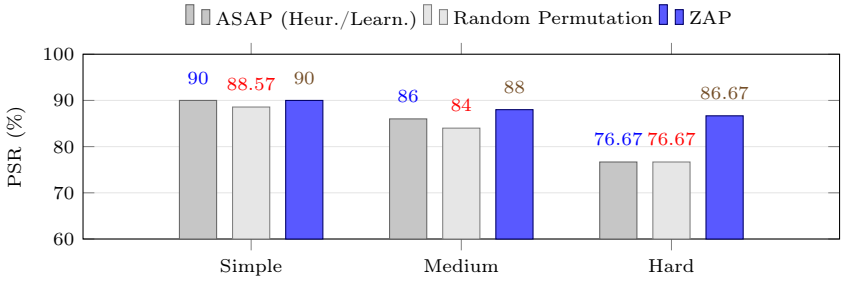

Fig. 4: Planning Success Rate (PSR, %) across task complexity tiers (Simple, Medium, Hard). ZAP consistently achieves higher mean PSR than baseline methods, with the largest margin observed on hard tasks.

Table 1: Performance comparison across difficulty and category tiers, measured by PSR (%). **Bold** indicates the best performance in each column.

Group	Method	By Difficulty			By Category				Mean
		Simple	Med.	Hard	Fact.	Furn.	Oth.	Toy	
Baseline	ASAP (Heur./Learn.)	90.00	86.00	76.67	83.87	**87.14**	**92.86**	76.19	86.00
	Random Permutation	88.57	84.00	76.67	80.65	87.14	89.29	76.19	84.67
Ours	**ZAP**	**90.00**	**88.00**	**86.67**	**93.55**	84.29	89.29	**95.24**	**88.67**

Efficacy of the Iterative Decision Aggregator (IDA). We first examine the criticality of the IDA module (Algorithm 1) by reverting the VLM-Planner to a greedy open-loop policy. This baseline commits to the VLM's primary heuristic proposal without the recursive *propose-and-verify* cycle or backtracking mechanism. As shown in Table 3, this decoupling causes substantial degradation in complex scenarios: while the Planning Success Rate (PSR) declines by a manageable 8.58% in simple assemblies, it plummets by 23.34% in the 'Hard' tier. This non-linear degradation demonstrates that as the entropy of the assembly state space increases, the digital-twin feasibility verifier (G_{contact}) acts as a strong corrective signal for spatial errors produced by the foundation model.

Multimodal Synergy: Implicit Manual vs. Perceptual Grounding. We further analyze the interplay between (i) semantic priors from the implicit manual/structured dossiers and (ii) direct visual grounding from raw images by selectively withholding input modalities:

– **Decoupling Visual Grounding:** Eliminating raw image inputs while preserving the Structured Part Dossiers (JSON) results in an overall PSR drop of 2.67%. Notably, the degradation is most acute in 'Hard' tasks (−6.67%), suggesting that while the "implicit manual" provides a robust topological prior, direct visual-spatial grounding is requisite for resolving fine-grained geometric ambiguities that linguistic descriptions alone fail to capture.

– **Decoupling Structural Priors:** Conversely, bypassing the VLM-Parser's structured dossiers forces the planner to reason directly from raw pixels. This leads to a consistent performance decay, particularly in 'Medium' tasks (−4.0%). This substantiates that the semantic-geometric dossiers effectively prune the search space by providing unambiguous functional affordances that are otherwise difficult to distill from raw vision in a zero-shot context.

The results in Table 2 support a clear synergistic trend: the robustness of our framework is not derived from a single heuristic but from the seamless integration of intuitive reasoning and deterministic physical feedback. The IDA module, in particular, fulfills the role of a cognitive-inspired verification-driven search engine, and serves as a key mechanism for managing combinatorial growth in long-horizon assembly tasks.

Table 2: Overall ablation study of the ZAP framework. We report the Planning Success Rate (PSR, %) and the performance delta (Δ) relative to the full framework.

Configuration	Success Rate (%)	Δ
ZAP (Full Framework)	**88.67**	-
w/o IDA (Greedy Policy)	74.00	−14.67
w/o Image Input	86.00	−2.67
w/o Structural Dossiers	86.67	−2.00

Table 3: Detailed ablation analysis across hierarchical difficulty tiers. Values in parentheses indicate the performance delta relative to the full framework.

Configuration	Simple	Medium	Hard
ZAP (Full)	**90.00**	**88.00**	**86.67**
w/o IDA	81.42 (−8.58)	68.00 (−20.00)	63.33 (−23.34)
w/o Image	88.57 (−1.43)	86.00 (−2.00)	80.00 (−6.67)
w/o Dossiers	88.57 (−1.43)	84.00 (−4.00)	83.33 (−3.34)

4.5 Sim-to-Real Robotic Assembly Experiments

To evaluate the transferability of the zero-shot plans generated by ZAP, we conduct representative end-to-end physical demonstrations within a verification-guided digital twin environment. This system leverages a high-fidelity MuJoCo

simulation as the digital-twin feasibility verifier G_{contact}, so that candidate transitions are kinematically validated before physical execution. Our setup comprises a UR5e manipulator, a Robotiq-85 gripper, and a multi-view sensing suite of three Intel RealSense D435 cameras (Fig. 5).

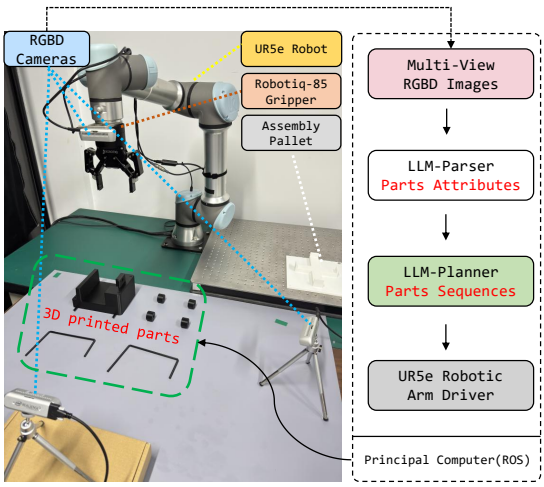

Fig. 5: UR5e real-world assembly system.

Quantitative Real-World Performance. Unlike prior manual-dependent works [27], we evaluate ZAP on three objects with increasing topological complexity: a desk lamp, an industrial stool, and a toy car. To demonstrate effectiveness, we performed 10 independent trials per object, achieving success rates of 100%, 90%, and 80%, respectively. We emphasize that these trials serve as a pilot study intended to validate the end-to-end feasibility of the ZAP framework rather than asserting exhaustive statistical completeness. The slight performance decay in the toy car assembly primarily stems from visual occlusions during the multi-stage process rather than planning failures, suggesting a limited sim-to-real gap in this pilot setup [23].

Failure Case and Robustness Analysis. We analyzed the primary failure modes in the pilot demonstrations. (i) **Perceptual Hallucination**: In 12.5% of complex cases, the VLM-Parser misidentified symmetric interfaces due to self-occlusion. (ii) **Cascading Errors**: While failures in parsing can cascade, our IDA module's backtracking mechanism recovered 60% of such cases by re-evaluating the search tree upon encountering physical resistance in the simulation. This indicates that our "propose-and-verify" paradigm can mitigate cascading planning failures in unstructured environments.

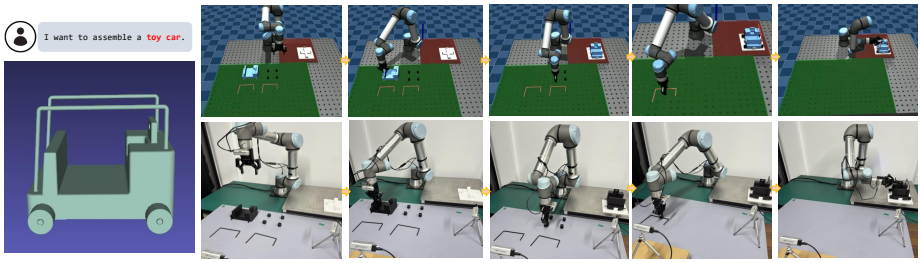

Fig. 6: Sim-to-real execution of the zero-shot assembly plan. We visualize the sequential alignment between the MuJoCo-based physical rehearsal (top) and the synchronized real-world execution on a toy car (bottom). ZAP infers assembly sequences from multi-view RGB-D observations without pre-authored manuals, while maintaining strong execution fidelity across long-horizon contact-rich stages.

5 Conclusion

In this paper, we introduced ZAP, a framework for zero-shot robotic assembly that reduces dependence on pre-existing geometric models and pre-authored manuals. Our approach, featuring a VLM-Parser for on-the-fly "implicit manual" generation and a Chain-of-Thought-based VLM-Planner, shows competitive flexibility and generalization relative to representative prior methods. While our digital twin validation showcases a promising path toward autonomous assembly, we identify three key avenues for future work. First, enhancing physical execution robustness by integrating force-feedback control and developing policies for contact-rich manipulation would allow the system to handle tighter tolerances. Second, future research should focus on enabling VLMs to reason more deeply about fine-grained geometric constraints, such as contact surfaces and mating geometries. Third, a crucial direction is to enhance the physical realism of the assembly benchmarks themselves. The geometric features at the connection points in the current dataset are not always distinct, and many parts lack the high-fidelity mechanical constraints (e.g., precise fastening mechanisms, friction models) found in real-world scenarios. Addressing these challenges is important for improving the reliability and dexterity of future robotic assembly systems.

6 Acknowledgements

This work was partially supported by the ZJU Kunpeng & Ascend Center of Excellence. We would like to express our sincere gratitude to all the staff at the ZJU Kunpeng & Ascend Center of Excellence for their valuable support and contributions to this work. We also thank the Center for providing access to the Ascend 910B and Ascend 310P computing platforms, which enabled the experimental validation conducted in this study.

References

1. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., et al.: Do as i can, not as i say: Grounding language in robotic affordances. arXiv preprint arXiv:2204.01691 (2022)
2. Ao, J., Wu, F., Wu, Y., Swikir, A., Haddadin, S.: Llm-as-bt-planner: Leveraging llms for behavior tree generation in robot task planning. arXiv preprint arXiv:2409.10444 (2024)
3. Burns, K., Jain, A., Go, K., Xia, F., Stark, M., Schaal, S., Hausman, K.: Genchip: Generating robot policy code for high-precision and contact-rich manipulation tasks. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 9596–9603. IEEE (2024)
4. Daneshmand, M., Noroozi, F., Corneanu, C., Mafakheri, F., Fiorini, P.: Industry 4.0 and prospects of circular economy: a survey of robotic assembly and disassembly. The International Journal of Advanced Manufacturing Technology **124**(9), 2973–3000 (2023)
5. Driess, D., Xia, F., Sajjadi, M.S., Lynch, C., Chowdhery, A., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., et al.: Palm-e: An embodied multimodal language model (2023)
6. Forlini, M., Babcinschi, M., Palmieri, G., Neto, P.: D-rmgpt: Robot-assisted collaborative tasks driven by large multimodal models. arXiv preprint arXiv:2408.11761 (2024)
7. Ghasemipour, S.K.S., Kataoka, S., David, B., Freeman, D., Gu, S.S., Mordatch, I.: Blocks assemble! learning to assemble with large-scale structured reinforcement learning. In: International Conference on Machine Learning. pp. 7435–7469. PMLR (2022)
8. Goldberg, A., Kondap, K., Qiu, T., Ma, Z., Fu, L., Kerr, J., Huang, H., Chen, K., Fang, K., Goldberg, K.: Blox-net: Generative design-for-robot-assembly using vlm supervision, physics simulation, and a robot with reset. arXiv preprint arXiv:2409.17126 (2024)
9. Guo, D., Xiang, Y., Zhao, S., Zhu, X., Tomizuka, M., Ding, M., Zhan, W.: Phygrasp: Generalizing robotic grasping with physics-informed large multimodal models. arXiv preprint arXiv:2402.16836 (2024)
10. Heo, M., Lee, Y., Lee, D., Lim, J.J.: Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. The International Journal of Robotics Research p. 02783649241304789 (2023)
11. Hu, M., Mu, Y., Yu, X., Ding, M., Wu, S., Shao, W., Chen, Q., Wang, B., Qiao, Y., Luo, P.: Tree-planner: Efficient close-loop task planning with large language models (2023)
12. Hu, W., Wang, C., Liu, F., Peng, X., Sun, P., Tan, J.: A grasps-generation-and-selection convolutional neural network for a digital twin of intelligent robotic grasping. Robotics and Computer-Integrated Manufacturing **77**, 102371 (2022)
13. Hu, Y., Xie, Q., Jain, V., Francis, J., Patrikar, J., Keetha, N., Kim, S., Xie, Y., Zhang, T., Fang, H.S., et al.: Toward general-purpose robots via foundation models: A survey and meta-analysis. arXiv preprint arXiv:2312.08782 (2023)
14. Huang, P., Liu, R., Aggarwal, S., Liu, C., Li, J.: Apex-mr: Multi-robot asynchronous planning and execution for cooperative assembly (2025), `https://arxiv.org/abs/2503.15836`
15. Kardos, C., Kovács, A., Váncza, J.: Decomposition approach to optimal feature-based assembly planning. CIRP Annals **66**(1), 417–420 (2017)

16. Li, S., Yan, Z., Wang, Z., Gao, Y.: Vlm-msgraph: Vision language model-enabled multi-hierarchical scene graph for robotic assembly. Robotics and Computer-Integrated Manufacturing **94**, 102978 (2025)
17. Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., Ichter, B., Florence, P., Zeng, A.: Code as policies: Language model programs for embodied control. arXiv preprint arXiv:2209.07753 (2022)
18. Liu, Y., Eyzaguirre, C., Li, M., Khanna, S., Niebles, J.C., Ravi, V., Mishra, S., Liu, W., Wu, J.: Ikea manuals at work: 4d grounding of assembly instructions on internet videos. Advances in Neural Information Processing Systems **37**, 80536–80566 (2024)
19. Macaluso, A., Cote, N., Chitta, S.: Toward automated programming for robotic assembly using chatgpt. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 17687–17693. IEEE (2024)
20. Mu, Y., Chen, T., Chen, Z., Peng, S., Lan, Z., Gao, Z., Liang, Z., Yu, Q., Zou, Y., Xu, M., et al.: Robotwin: Dual-arm robot benchmark with generative digital twins. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 27649–27660 (2025)
21. Nair, M.H., Rai, M.C., Poozhiyil, M., Eckersley, S., Kay, S., Estremera, J.: Robotic technologies for in-orbit assembly of a large aperture space telescope: A review. Advances in Space Research **74**(10), 5118–5141 (2024)
22. Suárez-Ruiz, F., Pham, Q.C.: A framework for fine robotic assembly. In: 2016 IEEE international conference on robotics and automation (ICRA). pp. 421–426. IEEE (2016)
23. Tang, B., Akinola, I., Xu, J., Wen, B., Handa, A., Van Wyk, K., Fox, D., Sukhatme, G.S., Ramos, F., Narang, Y.: Automate: Specialist and generalist assembly policies over diverse geometries. arXiv preprint arXiv:2407.08028 **1**(2) (2024)
24. Tang, C., Huang, D., Ge, W., Liu, W., Zhang, H.: Graspgpt: Leveraging semantic knowledge from a large language model for task-oriented grasping. IEEE Robotics and Automation Letters **8**(11), 7551–7558 (2023)
25. Tian, Y., Willis, K.D., Al Omari, B., Luo, J., Ma, P., Li, Y., Javid, F., Gu, E., Jacob, J., Sueda, S., et al.: Asap: Automated sequence planning for complex robotic assembly with physical feasibility. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 4380–4386. IEEE (2024)
26. Tian, Y., Xu, J., Li, Y., Luo, J., Sueda, S., Li, H., Willis, K.D., Matusik, W.: Assemble them all: Physics-based planning for generalizable assembly by disassembly. ACM Transactions on Graphics (TOG) **41**(6), 1–11 (2022)
27. Tie, C., Sun, S., Zhu, J., Liu, Y., Guo, J., Hu, Y., Chen, H., Chen, J., Wu, R., Shao, L.: Manual2skill: Learning to read manuals and acquire robotic skills for furniture assembly using vision-language models. arXiv preprint arXiv:2502.10090 (2025)
28. Wan, Y., Zhou, K., Chen, J., Dong, H.: Scanet: Correcting lego assembly errors with self-correct assembly network. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5940–5947. IEEE (2024)
29. Wang, L., Ling, Y., Yuan, Z., Shridhar, M., Bao, C., Qin, Y., Wang, B., Xu, H., Wang, X.: Gensim: Generating robotic simulation tasks via large language models. arXiv preprint arXiv:2310.01361 (2023)
30. Wang, R., Zhang, Y., Mao, J., Zhang, R., Cheng, C.Y., Wu, J.: Ikea-manual: seeing shape assembly step by step. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. pp. 28428–28440 (2022)
31. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems **35**, 24824–24837 (2022)

32. Willis, K.D., Jayaraman, P.K., Chu, H., Tian, Y., Li, Y., Grandi, D., Sanghi, A., Tran, L., Lambourne, J.G., Solar-Lezama, A., et al.: Joinable: Learning bottom-up assembly of parametric cad joints. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15849–15860 (2022)
33. Xu, J., Jin, S., Lei, Y., Zhang, Y., Zhang, L.: Reasoning tuning grasp: Adapting multi-modal large language models for robotic grasping. In: 2nd Workshop on Language and Robot Learning: Language as Grounding (2023)
34. Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., Narasimhan, K.: Tree of thoughts: Deliberate problem solving with large language models. Advances in neural information processing systems **36**, 11809–11822 (2023)
35. Yin, Y., Zheng, P., Li, C., Wan, K.: Enhancing human-guided robotic assembly: Ar-assisted dt for skill-based and low-code programming. Journal of Manufacturing Systems **74**, 676–689 (2024)
36. Zhang, D., Li, C., Zhang, R., Xie, S., Xue, W., Xie, X., Zhang, S.: Fm-ov3d: Foundation model-based cross-modal knowledge blending for open-vocabulary 3d detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 16723–16731 (2024)
37. Zhang, J., Zhang, W., Song, R., Ma, L., Li, Y.: Grasp for stacking via deep reinforcement learning. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 2543–2549. IEEE (2020)
38. Zhang, Z., Wang, Y., Zhang, Z., Wang, L., Huang, H., Cao, Q.: A residual reinforcement learning method for robotic assembly using visual and force information. Journal of Manufacturing Systems **72**, 245–262 (2024)
39. Zhao, Z., Lee, W.S., Hsu, D.: Large language models as commonsense knowledge for large-scale task planning. Advances in neural information processing systems **36**, 31967–31987 (2023)
40. Zheng, Y., Yao, L., Su, Y., Zhang, Y., Wang, Y., Zhao, S., Zhang, Y., Chau, L.P.: A survey of embodied learning for object-centric robotic manipulation. Machine Intelligence Research pp. 1–39 (2025)
41. Zhou, Q., Gu, Y., Li, J., Feng, B., Li, B., Bi, Y.: Towards zero-shot robot tool manipulation in industrial context: A modular vlm framework enhanced by multimodal affordance representation. Robotics and Computer-Integrated Manufacturing **98**, 103161 (2026)
42. Zhu, X., Jha, D.K., Romeres, D., Sun, L., Tomizuka, M., Cherian, A.: Multi-level reasoning for robotic assembly: From sequence inference to contact selection. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 816–823. IEEE (2024)
43. Zhu, Z., Hu, H.: Robot learning from demonstration in robotic assembly: A survey. Robotics **7**(2), 17 (2018)