AGE: Agentic Gaussian Editing in 3D Scenarios

Anonymous ECCV 2026 Submission

Paper ID #9447

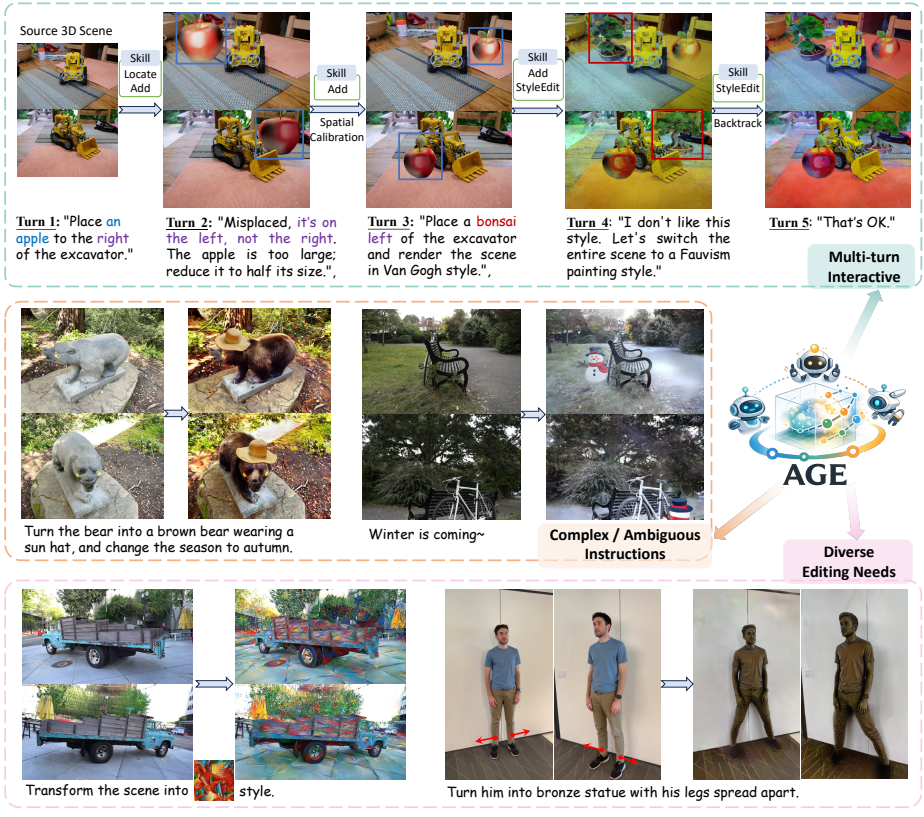


Fig. 1: Capabilities of AGE. Our framework enables a wide range of interactive 3D Gaussian editing tasks. Top: Multi-turn interaction where the system handles object insertion and global style transfer while supporting backtracking to correct errors. Middle: Execution of complex and ambiguous instructions. Bottom: Diverse editing needs including image-based style transfer and geometry deformation. AGE maintains long-term coherence and geometric consistency across all these scenarios.

Abstract. 3D Gaussian editing has recently attracted increasing attention with the rapid adoption of 3D Gaussian Splatting (3DGS). However, most existing approaches assume static user intent and rely on fixed execution pipelines. By treating editing as a one-shot operation, these methods are fundamentally limited due to the lack of interactive feedback

loops, insufficient 3D spatial reasoning, and poor generalization across diverse tasks. We present **AGE**, a multi-agent framework for continual and flexible 3D Gaussian editing that *reformulates the task as a long-horizon decision-making process*. Instead of executing edits in a single pass, AGE decomposes complex user goals through a dedicated planning agent and progressively applies modular editing skills. To enable reliable multi-turn interaction, we introduce a structured memory mechanism that records editing histories, perceptual feedback, and prior decisions, allowing the system to maintain long-term coherence, prevent destructive modifications, and support backtracking. We further integrate 3D perceptual analysis into both the planning and reflection agents, explicitly grounding decisions in geometric and structural cues rather than relying solely on 2D understanding. Extensive experiments demonstrate that AGE achieves robust, controllable, and coherent editing across diverse and challenging tasks, significantly outperforming existing pipeline-based approaches.

Keywords: 3D Gaussian Editing · Multi-agent System · Structured Memory

1 Introduction

The rapid advancement of neural scene representations has fundamentally reshaped the landscape of 3D content creation [2,4,12,34,35]. Among recent breakthroughs, 3D Gaussian Splatting (3DGS) has emerged as a highly efficient and expressive representation for real-time rendering and reconstruction [19]. Beyond reconstruction, its explicit and editable representation naturally facilitates direct manipulation in Gaussian space, giving rise to the task of 3D Gaussian editing—modifying geometry, appearance, and semantics within a reconstructed scene [3, 14, 45]. This capability is increasingly critical for immersive content creation, virtual reality, and digital asset production [10, 47, 52, 55, 56].

Despite promising progress, existing 3D Gaussian editing methods remain fundamentally limited in their formulation. Most approaches adopt pipeline-based paradigms that treat editing as a one-shot transformation: a user instruction is mapped to a predefined sequence of operations, producing a final result in a single pass [8, 11, 49, 53]. While such designs can yield visually plausible outputs under constrained settings, they implicitly assume that user intent is fully specified and conforms to predefined task templates [16, 62]. However, real-world editing is inherently complex and interactive. User instructions are often ambiguous, evolving, or multi-stage in nature [30,59,61]. Furthermore, current systems typically lack explicit 3D spatial reasoning, limiting their ability to handle multi-object coordination, geometry-aware transformations, or spatially consistent modifications. These shortcomings become particularly pronounced in long-horizon editing scenarios, where multiple sequential operations must remain coherent over time.

To address these challenges, we propose AGE, a multi-agent framework that reconceptualizes 3D Gaussian editing as an adaptive, long-horizon, interactive

decision-making process rather than a static execution pipeline. Inspired by collaborative human workflows [9, 21, 27, 38], AGE decomposes complex editing objectives into coordinated subtasks managed by specialized agents. A dedicated planning agent interprets user intent and generates structured, step-by-step strategies. Subsequently, the skill executor agent performs atomic Gaussian editing operations, enabling compositionality and reuse across diverse tasks. A reflector agent evaluates intermediate outcomes and revises future plans in collaboration with the backtracker agent when inconsistencies or undesired effects are detected. Together, these agents establish a closed-loop 3D Gaussian editing paradigm.

Central to our framework is a structured memory mechanism designed for continual editing. Instead of processing each instruction independently, AGE maintains an explicit record of editing history, perceptual observations, and prior decisions. This stateful design enables coherent multi-turn interaction and supports backtracking when necessary. By modeling editing as a sequential decision process over a persistent scene state, the system achieves long-term consistency across complex transformation chains. Crucially, AGE incorporates explicit 3D perceptual grounding into both planning and reflection stages. Agents reason directly over geometric attributes such as spatial distributions, depth relationships, and object scales. This geometry-aware reasoning significantly enhances robustness in tasks involving spatial rearrangement, object-level manipulation, and coordinated multi-object editing. As a result, AGE provides a controllable and adaptive editing framework capable of dynamically responding to user feedback while preserving global scene consistency.

Existing benchmarks for 3D Gaussian editing are limited in scope, primarily focusing on simple, single-step instructions. To better assess real-world applicability, we construct a more comprehensive benchmark that includes not only standard editing tasks, but also complex multi-step instructions, ambiguous commands, and multi-turn interactive scenarios. Extensive experiments demonstrate that AGE consistently outperforms existing pipeline-based methods in terms of robustness, controllability, and long-term coherence.

Our main contributions are summarized as follows:

- We extend 3D Gaussian editing to an adaptive, long-horizon interactive decision-making problem, moving beyond static pipeline-based execution.
- We introduce AGE, a spatially coordinated multi-agent framework with the closed-loop planning–execution–reflection–backtracking mechanism and modular editing skills.
- We propose a structured memory design with explicit 3D-grounded perceptual reasoning, enabling coherent multi-turn interaction and improved geometric awareness.
- We construct a comprehensive benchmark for 3D Gaussian editing covering complex, ambiguous, and multi-turn instructions, and show significant improvements over prior methods in extensive experiments.

2 Related Work

2.1 3D Gaussian Editing

3D Gaussian Splatting (3DGS) [19] enables efficient real-time neural rendering with explicit scene primitives. By directly representing geometry and appearance using structured 3D Gaussians, it supports intuitive 3D scene manipulation [39]. Existing 3D Gaussian editing approaches predominantly follow pipeline-based paradigms [26, 40]. Given a user instruction, these systems map it to a predefined sequence of operations, such as region selection, attribute modification, geometric transformation, or optimization-based refinement, and produce the edited result in a fixed execution pass [5, 33, 47, 49, 50]. Several methods further incorporate 2D foundation models for semantic localization or text-driven guidance and project image-space supervision back into the 3D Gaussian representation [17,29,48,58]. While these approaches achieve promising results for localized and appearance-level edits, the inherent complexity of 3D editing, coupled with the limited capabilities of foundation models, restricts existing methods to a fixed set of editing types. Moreover, the editing process typically lacks flexibility and adaptability once the execution pipeline is determined. Recent studies [8] have started to incorporate intelligent decision-making paradigms into 3D Gaussian editing. However, the lack of memory systems and feedback mechanisms, along with rigid workflows, still results in limited flexibility in current editing systems. In contrast to treating editing as a one-step transformation, we reformulate 3D Gaussian editing as a long-horizon decision-making process over a persistent scene state. This perspective enables continual interaction, structured reasoning, and dynamic adjustment throughout the editing process.

2.2 Multi-agent System

Recent advances in large language models (LLMs) have facilitated the emergence of agentic systems capable of structured reasoning and tool use [18,32,41,51,57]. Early single-agent frameworks address complex tasks through chain-of-thought reasoning [51], while subsequent work introduced planner–executor architectures to improve modularity and robustness in long-horizon scenarios [28, 42]. Multi-agent systems further enhance reliability and scalability by assigning specialized roles such as planning, execution, and reflection, while enabling collaborative decision-making [25,38,54]. Memory-augmented agents maintain contextual state across interactions [24, 36, 37], while reflection mechanisms revise plans based on intermediate outcomes, significantly improving performance in iterative and sequential tasks [31, 43]. Despite rapid progress in agentic frameworks, their application to 3D scene editing remains largely underexplored. Even when LLMs are incorporated for instruction parsing and region selection [8, 47], they are often limited to front-end translation modules rather than functioning as components within a flexible architecture. Our work bridges this gap by embedding a spatially grounded multi-agent framework directly into the 3D Gaussian editing process.

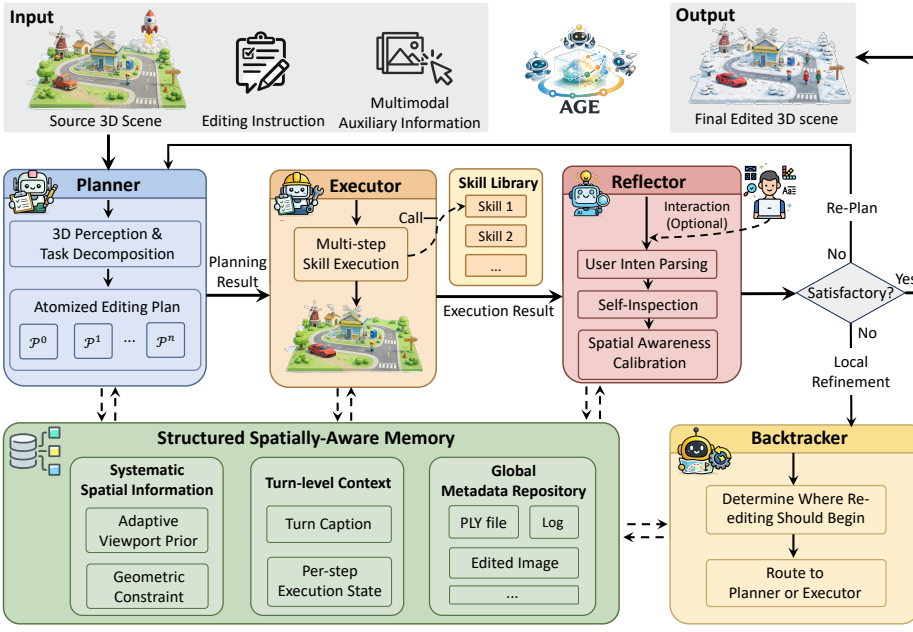


Fig. 2: Overview of the AGE framework. The system reformulates 3D Gaussian editing as a long-horizon sequential decision-making process. Given a source 3D scene and user instructions, the Planner decomposes the goal into a structured plan. The Executor then invokes specific skills from the library. Post-execution, the Reflector evaluates the results against user intent, triggering either a "self-check" completion or a re-planning request. If errors occur, the Backtracker identifies the optimal re-entry point. All agents are coordinated by a Structured Spatially-Aware Memory that maintains systematic spatial information, turn-level context, and global metadata to ensure reversible and coherent edits.

3 Method

3.1 Problem Formulation

Traditional 3D Gaussian editing approaches typically adopt a one-shot pipeline paradigm. Given an initial scene $\mathcal{G}$ and a user instruction I, these methods aim to find a mapping function f such that:

$$\mathcal{G}' = f(\mathcal{G}, I), \tag{1}$$

where f usually represents a fixed, predefined sequence of operations, such as first performing 2D edits and then projecting them into 3D, or using 3D diffusion to invert the target object and then re-generate it under the constraints of the editing instruction. However, this formulation assumes that user intent is static and fully specified, which fails to capture the inherently ambiguous and iterative nature of real-world editing.

To address these limitations, we reformulate 3D Gaussian editing as a long-horizon sequential decision-making problem over a persistent scene state. Instead of a single transformation, we define the process as a sequence of interactions over T discrete steps. At each step $t \in \{0, 1, ..., T-1\}$, the system takes the current scene state $\mathcal{G}_t$, a user instruction I_t, and a structured memory $\mathcal{M}_t$ as input to produce the next state:

$$\mathcal{G}_{t+1}, \mathcal{M}_{t+1} = \mathbf{F}(\mathcal{G}_t, I_t, \mathcal{M}_t), \tag{2}$$

where $\mathbf{F}$ denotes the AGE framework and $\mathcal{M}$ represents the stateful memory bank that evolves throughout the editing process. Under this formulation, the objective consists of three dimensions:

- **Instruction Following:** The final scene $\mathcal{G}_T$ must precisely satisfy the cumulative intent expressed in the sequence of instructions $\{I_t\}_{t=1}^{T}$.
- **Visual Fidelity:** The edited 3D Gaussians must maintain high-quality rendering performance and aesthetic appeal.
- **Long-term Coherence:** Modifications must remain consistent across multiple turns, preventing misunderstandings of the current instructions caused by forgetting the earlier editing process.

Notably, the traditional pipeline-based approach can be viewed as a degenerate case of our framework where $T = 1$ and no memory or reflection mechanisms are maintained. By treating the task as a sequential decision process, AGE enables interactive feedback loops and complex, multi-stage transformations that are beyond the reach of static pipelines.

3.2 Overview of AGE

Building upon the sequential decision-making formulation introduced in Sec. 3.1, we present AGE, a spatially-coordinated multi-agent framework tailored for long-horizon and interactive 3D Gaussian editing. As illustrated in Fig. 2, the system orchestrates four specialized agents, namely the Planner (**P**), Executor (**E**), Reflector (**R**), and Backtracker (**B**), to iteratively refine the scene state through a collaborative paradigm of planning, execution, reflection, and state revision.

Formally, given the current scene $\mathcal{G}_t$, user instruction I_t, and the structured memory $\mathcal{M}_t$ at step t, the framework operates through the following coordinated sequence:

$$\mathcal{P}_t, \mathcal{M}_t^p = \mathbf{P}(\mathcal{G}_t, I_t, \mathcal{M}_t), \quad \text{s.t. } \mathcal{P}_t = \{(s_t^0, I_t^0), (s_t^1, I_t^1), \ldots, (s_t^n, I_t^n)\}, \tag{3}$$

$$\mathcal{G}_{t+1}, \mathcal{M}_t^{pe} = \prod_{i=0}^{n} \mathbf{E}(\mathcal{G}_t^i, s_t^i, I_t^i, \mathcal{M}_t^p), \quad \text{s.t. } \mathcal{G}_t^0 = \mathcal{G}_t, \tag{4}$$

$$\mathcal{R}_{t+1}, \mathcal{M}_t^{per} = \mathbf{R}(\mathcal{G}_{t+1}, I_{t+1}, \mathcal{M}_t^{pe}), \tag{5}$$

$$\mathcal{B}_{t+1}, \mathcal{M}_{t+1} = \mathbf{B}(\mathcal{G}_{t+1}, I_{t+1}, \mathcal{M}_t^{per}), \tag{6}$$

where s_t^i and I_t^i denote the skill to be invoked and the corresponding atomic instruction in the i-th step of the editing plan $\mathcal{P}_t$, respectively. $\mathcal{R}_{t+1}$ and $\mathcal{B}_{t+1}$ represent the results inferred by the Reflector and the Backtracker based on the context, which can be used to update the memory repository and further guide subsequent editing processes. The system initializes with $\mathcal{M}_0 = \varnothing$.

A key innovation of our framework is the stateful memory mechanism $\mathcal{M}$, which is not a passive data store but a dynamic repository updated by each agent during their respective operations. This allows agents to adaptively extract historical context and spatial priors to inform their decisions. Furthermore, the framework supports an autonomous self-check mode, where the instruction I_{t+1} is internally set to verification prompts (e.g., "Verify if the editing is complete") to ensure high-fidelity convergence.

3.3 Spatially-Coordinated Multi-Agents

Unlike prior methods that treat language parsing as a simplistic front-end module, AGE embeds 3D spatial reasoning directly into each agent's decision-making process. By leveraging chain-of-thought reasoning with Multimodal Large Language Models (MLLMs), our framework ensures that every action is grounded in the geometric and semantic context of the 3D scene.

Planning Agent: Hierarchical Task Decomposition The Planner **P** serves as the strategic core of the framework, responsible for decomposing high-level, often ambiguous user instructions I_t into a structured sequence of atomic editing goals. This process follows a two-stage "perceive-then-plan" paradigm. (i) Adaptive Perceptual Selection: To mitigate the interference of irrelevant background information, the Planner first performs dense perception across the scene. It adaptively selects K views that are most relevant to the current instruction, ensuring that subsequent reasoning is focused on the critical 3D regions. (ii) Editing Strategy Generation: The Planner retrieves descriptions from the Skill Library (as shown in Fig. 3) and synthesizes a step-by-step plan $\mathcal{P}_t$. To maintain planning accuracy, the Planner solely retrieves the description of each skill. If the current turn is triggered by a failure identified by the Reflector or Backtracker, the Planner explicitly incorporates the "reflection thoughts" from the memory $\mathcal{M}$ to refine its strategy and avoid prior errors.

Skill Executor Agent: Atomic Skill Invocation The Executor **E** is tasked with the sequential implementation of the plan $\mathcal{P}_t$. For each step i, it functions as a precision-oriented operator that maps atomic instructions I_t^i to specific parameterizations of the selected skill s_t^i.

To maintain spatial consistency, the Executor does not operate in isolation; it continuously queries the Systematic Spatial Information stored in memory $\mathcal{M}$. By considering historical coordinate transformations and summarized spatial constraints (e.g., relative object scales and orientations), the Executor ensures that new Gaussians or modifications are seamlessly integrated into the existing 3D structure.

Reflection and Backtracking: The Closed-loop Feedback To ensure high-fidelity convergence and long-term coherence, we introduce a collaborative mechanism between the Reflector **R** and the Backtracker **B**.

The Reflector serves as a supervisory module that evaluates the delta between the current scene state and the user's objectives to determine the downstream workflow. It first assesses whether the editing process has reached convergence; termination is triggered either by explicit user approval or when an internal MLLM-based verification confirms that all instructions have been successfully fulfilled. If the current results exhibit discrepancies identified through negative user feedback or internal self-checks, the Reflector must decide the optimal correction strategy. Specifically, it adjudicates whether the failure warrants a global strategic re-planning by the Planner or a targeted re-execution of specific, erroneous steps within the previous turn's skill invocation sequence.

When re-execution is deemed necessary, the Backtracker determines the optimal entry point for modification. It generates concrete corrective strategies (e.g., "re-locate the object 0.2 units lower") and injects these as backtrack hints into the memory. This feedback loop allows the Executor to improve its skill invocation in the subsequent attempt, effectively correcting spatial errors or semantic misalignments.

3.4 Structured Spatially-Aware Memory

To maintain long-term coherence and facilitate precise 3D reasoning across multiple interaction turns, we design a Structured Spatially-Aware Memory ($\mathcal{M}$). Unlike passive data stores, $\mathcal{M}$ serves as a dynamic state repository that encodes geometric priors, execution histories, and perceptual feedback. As illustrated in Fig. 2, the memory bank is organized into three hierarchical components:

- **Systematic Spatial Information:** This part maintains the system's dynamic 3D understanding of the global environment. It encapsulates: (i) Adaptive Viewport Priors: Results from the Planner's adaptive perceptual selection, ensuring that future reasoning steps are grounded in the most relevant visual contexts. (ii) Geometric Constraints: 3D spatial boundaries, relative object scales, and orientation priors synthesized by the Reflector. These spatial priors are continuously queried by the Executor to ensure that newly generated or modified Gaussians are seamlessly integrated into the existing scene structure, preventing geometric misalignments.
- **Turn-level Context:** To support complex multi-turn interactions and error recovery, we store interaction data at two granularities. (i) Turn Captions: High-level summaries of each interaction turn, including the user's cumulative intent and the turn number. These captions serve as "anchor points" for the Backtracker when a strategic retreat to a much earlier state is required. (ii) Per-step Execution States: Fine-grained logs of atomic skill invocations within a single turn. This allows the system to perform "surgical" re-editing by re-executing only specific erroneous steps identified by the Reflector, without discarding the progress of the entire turn.

– **Global Metadata Repository:** This repository functions as a persistent archive of the 3D scene's metadata across the editing trajectory. By storing indexed links to the 3D Gaussian parameters and scene states at each step i, the system can render intermediate results to provide multimodal feedback for the agents' internal reasoning or allowing the user to observe the evolution of the 3D scene in real-time.

By modeling the editing process as a sequential decision-making task over this stateful memory $\mathcal{M}$, AGE effectively bypasses the limitations of one-shot pipelines, achieving robust and reversible 3D manipulations.

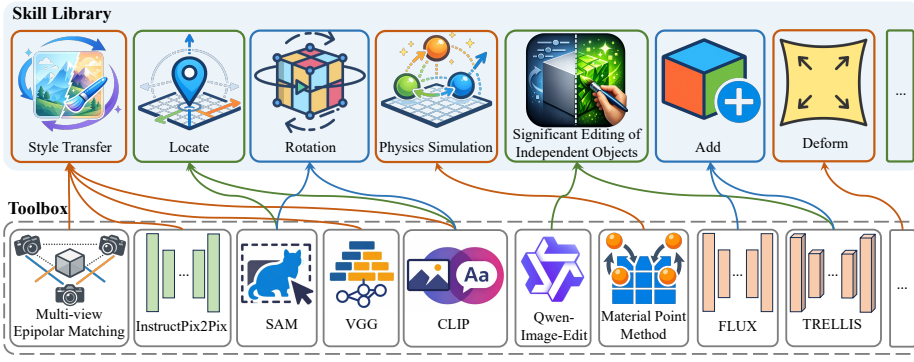
3.5 Skill Library: Modular Action Space

The Skill Library ($\mathcal{S}$) defines the functional boundary of AGE, serving as the interface between high-level agent reasoning and low-level 3D Gaussian manipulations. Rather than employing a monolithic editing model, we decompose the action space into a set of specialized, reusable skills, each backed by a collection of atomic tools from our underlying Toolbox. As illustrated in Fig. 3, this design enables complex cross-tool composition to fulfill diverse editing needs.

To ensure that the multi-agent system can precisely invoke and correct these actions, each skill $s \in \mathcal{S}$ is defined by a rigorous schema containing the following key attributes:

– **Functional Metadata** (`name`, `description`): These provide the semantic identity of the skill, allowing the Planner to match user intent with available capabilities. For instance, the 'Add' skill is described as a generator that inserts new 3D assets into a specified context.
– **Execution Interface** (`args_schema`, `runner`): The `args_schema` defines the required input parameters (e.g., 3D coordinates, scale factors, or text prompts) in a structured format. The `runner` serves as the concrete execution handle, invoking the corresponding software pipeline to modify the 3D Gaussian scene state.
– **Reasoning Guidance** (`arg_hints`, `arg_examples`): To improve the zero-shot planning accuracy of MLLMs, we provide explicit constraints and logic examples. These hints prevent common failures, such as generating objects with insufficient 3D priors or using incorrect coordinate frames.
– **Self-Correction Support** (`backtrack_hint_guidance`): This is a critical innovation of our framework that enables the closed-loop feedback mechanism. It provides the Backtracker with a specialized knowledge base on how to adjust parameters if the initial execution fails (e.g., providing specific logic for shifting objects along the z-axis to correct floating artifacts).

By decoupling the high-level planning from the specific implementation of editing tools, the Skill Library ensures that AGE is inherently extensible. New editing capabilities (e.g., physics simulation or advanced material editing) can be integrated by simply adding a new skill entry to the repository.



Skill Sample (Add)

```
name="add",
description="Generate a new object/asset according to prompt and insert it into the scene. (If the insertion position is
    determined by a reference object, the reference object must be located first.)",
args_schema=AddArgs,
runner=add_skill_runner,
arg_hints=("Constraints:"
    "-generate_prompt: A prompt abstracted from sub-instruction and BACKTRACK_HINT to generate a reference image."
    "-placement_position: A 3D coordinate list [x, y, z] representing the center of the object."
    "-scale: A floating-point number representing the uniform scaling factor of the object. Defaults to 1.0."
  "Guidance:"
    "- For optimal processing by the 3D generator, the subject within the reference image must exhibit a strong sense of
    three-dimensionality and  be placed against a clean background."
    "- When explicit coordinates or scale are present in the sub-instruction or HINT, extract them exactly as written; do not
    infer or approximate."
    "- Otherwise, ..."),
arg_examples=[{"generate_prompt": "A cute sitting Golden Retriever, 3D style, clean background.", "placement_position":
    [-0.8, 0.0, -0.5], "scale": 1.5},],
backtrack_hint_guidance={"guidance": (
    "- `placement_position` represents the [x, y, z] coordinates for the object's placement."
    "- The z-axis is perpendicular to the ground; to place an object 'higher', increase the z-coordinate."
    "- ..."),
  "hint_examples": (
    "- The generated coffee cup looks too small on the table. The original scale was 1.0; a suitable scale for re-editing is 1.3.
    As the user made no requests to modify the object's position, the original placement position provided in
    ARTIFACTS_SNAPSHOT is retained.")}
```

Fig. 3: Skill Library and Modular Action Space. AGE decouples high-level reasoning from low-level execution through a modular Skill Library. Each skill is an interface that maps to specific tools in the toolbox. A skill definition includes Functional Metadata for planning, an Execution Interface for parameterization, Reasoning Guidance to provide MLLMs with few-shot examples, and Self-Correction Support to guide the Backtracker in adjusting parameters after failed attempts. This design ensures the framework is extensible and robust to complex instruction parameterization.

4 Experiments

4.1 Experimental Setup

Implementation Details. The proposed AGE framework relies on a multi-agent system driven by Qwen3-VL-30B [1] as the core cognitive engine for the Planner and Reflector, and DeepSeek-R1-32B [13] as the cognitive engine for the Executor and Backtracker. To ensure consistent and stable reasoning, the

MLLMs are deployed via vLLM [22] with the generation temperature set to 0.02. The orchestration and communication between these agents are implemented using LangGraph [46]. Our Skill Library integrates a suite of state-of-the-art foundation models and vision tools, such as SAM [20] for precise segmentation, FLUX [23] for asset generation, and InstructPix2Pix [3] for style transfer. K is set to 8, and all execution phases are conducted on a single NVIDIA A100 GPU.

Dataset. We construct a diverse benchmark comprising 57 distinct editing scenarios, categorized into four subsets. (i) **Common Instructions** (20 scenarios): Standard tasks commonly employed in other 3D Gaussian editing literature. (ii) **Complex/Ambiguous Instructions** (20 scenarios): Scenarios with abstract or high-level semantics that require advanced spatial reasoning and multi-step task decomposition. (iii) **Multimodal Auxiliary Inputs** (10 scenarios): Tasks where instructions are augmented with multimodal guidance, such as reference images or click-and-drag information. (iv) **Multi-turn Interactions** (7 scenarios): Prolonged editing sequences featuring continuous user feedback, requiring the system to maintain long-term coherence and actively support backtracking. More details are provided in the Appendix.

Metrics. Following [15] and [6], we compute the CLIP Text-Image Direction Similarity (C_{sim}) and CLIP Direction Consistency (C_{con}). Additionally, we conduct the Image Aesthetics Assessment (IAA) using [60] to further evaluate the quality. Since traditional 2D metrics often struggle to capture fine-grained structural preservation and complex instruction adherence in 3D spaces, we introduce an automated evaluation protocol using `Gemini 3.1 Pro` [44] as an LLM-based judge. The model scores the rendered multi-view results on a scale of 1 to 10 across three key dimensions: (i) structural consistency of non-edited regions, (ii) completion degree of the editing instructions, (iii) overall aesthetic quality.

4.2 Comparison with Baselines

Qualitative Evaluation. We compare AGE against two representative 3D Gaussian editing methods: GaussianEditor [7] and DGE [6]. As shown in Fig. 4, our method demonstrates clear advantages in executing complex instructions and preserving spatial coherence. For instance, when tasked with replacing a bonsai tree or altering the season alongside object modifications, pipeline-based baselines often struggle to fully execute the commands, resulting in severe blurring or structural degradation of the background. In contrast, AGE successfully fulfills each aspect of the instruction while maintaining sharp geometric boundaries and the structural integrity of unedited regions.

Furthermore, Fig. 5 demonstrates AGE's exceptional versatility beyond standard editing paradigms. Driven by the modular Skill Library, the framework effortlessly adapts to highly diverse editing needs. This includes physics-aware geometry deformation and intricate localized material editing. These qualitative results clearly underscore our framework's capacity to achieve robust, controllable, and coherent editing across highly challenging tasks.

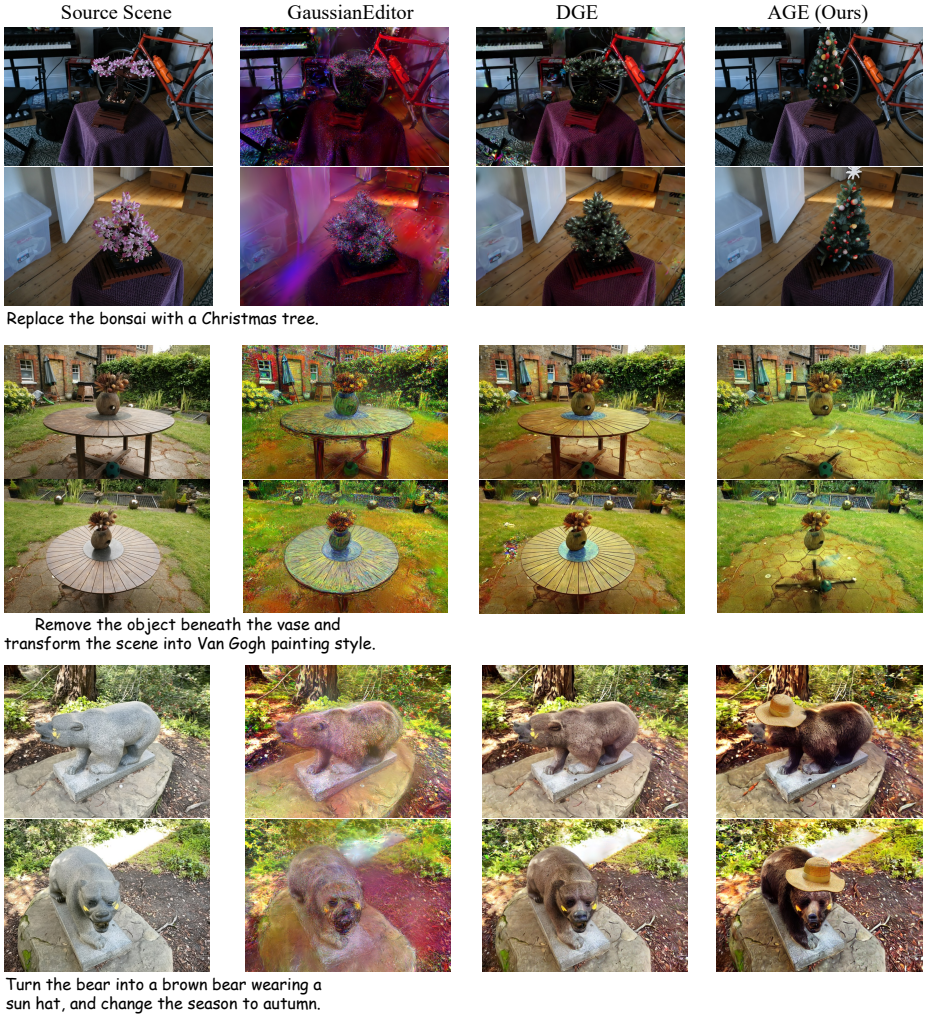


Fig. 4: Qualitative comparison with representative 3D Gaussian editing methods.

Quantitative Evaluation. As shown in Table 1, AGE demonstrates superior performance across all evaluated dimensions. Specifically, our framework achieves a C_{sim} score of 0.192, significantly outperforming GaussianEditor and DGE. This metric highlights our system's enhanced capability to accurately interpret and align with complex user instructions. Furthermore, our higher scores in C_{con} and IAA confirm that our method not only achieves better semantic alignment but also maintains higher multi-view consistency and visual fidelity.

The automated LLM-as-judge evaluation results, shown in Table 2, further corroborate these findings. AGE consistently excels in three critical dimensions: structural consistency of non-edited regions, editing instruction completion, and overall aesthetic quality. By reformulating editing as a long-horizon decision-

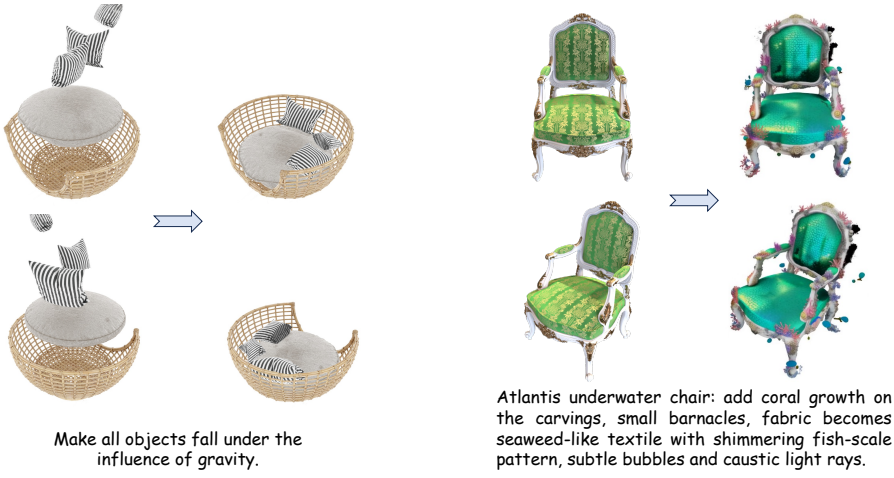


Fig. 5: Diverse editing capabilities enabled by the modular skill library.

Table 1: Comparison with other editing methods. AGE achieves the best performance.

Method	C_{sim}	C_{con}	IAA
GaussianEditor [7]	0.104	0.863	4.51
DGE [6]	0.133	0.892	5.79
AGE (Ours)	**0.192**	**0.907**	**6.65**

making process equipped with a structured memory mechanism, AGE effectively prevents the destructive modifications common in one-shot pipeline execution.

4.3 Discussion and Prospects

Multi-agent Collaboration in Interactive Editing As illustrated in Fig. 6, AGE performs editing through the collaboration of multiple specialized agents within a flexible and closed-loop workflow. Based on this, the system progressively refines the scene while maintaining spatial consistency and coherence across multiple interaction turns.

Synergy Between Reasoning and Execution We observe a decoupling between cognitive planning and actual execution during the experiment. The multi-agent system demonstrates strong capability in processing instructions and maintaining global scene coherence. However, when faced with highly demanding "atomic instructions", the performance bottleneck often arises not from the Planner's logic, but from the inherent limitations of the low-level tools in the Toolbox. For example, while the Backtracker can provide correct feedback, the execution tool may still struggle to generate a 3D Gaussian asset that perfectly matches the feedback information.

Table 2: The LLM-as-judge evaluation results.

Method	Structural Consistency	Completion Degree	Aesthetic Quality
GaussianEditor [7]	5.24	5.70	6.58
DGE [6]	7.39	5.23	8.03
AGE (Ours)	**8.90**	**7.17**	**8.52**

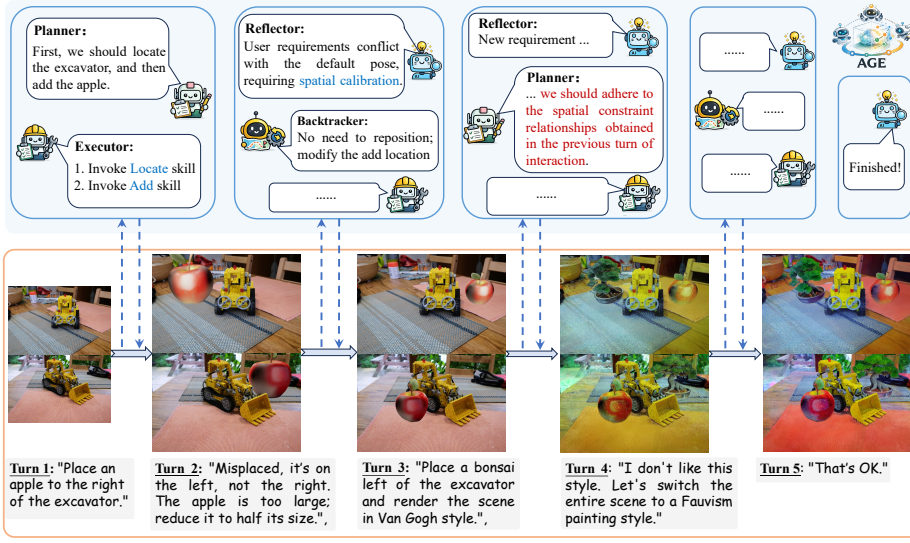


Fig. 6: Multi-agent collaborative workflow for interactive 3D Gaussian editing.

Future Outlook The architectural design of AGE makes it easy to update or expand the skill repository as more robust 3D foundation models emerge. Future work will focus on: (i) Skill Refinement: Developing more specialized 3D tools to reduce the "execution bottleneck". (ii) End-to-End Learning: Exploring ways to partially fine-tune the agents to better understand the nuances of 3D Gaussian geometry directly, rather than relying solely on 2D visual proxies.

5 Conclusion

In this paper, we present AGE, an agentic framework for 3D Gaussian editing that formulates editing as a long-horizon sequential decision-making process. AGE integrates a spatially-coordinated multi-agent system, including a Planner, Executor, Reflector, and Backtracker, together with a structured spatially-aware memory that maintains editing history and geometric context across turns. This design enables adaptive planning, iterative refinement, and coherent multi-turn interaction in complex 3D editing scenarios. Extensive experiments demonstrate that AGE achieves more robust instruction following, better structural preservation, and higher visual quality than existing pipeline-based methods.

References

1. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5470–5479 (2022)
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18392–18402 (2023)
4. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: European conference on computer vision. pp. 333–350. Springer (2022)
5. Chen, J.K., Wang, Y.X.: Proedit: Simple progression is all you need for high-quality 3d scene editing. arXiv preprint arXiv:2411.05006 (2024)
6. Chen, M., Laina, I., Vedaldi, A.: DGE: Direct Gaussian 3D Editing by Consistent Multi-view Editing (2024)
7. Chen, Y., Chen, Z., Zhang, C., Wang, F., Yang, X., Wang, Y., Cai, Z., Yang, L., Liu, H., Lin, G.: GaussianEditor: Swift and Controllable 3D Editing with Gaussian Splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 21476–21485 (Jun 2024)
8. Chi, Y., Li, X., Huang, Z., Rehg, J.M.: Vinedresser3d: Agentic text-guided 3d editing. arXiv preprint arXiv:2602.19542 (2026)
9. Ding, W., Chen, J., Chen, M., Xie, F., Mao, Q., Dames, P.: Pfea: An llm-based high-level natural language planning and feedback embodied agent for human-centered ai. arXiv preprint arXiv:2510.24109 (2025)
10. Dong, J., Wang, Y.X.: 3dgs-drag: Dragging gaussians for intuitive point-based 3d editing. arXiv preprint arXiv:2601.07963 (2026)
11. Fang, J., Wang, J., Zhang, X., Xie, L., Tian, Q.: Gaussianeditor: Editing 3d gaussians delicately with text instructions. In: CVPR (2024)
12. Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5501–5510 (2022)
13. Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025)
14. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-nerf2nerf: Editing 3d scenes with instructions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19740–19750 (2023)
15. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-NeRF2NeRF: Editing 3D Scenes with Instructions (2023)
16. He, S., Ji, P., Yang, Y., Wang, C., Ji, J., Wang, Y., Ding, H.: A survey on 3d gaussian splatting applications: Segmentation, editing, and generation. arXiv preprint arXiv:2508.09977 (2025)
17. Huang, S.Y., Chou, Z.T., Wang, Y.C.F.: 3d gaussian inpainting with depth-guided cross-view consistency. In: Proceedings of the computer vision and pattern recognition conference. pp. 26704–26713 (2025)

18. Karpas, E., Abend, O., Belinkov, Y., Lenz, B., Lieber, O., Ratner, N., Shoham, Y., Bata, H., Levine, Y., Leyton-Brown, K., et al.: Mrkl systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. arXiv preprint arXiv:2205.00445 (2022)
19. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., et al.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
20. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4015–4026 (2023)
21. Kulkarni, M.: Agent-s: Llm agentic workflow to automate standard operating procedures. arXiv preprint arXiv:2503.15520 (2025)
22. Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the 29th symposium on operating systems principles. pp. 611–626 (2023)
23. Labs, B.F., Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., et al.: Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. arXiv preprint arXiv:2506.15742 (2025)
24. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in neural information processing systems **33**, 9459–9474 (2020)
25. Li, G., Hammoud, H., Itani, H., Khizbullin, D., Ghanem, B.: Camel: Communicative agents for" mind" exploration of large language model society. Advances in neural information processing systems **36**, 51991–52008 (2023)
26. Li, L., Huang, Z., Feng, H., Zhuang, G., Chen, R., Guo, C., Sheng, L.: Voxhammer: Training-free precise and coherent 3d editing in native 3d space (2025), `https://arxiv.org/abs/2508.19247`
27. Li, Z., Xu, S., Mei, K., Hua, W., Rama, B., Raheja, O., Wang, H., Zhu, H., Zhang, Y.: Autoflow: Automated workflow generation for large language model agents. arXiv preprint arXiv:2407.12821 (2024)
28. Liu, B., Jiang, Y., Zhang, X., Liu, Q., Zhang, S., Biswas, J., Stone, P.: Llm+ p: Empowering large language models with optimal planning proficiency, 2023. URL https://arxiv. org/abs/2304.11477 (2024)
29. Liu, Z., Ouyang, H., Wang, Q., Cheng, K.L., Xiao, J., Zhu, K., Xue, N., Liu, Y., Shen, Y., Cao, Y.: Infusion: Inpainting 3d gaussians via learning depth completion from diffusion prior. arXiv preprint arXiv:2404.11613 (2024)
30. Ma, S., Guo, Y., Su, J., Huang, Q., Zhou, Z., Wang, Y.: Talk2image: A multi-agent system for multi-turn image generation and editing. arXiv preprint arXiv:2508.06916 (2025)
31. Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., et al.: Self-refine: Iterative refinement with self-feedback. Advances in neural information processing systems **36**, 46534–46594 (2023)
32. Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., et al.: Augmented language models: a survey. arXiv preprint arXiv:2302.07842 (2023)

33. Mikaeili, A., Perel, O., Safaee, M., Cohen-Or, D., Mahdavi-Amiri, A.: Sked: Sketch-guided text-based 3d editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14607–14619 (2023)
34. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
35. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG) **41**(4), 1–15 (2022)
36. Packer, C., Fang, V., Patil, S., Lin, K., Wooders, S., Gonzalez, J.: Memgpt: towards llms as operating systems. (2023)
37. Park, J.S., O'Brien, J., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative agents: Interactive simulacra of human behavior. In: Proceedings of the 36th annual acm symposium on user interface software and technology. pp. 1–22 (2023)
38. Qian, C., Liu, W., Liu, H., Chen, N., Dang, Y., Li, J., Yang, C., Chen, W., Su, Y., Cong, X., et al.: Chatdev: Communicative agents for software development. In: Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers). pp. 15174–15186 (2024)
39. Qin, H., Sun, Y., Wang, M., Kong, M., Lu, M., Zhu, Q.: Variation-aware flexible 3d gaussian editing. arXiv preprint arXiv:2602.11638 (2026)
40. Qu, Y., Chen, D., Li, X., Li, X., Zhang, S., Cao, L., Ji, R.: Drag your gaussian: Effective drag-based editing with score distillation for 3d gaussian splatting. arXiv preprint arXiv:2501.18672 (2025)
41. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools. Advances in neural information processing systems **36**, 68539–68551 (2023)
42. Shen, Y., Song, K., Tan, X., Li, D., Lu, W., Zhuang, Y.: Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. Advances in Neural Information Processing Systems **36**, 38154–38180 (2023)
43. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S.: Reflexion: Language agents with verbal reinforcement learning. Advances in neural information processing systems **36**, 8634–8652 (2023)
44. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
45. Vachha, C., Haque, A.: Instruct-gs2gs: Editing 3d gaussian splats with instructions (2024). URL https://instruct-gs2gs. github. io **6**, 15 (2024)
46. Wang, J., Duan, Z.: Agent ai with langgraph: A modular framework for enhancing machine translation using large language models. arXiv preprint arXiv:2412.03801 (2024)
47. Wang, J., Fang, J., Zhang, X., Xie, L., Tian, Q.: Gaussianeditor: Editing 3d gaussians delicately with text instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20902–20911 (2024)
48. Wang, S., Zhang, S., Millerdurai, C., Westermann, R., Stricker, D., Pagani, A.: Inpaint360gs: Efficient object-aware 3d inpainting via gaussian splatting for 360deg scenes. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 117–127 (2026)
49. Wang, Y., Yi, X., Wu, Z., Zhao, N., Chen, L., Zhang, H.: View-Consistent 3D Editing with Gaussian Splatting (2024)

50. Wang, Y., Yi, X., Wu, Z., Zhao, N., Chen, L., Zhang, H.: View-consistent 3d editing with gaussian splatting. In: European Conference on Computer Vision. pp. 404–420. Springer (2024)
51. 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)
52. Wu, J., Bian, J.W., Li, X., Wang, G., Reid, I., Torr, P., Prisacariu, V.A.: Gaussctrl: Multi-view consistent text-driven 3d gaussian splatting editing. In: European conference on computer vision. pp. 55–71. Springer (2024)
53. Wu, J., Bian, J.W., Li, X., Wang, G., Reid, I., Torr, P., Prisacariu, V.A.: GaussCtrl: Multi-View Consistent Text-Driven 3D Gaussian Splatting Editing (2024)
54. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., et al.: Autogen: Enabling next-gen llm applications via multi-agent conversations. In: First conference on language modeling (2024)
55. Xu, T., Chen, J., Chen, P., Zhang, Y., Yu, J., Yang, W.: Tiger: Text-instructed 3d gaussian retrieval and coherent editing. arXiv preprint arXiv:2405.14455 (2024)
56. Yan, Z., Li, L., Shao, Y., Chen, S., Wu, Z., Hwang, J.N., Zhao, H., Remondino, F.: 3dsceneeditor: Controllable 3d scene editing with gaussian splatting. arXiv preprint arXiv:2412.01583 (2024)
57. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K.R., Cao, Y.: React: Synergizing reasoning and acting in language models. In: The eleventh international conference on learning representations (2022)
58. Ye, M., Danelljan, M., Yu, F., Ke, L.: Gaussian grouping: Segment and edit anything in 3d scenes. In: European conference on computer vision. pp. 162–179. Springer (2024)
59. Ye, R., Zhang, J., Liu, Z., Zhu, Z., Yang, S., Li, L., Fu, T., Dernoncourt, F., Zhao, Y., Zhu, J., et al.: Agent banana: High-fidelity image editing with agentic thinking and tooling. arXiv preprint arXiv:2602.09084 (2026)
60. Yi, R., Tian, H., Gu, Z., Lai, Y.K., Rosin, P.L.: Towards artistic image aesthetics assessment: a large-scale dataset and a new method. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22388–22397 (2023)
61. Zhou, Z., Deng, Y., He, X., Dong, W., Tang, F.: Multi-turn consistent image editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15792–15801 (2025)
62. Zhu, C.Y., Liu, X.Y., Xu, K., Yi, R.J.: A survey on 3d editing based on nerf and 3dgs. Frontiers of Computer Science **20**(4), 2004701 (2026)