🎨 GENMAC: Compositional Text-to-Video Generation with Multi-Agent Collaboration

Kaiyi Huang[1]   Yukun Huang[1]   Xuefei Ning[2]   Zinan Lin[3]
Yu Wang[2]   Xihui Liu[1†]

[1] The University of Hong Kong   [2] Tsinghua University   [3] Microsoft Research
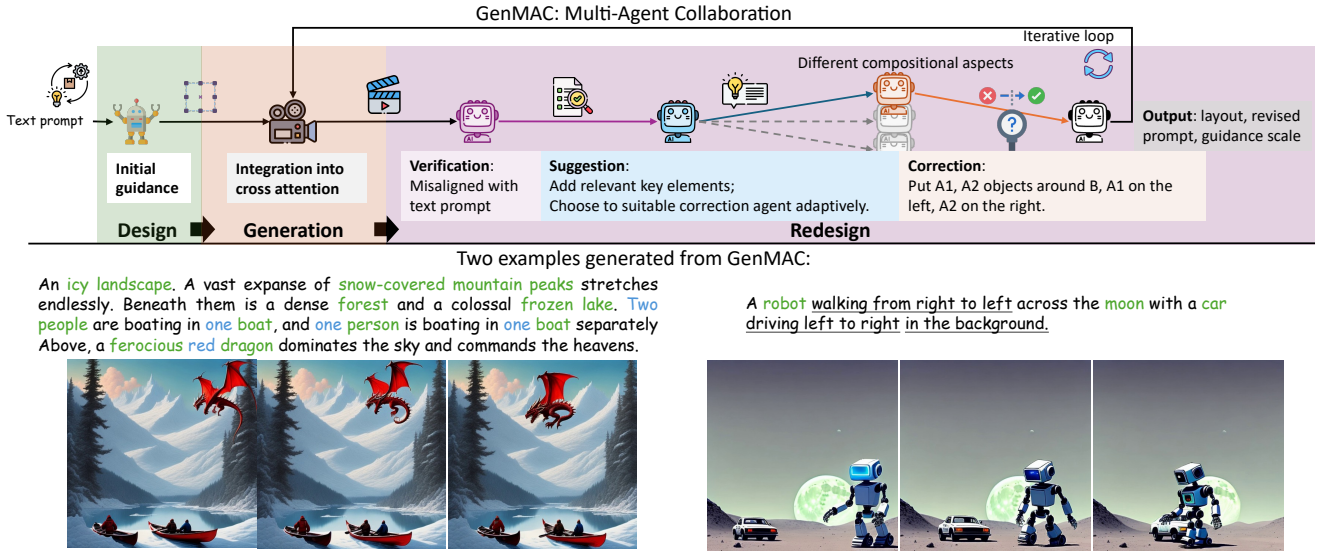Project Page: https://karine-h.github.io/GenMAC/

Figure 1. The first row illustrates our multi-agent collaboration approach, showcasing the collaborative workflow, task decomposition in the REDESIGN stage, and adaptive self-routing for correction agents. The second row presents videos generated by GENMAC based on **complex compositional prompts** involving **multiple objects, attribute binding, quantity, and dynamic motion binding**.

Abstract

Text-to-video generation models have shown significant progress in the recent years. However, they still struggle with generating complex dynamic scenes based on compositional text prompts, such as attribute binding for multiple objects, temporal dynamics associated with different objects, and interactions between objects. Our key motivation is that complex tasks can be decomposed into simpler ones, each handled by a role-specialized MLLM agent. Multiple agents can collaborate together to achieve collective intelligence for complex goals. In this paper, We propose GENMAC, an iterative, multi-agent framework that enables compositional text-to-video generation. The collaborative workflow includes three stages: DESIGN, GENERATION, and REDESIGN, with an iterative loop between the GENERATION *and* REDESIGN *stages to progressively verify and refine the generated videos. The* REDESIGN *stage is the most challenging stage that aims to verify the generated videos, suggest corrections, and redesign the text prompts, frame-wise layouts, and guidance scales for the next iteration of generation. To avoid hallucination of a single MLLM agent, we decompose this stage to four sequentially-executed MLLM-based agents: verification agent, suggestion agent, correction agent, and output structuring agent. Furthermore, to tackle diverse scenarios of compositional text-to-video generation, we design a self-routing mechanism to adaptively select the proper correction agent from a collection of correction agents each specialized for one scenario. Extensive experiments demonstrate the effectiveness of GENMAC, achieving state-of-the art performance in compositional text-to-video generation.*

---

† Corresponding author.

1. Introduction

With the rapid development of diffusion models [17, 44, 45], text-to-video [3, 4, 16, 18, 20, 23, 33, 43, 52, 54, 59, 60, 73] generation has achieved impressive advancements in creating compelling visual content. However, current models face significant challenges when tasked with compositional text-to-video generation, particularly in scenarios involving complex spatiotemporal dynamics. Accurately generating videos following text prompts that capture intricate compositions, such as multiple objects, attribute binding, diverse actions, and interactions over time, remains a challenging problem [22, 46].

Unfortunately, existing techniques [51, 57, 63, 65] fall short in following complex text prompts for compositional text-to-video generation. Previous *single-pass* approaches [51, 65] generate an entire video in a single pass based on a text prompt. Due to the complexity of compositional prompts, the videos generated with a single pass often miss critical contextual details and therefore fail to follow the text prompts. On the other hand, there have been efforts to introduce self-correction mechanisms to text-to-image generation [57, 63], where a Multimodal Large Language Model (MLLM) is used as a *single-agent* to correct misalignments. However, extending such single-agent approaches to video generation presents unique challenges: (1) Considering the increased complexity from images to videos, current MLLMs are not capable of such complex tasks involving accurate visual understanding, multi-step reasoning, and task planning, and may lead to severe hallucinations. (2) Different prompts and generated videos require diverse capabilities to handle distinct compositional aspects, for example, consistency across the video, temporal dynamics, spatial dynamics, *etc*. Relying on a single predefined agent limits the flexibility and generalizability of self-correction.

Our key insight is that even though individual agents may not be capable of complex tasks, task decomposition and role specialization can enable multi-agent collaboration to achieve collective intelligence for complex goals. Inspired by this, we propose two principles to build the multi-agent collaboration system for compositional text-to-video generation: (1) Complex tasks that require multiple steps of observing, understanding, reasoning, and planning, can be decomposed into sequentially-executed simple tasks. (2) Considering the complexity in compositional video generation, where different prompts, video outputs, and refinements may be needed, different scenarios may require different role-specialized "expert" agents to handle. The proper expert agents should be selected adaptively based on the current scenarios and requirements.

Inspired by these insights and principles, we propose an *iterative*, *multi-agent* framework for compositional text-to-video generation. Our overall collaborative workflow is an iterative process is composed of three stages: DESIGN, GENERATION, and REDESIGN, enabling progressive and effective self-corrections over time. The DESIGN stage leverages an MLLM to establish a high-level structure, determining object layout across frames based on the text prompt. The GENERATION stage leverages an off-the-shelf video genereation model conditioned on text prompts and layout controls to synthesize videos using the designed layout and and tet prompts. The REDESIGN stage verifies alignment between the generated video and the text prompt, making necessary adjustments to the design of objects, layouts, or prompts for the next iteration of generation. The REDESIGN stage and GENERATION stage are executed in an iterative loop alternately.

The REDESIGN stage is the most challenging one which requires accurate understanding of videos contents, semantic reasoning of spatial-temporal dynamics, and planning for the correction and refinement in the next generation iteration. Thus, we decompose the REDESIGN stage into multiple sequential tasks - verification, suggestion, correction, and output structuring, executed by different specialized expert agents. Furthermore, to handle the complex scenarios of generated videos, text prompts, and refinement needs, we design a suite of specialized correction agents for correcting the designs from the perspectives of consistency, temporal dynamics, and spatial dynamics, respectively. A self-routing mechanism is introduced to adaptively select the suitable agent for the current scenario.

To the best of our knowledge, we are the first to address the challenging task of compositional text-to-video generation with multi-agent collaboration. Our core insight is task decomposition and role specialization for multi-agent collaboration and collective intelligence. We propose GEN-MAC, an iterative workflow with DESIGN, GENERATION, and REDESIGN stages. In the most challenging REDESIGN stage, we propose the novel sequential task decomposition and adaptive self-routing for specialized agent selection. Extensive experiments demonstrate that our proposed GEN-MAC achieves state-of-the-art performance in compositional text-to-video generation in various aspects, significantly outperforming existing methods.

2. Related Work

Text-to-Video Generation Models. Text-to-video generation [4, 15, 16, 18, 23, 33, 43, 54, 73] has seen advancements with the development of diffusion models [17]. More recently, language model-based methods [5, 6, 25, 52, 68, 69] have enabled large-scale training, leading to significant improvements in generating high-quality videos.

Compositional Text-to-Video Generation. There have been studies on compositional text-to-image generation [7, 9, 10, 13, 14, 22, 24, 27–29, 31, 32, 35, 37, 39, 41, 49, 55, 58, 62, 62, 64]. T2I-CompBench [22] introduces
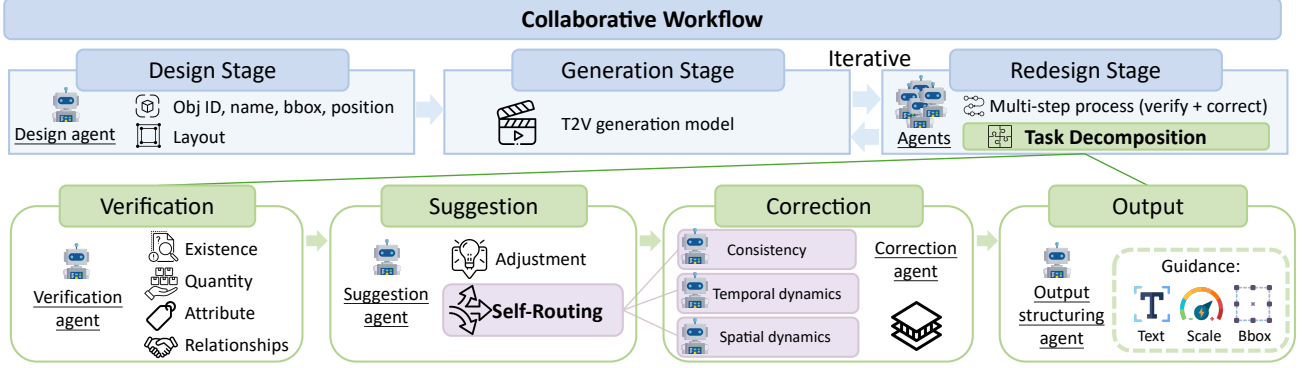
Figure 2. **Framework** of GENMAC. Collaborative workflow includes three stages with an iterative loop: DESIGN, GENERATION, and REDESIGN (Section 3.1). Task decomposition decomposes the redesign stage into four sub-tasks, handled by four agents: verification agent, suggestion agent, correction agent, and output structuring agent (Section 3.2). Self-routing mechanism allows for adaptive selection of suitable correction agent to address the diverse requirements for compositional text-to-video generation (Section 3.3).

the first comprehensive benchmark in evaluating compositionality in text-to-image generation models, with attribute binding, relationships, and complex compositions. T2V-CompBench [46] extends the compositional evaluation to text-to-video generation with the consideration of temporal dimensions. VideoTetris [51] proposes a framework of spatio-temporal compositional diffusion that enables compositional T2V generation. Vico [65] builds a spatial-temporal attention graph to update the noise latent. There exist works that employ an LLM for planning layouts, such as RPG [64] for text-to-image generation, and LVD [29] and VideoDirectorGPT [30] for video generation. However, the existing works focus on generation in one go, failing to meet complex compositional requirements. Our work introduces a collaborative workflow with iterative loop that allows for precise alignment with compositional prompts, progressively refining key elements to achieve greater coherence across spatial and temporal dimensions.

LLM-based Agents. Recent advancements in (M)LLMs have boosted the development of highly capable AI agents, applied across various domains, such as software development [40, 56], robotics [12], scientific research [50], society simulation [38], and beyond. A rapidly growing research focuses on automating interactions with computer environments to solve tasks, such as web manipulation [11, 67], gaming [53], command-line coding [47], and text-to-image generation [57]. Various approaches [19, 38, 48, 61, 71] have been proposed to enable collaboration and communication among multi-agent to overcome hallucinations. While these methods have shown promising results in areas such as automated coding, they often rely on homogeneous agents, limiting the diversity and specialization required for more complex tasks as compositional text-to-video generation. To address these limitations, our work introduces a heterogeneous and hierarchical multi-agent system de-

signed to handle various aspects of compositional requirements in text-to-video generation, expanding the range and effectiveness of multi-agent collaboration in this domain.

3. Methodology

Following the principle of task decomposition and role specialization, we introduce GENMAC, a multi-agent framework for compositional text-to-video generation. GENMAC follows a three-stage workflow: DESIGN → GENERATION → REDESIGN, with an iterative loop, as outlined in Section 3.1. Next, we introduce the sequential task decomposition to enable multi-agent collaboration for the most challenging REDESIGN stage in Section 3.2. Further, to handle the diverse aspects of design correction, we introduce an adaptive self-routing mechanism to select the most suitable agent for the current situation in Section 3.3.

3.1. Overall Collaborative Workflow

Inspired by the human artistic workflow, our multi-agent collaborative framework adopts a DESIGN → GENERATION → REDESIGN pipeline, as shown in Figure 2.

Stage I: DESIGN. Previous studies have shown that LLMs are able to predict dynamic scene layouts based on text prompts [29]. Inspired by this, our DESIGN stage translates the input text prompt into a structured layout, which outlines the key instances, spatial relationships, and temporal dynamics required for compositional video generation. We leverage an LLM to generate structured bounding boxes (which include object IDs, names, box sizes, and positions) for each frame and each instance based on the given text prompt. This stage provides dynamic layout and semantic information to guide the generation stage.

Stage II: GENERATION. In the GENERATION stage, videos are generated conditioned on the structured lay-
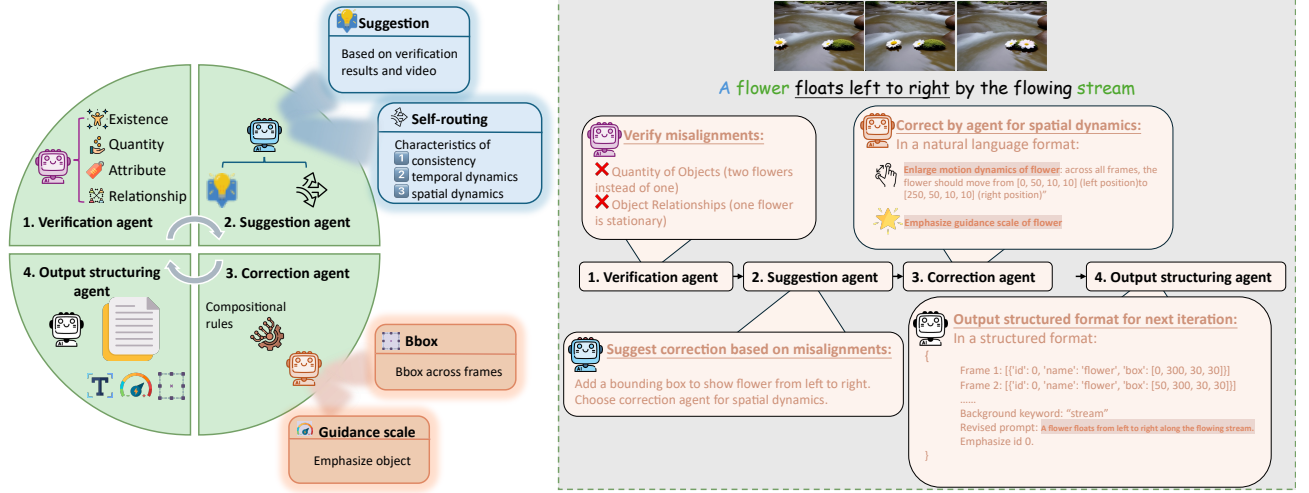
3

Figure 3. **Illustration** of Task Decomposition for the REDESIGN stage (Section 3.2). The diagram illustrates the allocation of roles: verification agent, suggestion agent, correction agent, and output structuring agent within a sequential task breakdown, highlighting the clear responsibilities of each agent.

out and guidance scale predicted from the DESIGN stage. Specifically, we employ an off-the-shelf text-to-video generation model to synthesize visual content that follows the given layout. To incorporate dynamic bounding box control, we follow LVD [29] to inject the structured layout into video diffusion models by guiding the attention maps. In this stage, guidance scale is a hyperparameter defined as the scaling coefficient for bounding box guidance in the diffusion process. The guidance scale is initialized as a predefined value in the first generation iteration, and then adjusted by the REDESIGN stages for subsequent generation iterations. Details are shown in Appendix A.

Stage III: REDESIGN. The REDESIGN stage is the core and most challenging stage of our framework. It aims to detect misalignment between the generated video and the complex compositional prompt, and adjust the design accordingly for re-generation. We find that a single MLLM agent performs poorly in this complex task. However, if we decompose the final goal into a sequence of simple tasks executed one by one, multiple MLLM agents can collaborate together to produce less hallucinations and more reliable results. We explain the task decomposition and multiagent collaboration for the REDESIGN stage in Section 3.2 and Section 3.3.

REDESIGN-GENERATION Loop. For complex compositions, a single pass through the workflow may not address all issues in the generated video. Therefore, we introduce an iterative refinement loop between the GENERATION and REDESIGN stages, allowing progressive correction to meet compositional requirements like attribute binding, spatial relationships, and object counts. With guidance from the REDESIGN stage, including bounding boxes, guid-

ance scales, and revised text prompts, the GENERATION stage iteratively improves the video generation results.

3.2. Task Decomposition for Redesign Stage

The REDESIGN stage requires accurate understanding of the generated videos and text prompts, multi-step reasoning on the video-text misalignments and possible corrections, as well as planning for the new design. We found that this task, especially for videos containing spatial-temporal dynamics, is too difficult for a single MLLM, resulting in hallucinations and inconsistent results (shown in Table 2). However, we observe that a single MLLM can be prompted to be an "expert" for a specific sub-task, *e.g.*, verifying text-video alignment, or suggesting how to correct the video. Motivated by the observation, we decompose the REDESIGN stage into sequentially executed easy tasks, each handled by a specialized MLLM-based expert agent (Figure 3).

Verification Agent checks how well the video content aligns with the text prompt, focusing on four key aspects: object existence, object quantity, attribute binding, and relationship/interaction. This agent takes the text prompts and generated videos as input, and provides information on the misalignments that need to be addressed in the next generation iteration, *e.g.*, "There are two flowers in the video, while the text prompt indicates one flower".

Suggestion Agent is responsible for suggesting how to refine the design and adaptively selecting the suitable correction agent (see Section 3.3). The inputs to this agent include the generated videos and the output of the verification agent (*i.e.*, misalignments that need to be corrected). The outputs of this agent are the suggestions for correction (*e.g.*, "adding a bounding box for the missing tree", or "move the

4

box of apple to the left and make it smaller") and the selection of suitable correction agent (see Section 3.3).

✄ **Correction Agent** is responsible for correcting the bounding box layout, and guidance scale in the current design. The correction agent takes the generated video, current design (bounding box layout and guidance scale), and suggestion from the suggestion agent as inputs. It outputs the corrections to the current design of bounding boxes and guidance scales. For example, "at frame 1, the apple is at [60, 50, 10, 10]; later, it should move to [30, 50, 10, 10]".

👤 **Output Structuring Agent** translates the correction results from the correction agent into structured outputs in json format, ready to be passed to the next generation stage. It takes the generated videos and corrections from the correction agent as input, and outputs the new design in json format. The formatted new design includes information of bounding boxes, (revised) text prompts, and guidance scale.

3.3. Adaptive Self-Routing for Correction Agents

Considering the complexity of tackling spatial-temporal dynamics and consistency for compositional text-to-video generation, it requires the agent to be capable of handling diverse aspects and making proper decisions based on the current situation. We found that a single predefined agent cannot address all required aspects effectively (Table 2). Therefore, we propose a suite of MLLM-based specialized agents, each designed to be an expert for a distinct aspect of video generation. Through an analysis of compositional text-to-video generation, we observe that the most common issues can be categorized into three categories: *consistency*, *temporal dynamics*, and *spatial dynamics*. Therefore, we design the three expert agents for the three perspectives.

Correction Agent for Consistency. For tasks requiring temporal consistency, such as keeping the attribute and spatial layout to be the consistent across the video frames, we introduce a correction agent focusing on maintaining the consistency over time.

Correction Agent for Temporal Dynamics. This agent is designed for cases with temporal dynamics, such as attribute changes or dynamic actions. It adjusts the layout and descriptions dynamically over time, ensuring that each frame reflects the evolving attributes accurately.

Correction Agent for Spatial Dynamics. One challenging scenario for compositional text-to-video generation is the change of object locations over time. We design an expert agent for this scenario to handle moving objects in videos. The agent is particularly good at understanding and reasoning dynamic locations and spatial relationships.

In our self-routing mechanism, the suggestion agent adaptively selects the appropriate correction agent based on the current generated video and the video-text misalignments that needs to be addressed. For example, it routes to the consistency agent to improve the temporal consistency of attributes across the video, and routes to the temporal dynamics agent if the generated video fails to reflect the change of object states over time. An example is illustrated in the right part of Figure 3, where selecting the correction agent for spatial dynamics enables larger motion dynamics, such as the bounding box of the flower moving from the leftmost to the rightmost position. This self-routing process allows GENMAC to make context-aware, precise corrections by selecting the most suitable agent.

4. Experiments

We present the experimental setup in Section 4.1, baseline comparisons in Section 4.2 and Section 4.3, iterative analysis in Section 4.4, and ablation studies of key components in Section 4.5.

4.1. Experimental Setups

Implementation Details. We apply our GENMAC on VideoCrafter2 [8] as the backbone for the GENERATION stage to generate videos with 65 frames, 512x512 resolution. We use GPT-4o [36] as LLM agent. See more details in Appendix B.1.

Evaluated Models. We compare our approach with 17 text-to-video generation models, including 15 open-source models and 2 commercial models: ModelScope [54], ZeroScope [1], Latte [34], Show-1 [72], VideoCrafter2 [8], Open-Sora 1.1 and 1.2 [21], Open-Sora-Plan v1.0.0 and v1.1.0 [26], CogVideoX-5B [66], AnimateDiff [15], VideoTetris [51], Vico [65], MagicTime [70], LVD [29], Pika [2], and Gen-3 [42].

Benchmark and Evaluation Metrics. We use T2V-CompBench [46] as the benchmark to evaluate the quality of compositional text-to-video generation from seven aspects: consistent and dynamic attribute binding, spatial relationships, motion binding, action binding, object interactions, and generative numeracy.

4.2. Quantitative Comparisons

We quantitatively compare our GENMAC with text-to-video generation models, evaluating seven crucial compositional aspects in Table 1. Our GENMAC consistently achieves consistently better performance across seven categories than all the 17 baselines. Among the baselines, the foundation models such as Open-Sora-Plan [26], Open-Sora [21], VideoCrafter2 [8], CogVideoX [66], the commercial Gen-3 [42], and the methods specifically designed for compositionality like VideoTetris [51] and Vico [65], can achieve higher quality. Our method achieves superior performances compositionality, with an exceptional increase in generative numeracy (76.43% above the second-best), and notable improvements in spatial relationships (31.56%), motion binding (16.46%), action binding

5

Table 1. **Quantitative Comparison on T2V-CompBench.** Compared with existing text-to-video generation models and compositional methods, GENMAC demonstrates exceptional performances in consistent attribute binding, dynamic attribute binding, spatial relationships, motion binding, action binding, object interactions, and generative numeracy, indicating our method achieves superior compositional generation ability. We highlight the best score in green, and the second-best value in blue. The baseline data are sourced from [46].

Model	Consist-attr	Dynamic-attr	Spatial	Motion	Action	Interaction	Numeracy
Metric	Grid-LLaVA ↑	D-LLaVA ↑	G-Dino ↑	DOT ↑	Grid-LLaVA ↑	Grid-LLaVA ↑	G-Dino ↑
ModelScope [54]	0.5483	0.1654	0.4220	0.2552	0.4880	0.7075	0.2066
ZeroScope [1]	0.4495	0.1086	0.4073	0.2319	0.4620	0.5550	0.2378
Latte [34]	0.5325	0.1598	0.4476	0.2187	0.5200	0.6625	0.2187
Show-1 [72]	0.6388	0.1828	0.4649	0.2316	0.4940	0.7700	0.1644
VideoCrafter2 [8]	0.6750	0.1850	0.4891	0.2233	0.5800	0.7600	0.2041
Open-Sora 1.1 [21]	0.6370	0.1762	0.5671	0.2317	0.5480	0.7625	0.2363
Open-Sora 1.2 [21]	0.6600	0.1714	0.5406	0.2388	0.5717	0.7400	0.2556
Open-Sora-Plan v1.0.0 [26]	0.5088	0.1562	0.4481	0.2147	0.5120	0.6275	0.1650
Open-Sora-Plan v1.1.0 [26]	0.7413	0.1770	0.5587	0.2187	0.6780	0.7275	0.2928
CogVideoX-5B [66]	0.7220	0.2334	0.5461	0.2943	0.5960	0.7950	0.2603
AnimateDiff [15]	0.4883	0.1764	0.3883	0.2236	0.4140	0.6550	0.0884
VideoTetris [51]	0.7125	0.2066	0.5148	0.2204	0.5280	0.7600	0.2609
Vico [65]	0.7025	0.2376	0.4952	0.2225	0.5480	0.7775	0.2116
LVD [29]	0.5595	0.1499	0.5469	0.2699	0.4960	0.6100	0.0991
MagicTime [70]	-	0.1834	-	-	-	-	-
Pika [2] (Commercial)	0.6513	0.1744	0.5043	0.2221	0.5380	0.6625	0.2613
Gen-3 [42] (Commercial)	0.7045	0.2078	0.5533	0.3111	0.6280	0.7900	0.2169
GENMAC (Ours)	0.7875	0.2498	0.7461	0.3623	0.7273	0.8250	0.5166

Figure 4. **Qualitative Comparison.** Our proposed GENMAC generates videos that accurately adhere to complex compositional scenarios, demonstrating a clear advantage in handling such requirements in comparision with SOTA text-to-video models.

(7.27%), consistent attribute binding (6.23%), dynamic attribute binding (5.13%), and interactions (4.43%).

4.3. Qualitative Comparisons

Comparison with Existing Methods. We show visual comparisons on the video frames of our proposed GEN-MAC and VideoCrafter2 [8], CogVideoX-5B [66], and

Table 2. **Ablation Study**. The `complete framework` achieves the highest scores.

Metric	Consist-attr Grid-LLaVA ↑	Dynamic-attr D-LLaVA ↑	Spatial G-Dino ↑	Motion DOT ↑	Action Grid-LLaVA ↑	Interaction Grid-LLaVA ↑	Numeracy G-Dino ↑
Multiple stages and iterative refinement							
GENERATION	0.6663	0.2308	0.5106	0.2178	0.5640	0.8125	0.2869
+ REDESIGN	0.7208	0.2310	0.6680	0.2468	0.6545	0.8000	0.2869
+ iterative	0.7495	0.2402	0.7032	0.2608	0.7060	0.8125	0.4188
DESIGN + GENERATION	0.7045	0.2320	0.7264	0.3327	0.6880	0.7525	0.4113
+ REDESIGN	0.7513	0.2378	0.7361	0.3474	0.7160	0.7850	0.4794
Role specialization in the REDESIGN *stage*							
Single-agent	0.7200	0.2382	0.7336	0.3336	0.6740	0.7700	0.3984
+ iterative	0.7150	0.2258	0.7336	0.3323	0.6808	0.7700	0.3984
Verification + Correction	0.7138	0.2251	0.7134	0.3179	0.6680	0.7125	0.4284
+ iterative	0.7113	0.2260	0.7149	0.3318	0.6640	0.7686	0.4222
Verification + Suggestion + Correction	0.7370	0.2324	0.7300	0.3173	0.7080	0.7825	0.4469
+ iterative	0.7588	0.2440	0.7450	0.3196	0.7184	0.8175	0.4766
Self-routing for the correction agent							
w/o self-routing	0.7175	0.2316	0.7391	0.3431	0.7240	0.8025	0.4348
+ iterative	0.7325	0.2296	0.7408	0.3517	0.7160	0.8150	0.4647
GENMAC (**ours**)	**0.7875**	**0.2498**	**0.7461**	**0.3623**	**0.7273**	**0.8250**	**0.5166**

A small mouse in a tattered waistcoat reads a tiny book by the light of a glowing mushroom, with dew drops glistening on the grass around him.

Three roses and two sunflowers.

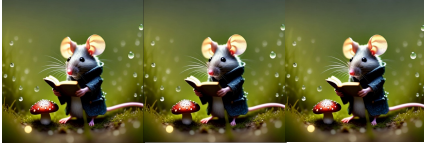A sailboat sails from right to left across the blue waters.

Figure 5. **Qualitative Results.** Our proposed GENMAC generates videos that highly aligned with complex compositional prompts, including attribute binding, multiple objects, quantity, and dynamic motion binding.

VideoTetris [51]. Figure 4 presents the visual comparisons of the video frames generated by GENMAC and existing models, including VideoCrafter2 [8], CogVideoX-5B [66], and VideoTetris [51]. We can observe that existing models struggle to meet compositional requirements. In the left example, VideoCrafter2 [8] omits the VR glasses, VideoTetris [51] generates two cats instead of one and misses the VR glasses, and CogVideoX-5B [66] only shows part of the cat near Big Ben. In the right example, VideoTetris [51] does not depict "sitting on haunches," while VideoCrafter2 [8] and CogVideoX-5B [66] only show partial views of the bear. These examples highlight the challenges in compositional text-to-video generation. In contrast, our proposed GENMAC generates videos that accurately adhere to complex compositional scenarios. See more examples in Appendix B.2.

More Qualitative Examples. The results in Figure 5 show that GENMAC demonstrates better performances in compositionality. See more examples in Appendix B.3.

4.4. Analysis on Iterative Generation

An Example. Figure 6 presents an iterative refinement example. The design agent initially establishes layouts of a rabbit police officer across frames. However, the generated video does not adhere to the "directing traffic" element in the prompt. In the first iteration of REDESIGN, the agents identify the misaligned elements and increase the guidance scale for the bounding box of the rabbit police officer. The generated video, however, shows only a rabbit without the necessary traffic context or police uniform. In the second iteration, agents in REDESIGN stage detect these discrepancies and explicitly add elements like toy cars to indicate traffic, while further increasing the the guidance scale for the bounding box of the rabbit police officer. Additionally, this iteration revises the prompt to include the new elements and reinforce the scenario (*i.e.*, "on the street"). See more examples in Appendix B.4

Iterative Refinement of Different Compositional Aspects. T2V-CompBench consists of seven subsets of
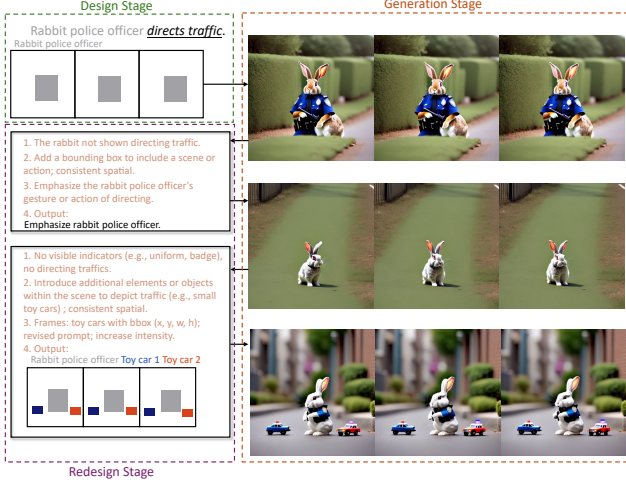
Figure 6. **Visualization** of the iterative refinement process in our multi-agent framework, demonstrating iterations enhance scene accuracy by progressively aligning video content with compositional prompts.
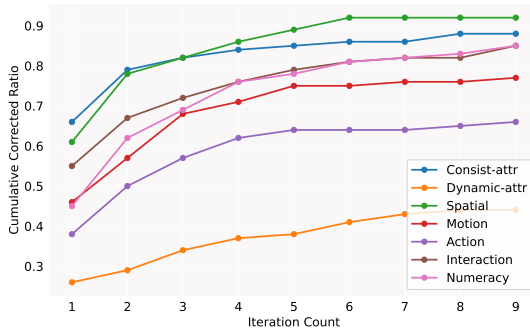

Figure 7. Cumulative Corrected Ratio. For each subset in T2V-CompBench, we calculate the ratio of prompts that have completed the refinement and exited the GENMAC loop to the total size of the subset in each iteration. Dynamic attribute binding remains challenging, while generative numeracy, spatial relationships, and motion binding show substantial improvements from iteration 1 to 9.

prompts, each emphasizing one of the seven compositional aspects. We calculate the cumulative corrected ratio within each subset at every iteration, which is the ratio of prompts that have completed the refinement and exited the GEN-MAC loop to the total size of the subset. As shown in Figure 7, the corrected ratio gradually increases with iterations across all compositional aspects, demonstrating the necessity of iterative refinement.

Among the seven compositional aspects, we can observe that dynamic attribute binding presents the greatest challenge, consistently showing the lowest corrected ratios across iterations. In contrast, consistent attribute binding

and spatial relationships begin with higher corrected ratios. GENMAC demonstrates particular strengths in enhancing generation quality for certain compositional aspects, namely numeracy, spatial relationships, and motion binding, with improvements of 40%, 31%, and 31%, respectively, from iteration 1 to 9.

4.5. Ablation Study

To evaluate the effectiveness of each design component in GENMAC, we perform three ablation studies in Table 2.
Effect of Multiple Stages and Iterative Refinement. The method with only a GENERATION stage yields the lowest generation quality. Introducing a DESIGN stage ("DESIGN+GENERATION") improves the quality. Adding one ("+REDESIGN") or multiple iterative REDESIGN stages ("+iterative") further enhances the quality.
Effect of Role Specialization in the REDESIGN Stage. Separating roles in the REDESIGN stage significantly enhances the generation quality. (1) Quantitative results show that the multi-agent design can bring notable improvements over single-agent framework, *e.g.*, the iterative single-agent framework can only achieve 0.715 on consistent attribute binding, much lower than that of GENMAC (0.7875). (2) Removing the output structuring agent and suggestion agent from the REDESIGN stage leads to significant degradation in quality. For instance, the 2-agent REDESIGN achieves a score of only 0.7113 for consistent attribute binding, compared to 0.7588 for the 3-agent REDESIGN, and 0.7875 for the 4-agent REDESIGN (GENMAC).
Effect of Self-Routing for the Correction Agent. We compare our method with a method version without the self-routing mechanism for the correction agent. In this simplified version, one single correction agent handles information from all compositional aspects. The results clearly highlight the advantage of the self-routing mechanism.

5. Conclusion

In this paper, we address the challenges faced by state-of-the-art video generation models in producing complex compositional video content. Specifically, we introduce an iterative, multi-agent framework that enables high-quality compositional generation. Our workflow incorporates iterative refinement and decomposes the task into three manageable stages: DESIGN, GENERATION, and REDESIGN. We further decompose the core REDESIGN stage into four sequential tasks executed by specialized agents: verification, suggestion, correction, and output structuring. Finally, we design a self-routing mechanism that adaptively selects among multiple correction agents, enabling better handling of diverse compositional aspects. Extensive experiment results confirm the effectiveness and superiority of our method in generating compositional text-to-video generation.

References

[1] ZeroScope. https://huggingface.co/cerspense/zeroscope_v2_576w/, 2023. Accessed: 2024-11-14. 5, 6

[2] Pika Art. https://pika.art/, 2023. Accessed: 2024-11-14. 5, 6

[3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

[4] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2

[5] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022. 2

[6] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 2

[7] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. In *ACM Trans. Graph.*, 2023. 2

[8] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. *arXiv preprint arXiv:2401.09047*, 2024. 5, 6, 7, 1, 2

[9] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *WACV*, 2024. 2

[10] Xiaohui Chen, Yongfei Liu, Yingxiang Yang, Jianbo Yuan, Quanzeng You, Li-Ping Liu, and Hongxia Yang. Reason out your layout: Evoking the layout master from large language models for text-to-image synthesis. *arXiv preprint arXiv:2311.17126*, 2023. 2

[11] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web, 2023. 3

[12] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model, 2023. 3

[13] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *ICLR*, 2023. 2

[14] Hanan Gani, Shariq Farooq Bhat, Muzammal Naseer, Salman Khan, and Peter Wonka. Llm blueprint: Enabling text-to-image generation with complex and detailed prompts. *arXiv preprint arXiv:2310.10640*, 2023. 2

[15] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2, 5, 6

[16] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022. 2

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 2

[18] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[19] Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. Metagpt: Meta programming for a multi-agent collaborative framework, 2024. 3

[20] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[21] hpcaitech. Open-sora: Democratizing efficient video production for all, 2024. 5, 6

[22] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *NeurIPS*, 2024. 2

[23] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023. 2

[24] Yunji Kim, Jiyoung Lee, Jin-Hwa Kim, Jung-Woo Ha, and Jun-Yan Zhu. Dense text-to-image generation with attention modulation. In *ICCV*, 2023. 2

[25] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Rachel Hornung, Hartwig Adam, Hassan Akbari, Yair Alon, Vighnesh Birodkar, et al. Videopoet: A large language model for zero-shot video generation. *arXiv preprint arXiv:2312.14125*, 2023. 2

[26] PKU-Yuan Lab and Tuzhan AI etc. Open-sora-plan, 2024. 5, 6, 2

[27] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *ICCV*, 2023. 2

[28] Zhiheng Li, Martin Renqiang Min, Kai Li, and Chenliang Xu. Stylet2i: Toward compositional and high-fidelity text-to-image synthesis. In *CVPR*, 2022.

[29] Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. Llm-grounded video diffusion models. *arXiv preprint arXiv:2309.17444*, 2023. 2, 3, 4, 5, 6, 1

[30] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. In *COLM*, 2024. 3

[31] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *ECCV*, 2022. 2

[32] Xuantong Liu, Tianyang Hu, Wenjia Wang, Kenji Kawaguchi, and Yuan Yao. Referee can play: An alternative approach to conditional generation via model inversion. *arXiv preprint arXiv:2402.16305*, 2024. 2

[33] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jingren Zhou, and Tieniu Tan. Videofusion: Decomposed diffusion models for high-quality video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10209–10218, 2023. 2

[34] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024. 5, 6

[35] Tuna Han Salih Meral, Enis Simsar, Federico Tombari, and Pinar Yanardag. Conform: Contrast is all you need for high-fidelity text-to-image diffusion models. *arXiv preprint arXiv:2312.06059*, 2023. 2

[36] OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, 2024. Accessed: 2024-11-14. 5

[37] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *NeurIPS*, 2021. 2

[38] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior, 2023. 3

[39] Maitreya Patel, Changhoon Kim, Sheng Cheng, Chitta Baral, and Yezhou Yang. Eclipse: A resource-efficient text-to-image prior for image generations. *arXiv preprint arXiv:2312.04655*, 2023. 2

[40] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chatdev: Communicative agents for software development, 2024. 3

[41] Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. *NeurIPS*, 2024. 2

[42] Runway AI. Gen-3. https://runwayml.com/blog/introducing-gen-3-alpha/, 2024. Accessed: 2024-11-14. 5, 6, 2

[43] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2

[44] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. 2

[45] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. 2

[46] Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation, 2024. 2, 3, 5, 6

[47] Qiushi Sun, Zhirui Chen, Fangzhi Xu, Kanzhi Cheng, Chang Ma, Zhangyue Yin, Jianing Wang, Chengcheng Han, Renyu Zhu, Shuai Yuan, Qipeng Guo, Xipeng Qiu, Pengcheng Yin, Xiaoli Li, Fei Yuan, Lingpeng Kong, Xiang Li, and Zhiyong Wu. A survey of neural code intelligence: Paradigms, advances and beyond, 2024. 3

[48] Qiushi Sun, Zhangyue Yin, Xiang Li, Zhiyong Wu, Xipeng Qiu, and Lingpeng Kong. Corex: Pushing the boundaries of complex reasoning through multi-model collaboration, 2024. 3

[49] Ashkan Taghipour, Morteza Ghahremani, Mohammed Bennamoun, Aref Miri Rekavandi, Hamid Laga, and Farid Boussaid. Box it to bind it: Unified layout control and attribute binding in t2i diffusion models. *arXiv preprint arXiv:2402.17910*, 2024. 2

[50] Xiangru Tang, Qiao Jin, Kunlun Zhu, Tongxin Yuan, Yichi Zhang, Wangchunshu Zhou, Meng Qu, Yilun Zhao, Jian Tang, Zhuosheng Zhang, Arman Cohan, Zhiyong Lu, and Mark Gerstein. Prioritizing safeguarding over autonomy: Risks of llm agents for science, 2024. 3

[51] Ye Tian, Ling Yang, Haotian Yang, Yuan Gao, Yufan Deng, Jingmin Chen, Xintao Wang, Zhaochen Yu, Xin Tao, Pengfei Wan, Di Zhang, and Bin Cui. Videotetris: Towards compositional text-to-video generation, 2024. 2, 3, 5, 6, 7

[52] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *International Conference on Learning Representations*, 2022. 2

[53] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models, 2023. 3

[54] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. 2, 5, 6

[55] Ruichen Wang, Zekang Chen, Chen Chen, Jian Ma, Haonan Lu, and Xiaodong Lin. Compositional text-to-image synthesis with attention map control of diffusion models. *arXiv preprint arXiv:2305.13921*, 2023. 2

[56] Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang H. Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Yanjun Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, Junyang Lin, Robert Brennan, Hao Peng, Heng Ji, and Graham Neubig. Openhands: An open platform for ai software developers as generalist agents, 2024. 3

[57] Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. Genartist: Multimodal llm as an agent for unified image generation and editing, 2024. 2, 3

[58] Zhenyu Wang, Enze Xie, Aoxue Li, Zhongdao Wang, Xihui Liu, and Zhenguo Li. Divide and conquer: Language models can plan and self-correct for compositional text-to-image generation. *arXiv preprint arXiv:2401.15688*, 2024. 2

[59] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021. 2

[60] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. Nüwa: Visual synthesis pretraining for neural visual world creation. In *European conference on computer vision*, pages 720–736. Springer, 2022. 2

[61] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation, 2023. 3

[62] Qiucheng Wu, Yujian Liu, Handong Zhao, Trung Bui, Zhe Lin, Yang Zhang, and Shiyu Chang. Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis. In *ICCV*, 2023. 2

[63] Tsung-Han Wu, Long Lian, Joseph E. Gonzalez, Boyi Li, and Trevor Darrell. Self-correcting llm-controlled diffusion models, 2023. 2

[64] Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. *arXiv preprint arXiv:2401.11708*, 2024. 2, 3

[65] Xingyi Yang and Xinchao Wang. Compositional video generation as flow equalization, 2024. 2, 3, 5, 6

[66] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 5, 6, 7, 2

[67] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents, 2023. 3

[68] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10459–10469, 2023. 2

[69] Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023. 2

[70] Shenghai Yuan, Jinfa Huang, Yujun Shi, Yongqi Xu, Ruijie Zhu, Bin Lin, Xinhua Cheng, Li Yuan, and Jiebo Luo. Magictime: Time-lapse video generation models as metamorphic simulators. *arXiv preprint arXiv:2404.05014*, 2024. 5, 6

[71] Zhengqing Yuan, Yixin Liu, Yihan Cao, Weixiang Sun, Haolong Jia, Ruoxi Chen, Zhaoxu Li, Bin Lin, Li Yuan, Lifang He, Chi Wang, Yanfang Ye, and Lichao Sun. Mora: Enabling generalist video generation via a multi-agent framework, 2024. 3

[72] David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. Show-1: Marrying pixel and latent diffusion models for text-to-video generation. *arXiv preprint arXiv:2309.15818*, 2023. 5, 6

[73] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 2

GenMAC: Compositional Text-to-Video Generation with Multi-Agent Collaboration

Supplementary Material

A. Framework Details

Details in the GENERATION Stage. In the GENERATION stage, the video is generated based on the structured layout predicted in the DESIGN stage. The video is then progressively refined through the REDESIGN-GENERATION loop by adjusting multiple types of guidance including structured layout, guidance scale, and text prompt dynamically.

Following the approach in LVD [29], we utilize the scene layouts across frames predicted by the LLM in the DESIGN stage to direct the initial video generation. During the denoising process, the generation model integrates the information from the text prompt into the latent features via cross-attention layers. To ensure an object appears within its designated bounding box, LVD [29] designs an energy function $\mathcal{L}$ that enforces this constraint. By applying gradient descent of this energy function on partially denoised frames, we can gradually align the video output with the compositional layout specified by the prompt.

Specifically, given the text prompt $\mathcal{P}$, we extract all object tokens (*e.g.*, nouns) as $\mathcal{O} = \{o_1, ..., o_k\}$ from $\mathcal{P}$. For each object token in $\mathcal{O}$, our goal is to encourage the values of A_t^o within the designated bounding box region to be high, where A_t^o denotes the cross-attention map from the latent layers to the object token o at timestep t. The energy function is defined as:

$$
\begin{aligned}
\mathcal{L} &= \sum_{o \in O} \mathcal{L}_o \\
\mathcal{L}_o &= -\beta \cdot \text{Topk}(A_t^o \cdot M_t^o) \\
&\quad + \text{Topk}\big(A_t^o \cdot (1 - M_t^o)\big),
\end{aligned}
\tag{A1}
$$

where M_t^o is a mask indicating the designated bounding box region (ones inside and zeros outside the bounding box) for the object token o at timestep t. Topk computes the average of top-k values in a matrix. This loss function encourages high attention values within the bounding box region. β denotes the guidance scale.

After computing $\mathcal{L}$, we update the latent feature z_t by descending in the direction of its gradient:

$$
z_t' \leftarrow z_t - \alpha_t \cdot \nabla_{z_t} \mathcal{L},
\tag{A2}
$$

where α_t is a scalar defining the step size of the update. This procedure is applied for a subset of denoising timesteps $t = T, T - 1, \ldots, t_{end}$.

Note that different from LVD [29] that uses a fix input text prompt, and predefined guidance scale, our RE-DESIGN stage dynamically adjusts the text prompt and guidance scale by multi-agent collaboration.

Formulation of Agents in the REDESIGN Stage. We assume that the iterative refinement loop at iteration i can be denoted as $(\rho_i, \epsilon_i, \mathcal{V}_i)$, where ρ denotes Reasoning texts in the form of flexible natural language, ϵ denotes Execution texts in structured forms, $\mathcal{V}$ denotes the Video. To be more concrete, Reasoning denotes the MLLM-output text for video understanding, verification, and correction, while Execution means to translate the reasoning results into a more structured format. ϵ can be further expressed as (ϵ^c, ϵ^s), where ϵ^c denotes the appropriate selected correction agent, and ϵ^s denotes the structured outputs. Let π_{veri}, π_{sugg}, π_{corr}, and π_{output} be the verification agent, suggestion agent, correction agent, and output structuring agent separately. ρ_i' and ρ_i'' represent refined reasoning steps that build upon ρ_i through successive agents π_{sugg} and π_{corr}, integrating additional execution choices and previous structured outputs. $\mathcal{S}$ represents the prompts containing target task information and role allocation requirements. Then, the responsibilities and workflows of each agent are defined as follows:

$$
\rho_i = \pi_{\text{veri}}(\mathcal{S}, \mathcal{V}_i, \mathcal{P}),
\tag{A3}
$$

$$
\rho_i', \epsilon_i^c = \pi_{\text{sugg}}(\mathcal{S}, \mathcal{V}_i, \rho_i),
\tag{A4}
$$

$$
\rho_i'' = \pi_{\text{corr}}(\mathcal{S}, \mathcal{V}_i, \rho_i', \epsilon_i^c, \epsilon_{i-1}^s),
\tag{A5}
$$

$$
\epsilon_i^s = \pi_{\text{output}}(\mathcal{S}, \mathcal{V}_i, \rho_i'').
\tag{A6}
$$

Note that π_{corr} uses the structured output from the previous iteration (ϵ_{i-1}^s), as a reference to evaluate the current suggestion (ϵ_i^c) and guide decisions on whether and how to revise it. By analyzing ϵ_{i-1}^s and ϵ_i^c, π_{corr} can identify discrepancies, and unresolved issues. π_{corr} compares the structured output ϵ_{i-1}^s with the new suggestion from π_{sugg} to reason about the need for revisions. This process involves evaluating whether the layout needs adjustments (*e.g.*, structural changes) or whether guidance scales should be emphasized to better align with the desired outcomes. The full examples are illustrated in Table A3 and Table A4.

B. Additional Experimental Results

B.1. Implementation Details.

We apply our GENMAC on VideoCrafter2 [8] as the backbone for the GENERATION stage to generate videos with 65 frames, 512×512 resolution. We find that VideoCrafter2

already works well, and our framework is compatible with other models too. Using more advanced models could potentially enhance performances, which we leave for future exploration. We set 1.0 as the initialized guidance scale, and 0.05 as the incremental step.

B.2. Qualitative Comparisons

We show visual comparisons on the video frames of our proposed method GENMAC and VideoCrafter2 [8], CogVideoX-5B [66], Gen-3 [42], VideoTetris [51], and Open-Sora-Plan [26].

In the first example of Figure A10 (the first row). VideoCrafter2 [8] omits the VR glasses in the last frame. CogVideoX [66] generates the video that the Big Ben is not always on the left side of the cat. Open-Sora-Plan [26] depicts only a cat's head on a human body. VideoTetris [51] generates two cats instead of one and omits the VR glasses, and Gen-3 [42] does not follow the specified scene layout, generating two "Big Ben", one on the left and the other on the right. In the second example of Figure A10 (the second row), VideoTetris [51] and Gen-3 [42] do not depict "sitting on haunches", and Open-Sora-Plan [26] omits "clutching a frothy mug of beer". VideoCrafter2 [8] does not depict "sitting on haunches" with only the head of the bear. CogVideoX [66] only shows partial views of the bear. In the third example of Figure A10 (the third row), VideoCrafter2 [8], Open-Sora-Plan [26], and Gen-3 [42] only generate "one wolf" instead of "a wolf" and "a fox". CogVideoX [66] fails to generate "a microphone" , while VideoTetris [51] lacks the action "plays the drum".

As shown in Figure A11, all existing models fail to generate correct videos with the compositional prompts, including multiple objects, attribute binding, generative numeracy, and action binding. For example, in the first prompt in Figure A11 (the first row), VideoCrafter2 [8], CogVideoX [66], VideoTetris [51] and Open-Sora-Plan [26] fail to generate "a glass sculpture", while Gen-3 [42] omits "ancient vase". For the second prompt in Figure A11 (the second row), Gen-3 [42] and Open-Sora-Plan [26] lack "porcelain rabbit" in some or all frames, while VideoCrafter2 [8], CogVideoX [66], and VideoTetris [51] generate several "golden cactus" instead of "one" indicated in the prompt. For the third prompt in Figure A11 (the third row), Open-Sora-Plan [26] omits the "butterfly", while Gen-3 [42] fails to include the "snail". CogVideoX [66] and VideoTetris [51] do not accurately depict the action "a snail races in a miniature car", and in VideoCrafter2 [8], the wings of the butterfly appear to grow on the snail.

These examples demonstrate the challenges of compositional text-to-video generation faced by both the open-source and the commercial models. Our proposed GEN-MAC correctly reflect the composition of multiple objects, attribute binding, generative numeracy, showing advantages in compositionality.

B.3. Qualitative Results

We show qualitative results in Figure A12 and Figure A13. In Figure A12, our proposed GENMAC show ability to adhere to complex compositional prompts, including attribute binding for multiple objects, temporal dynamics for object movement, and interactions. In Figure A13, we show the qualitative results in various settings of generative numeracy, multiple objects with different attribute bindings, indicating that our proposed GENMAC exhibit superior performances in controllability of compositionality.

B.4. Results on Iterative Generation

We provide visual examples in Figure A8 to illustrate how our multi-agent framework works. In the case of Figure A8a, the design agent creates an initial layout for the rope and boat, but the first generated video lacks the implied "tug" motion. In the REDESIGN stage, verification agent detects this absence of interaction, and suggestion agent proposes adjusting the bounding boxes to create tension between the objects. The correction agent then adjusts the bounding boxes, and the output structuring agent standardizes the guidance, resulting in a refined video that aligns with the prompt.

In Figure A8b, despite the corrected guidance from the DESIGN stage, the initial generated video still exhibits mismatches in both object quantity and motion direction. Over the next two iterations, agents in the REDESIGN stage progressively increase the guidance scale of the car while jointly refining both the motion direction and object quantity to achieve alignment with the prompt.

B.5. Analysis of Various Guidance Settings

We provide an analysis of various guidance settings, including structured layout, guidance scale, and new text prompt in T2V-CompBench [46] in Figure A9. The number of corrections across iterations from one to five is presented in Figure A9a. Since only the structured layout is provided during the DESIGN stage, with the guidance scale and text prompt set to default values, we attribute the corrections entirely to the structured layout (iteration one). With the REDESIGN stage engaged (from iteration two to five), the overall trend of all guidance shows a decline. The guidance scale contributes most to the corrections across iterations two to four, while the contributions of structured layout and new text prompt vary with iterations.

The contribution (%) of different guidance types to the video scores with the DESIGN and REDESIGN stages is depicted in Figure A9b, while with only the REDESIGN stage is shown in Figure A9c. With both DESIGN and REDESIGN stages, structured layout contributes up to 80.4%, followed by new text prompt (12.6%) and guidance scale (7.0%).

(a) **Visualization** of multi-agent collaboration. Initial generation lacks "tug" motion between the rope and boat; REDESIGN agents adjust spatial alignment and visual tension, leading to a final video that aligns with the prompt's interaction requirements.

(b) **Visualization** of the iterative refinement in correcting object quantity and motion direction. The REDESIGN agents adjust guidance scale and alignment over successive iterations, progressively enhancing adherence to the prompt.
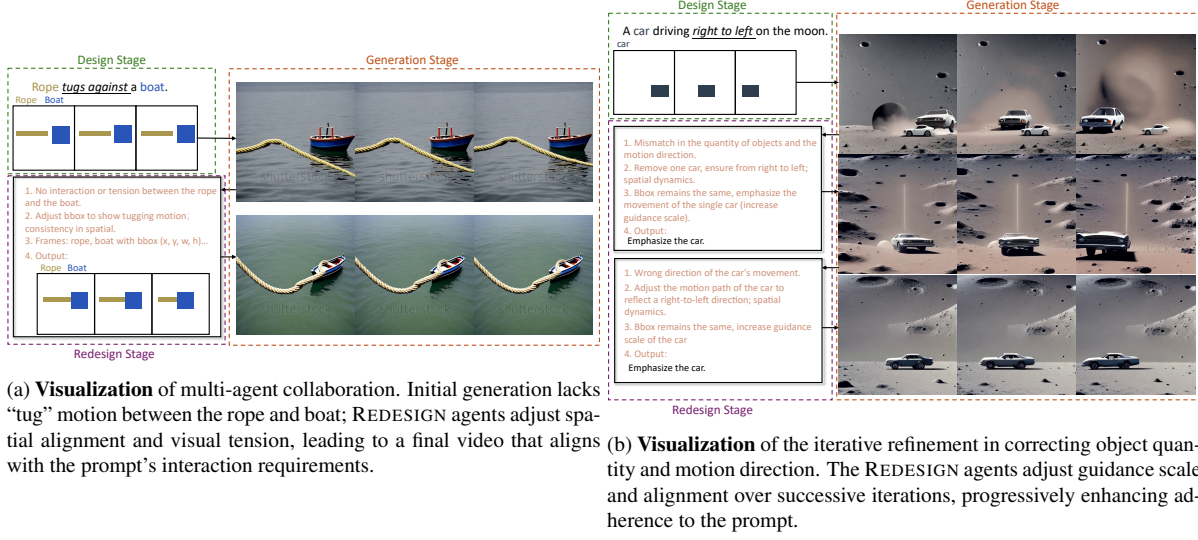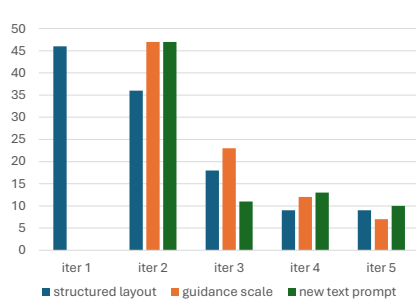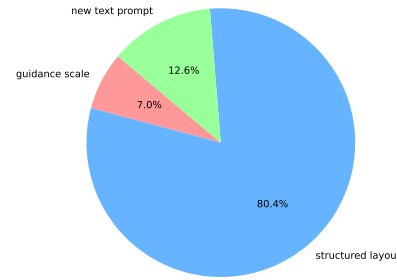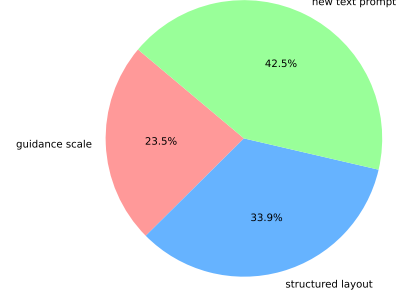
Figure A8. **Visualization** of the multi-agent collaboration.

(a) The number of corrections.

(b) The contribution (%) of different guidance types to the video scores with DESIGN and REDESIGN stages.

(c) The contribution (%) of different guidance types to the video scores with only the REDESIGN stage.

Figure A9. Illustration of the number of corrections and contributions (%) in T2V-CompBench of different guidance types: structured layout, guidance scale, and new text prompt.

With only the REDESIGN stage, the contributions of various guidance are relatively balanced, with the new text prompt contributing slightly more than the structured layout, followed by the guidance scale.

C. Limitation and Potential Negative Social Impacts

Our method GENMAC employs MLLMs as multi-agent for compositional video generation. Although GENMAC shows substantial enhancement over existing methods in compositional text-to-video generation, there is still potential for further improvement. The method depends on the performance of the MLLMs used. Here we adopt GPT-4o as MLLM, for those tasks that exceed the capability of GPT-4o, our method may fail. Besides, GENMAC inherits limitations from the base generation model when it comes to

generating objects or actions it struggles with.

For the potential negative social impacts, the community must recognize the impacts that can result from the misuse of video generation models. These impacts include the creation of misleading or harmful content, which could intensify challenges such as the spread of misinformation and the proliferation of deepfakes.

A giant cat wearing a VR glasses, walking in London street. In the background, the Big Ben is on left.

A large, fluffy bear with a gentle expression, sitting comfortably on its haunches. In one paw, it's clutching a frothy mug of beer, the bubbles catching the light. Scattered around its feet are peanuts.
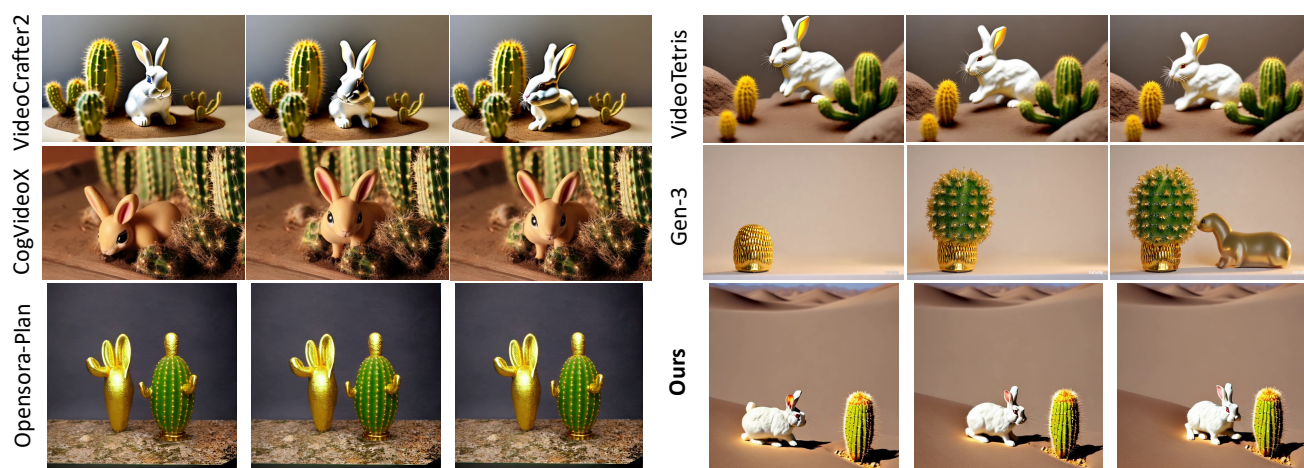

A wolf howls into a microphone and a fox plays the drums.

Figure A10. More qualitative comparisons.

Ancient vase displayed next to a glass sculpture.

Porcelain rabbit hopping by a golden cactus.

A snail races slowly in a miniature car, a butterfly flies alongside.

Figure A11. More qualitative comparisons.

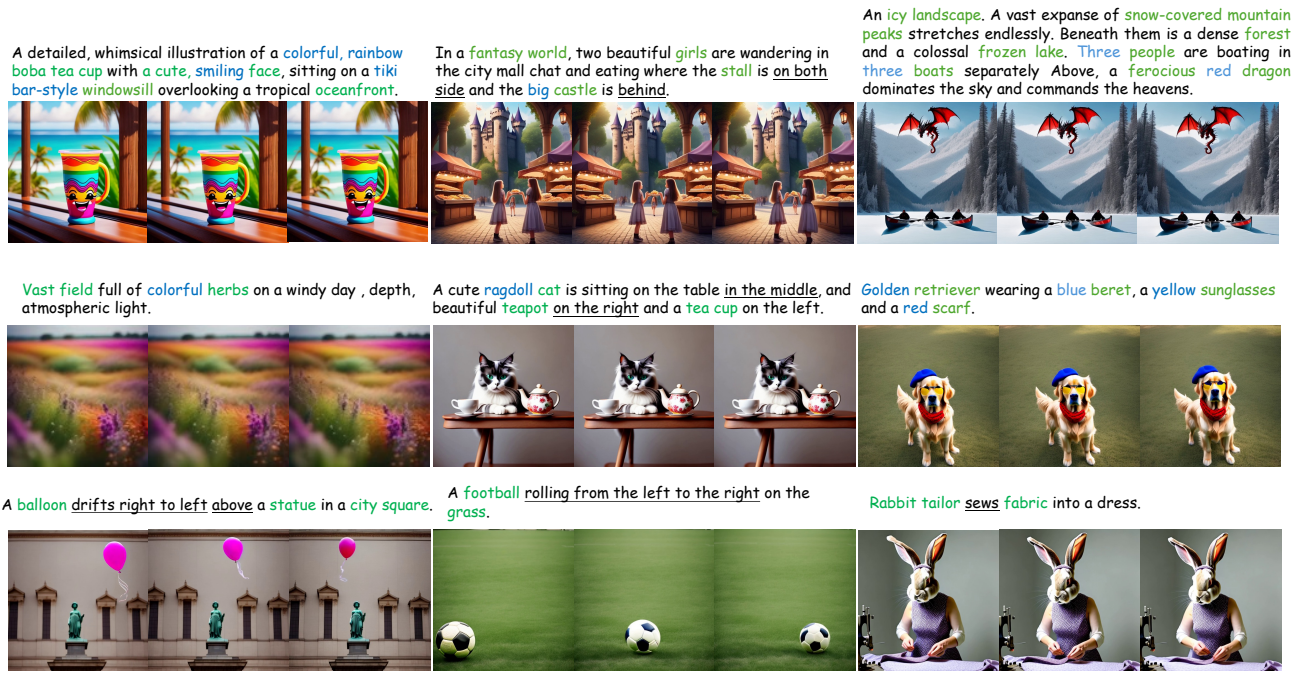Figure A12. **Qualitative results** of GENMAC. GENMAC shows ability to adhere to complex compositional prompts, including attribute binding for multiple objects, temporal dynamics for object movement, and interactions.
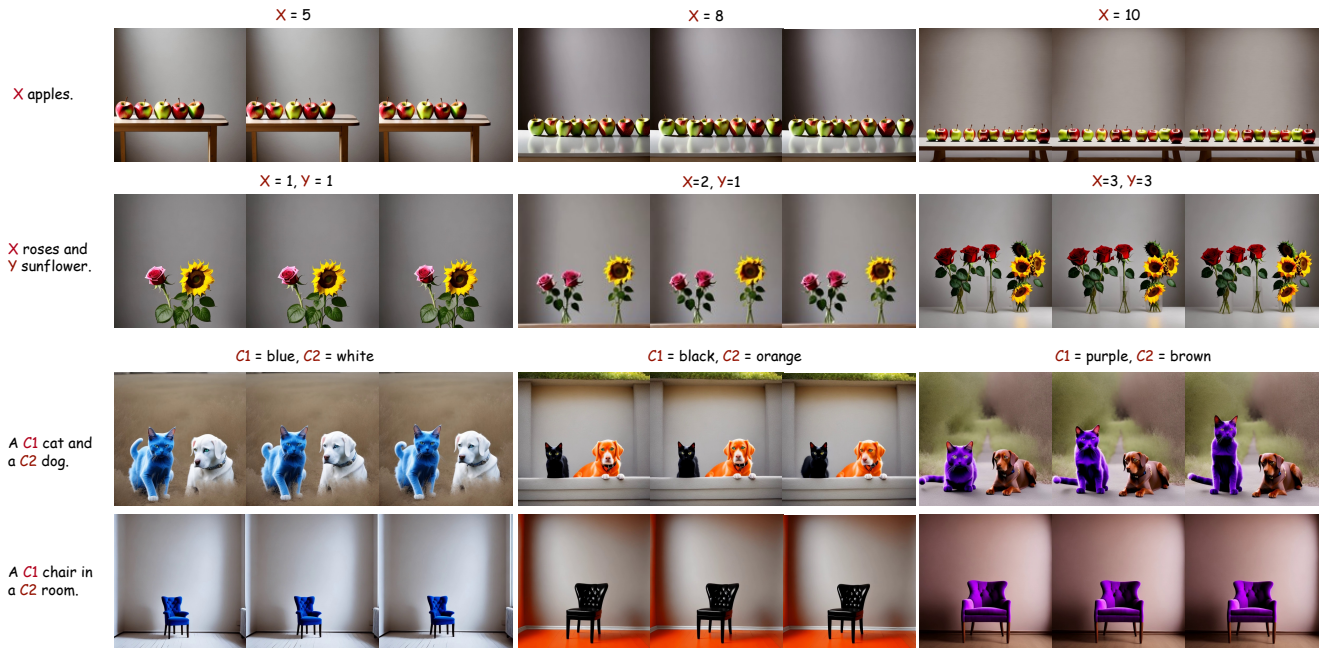

Figure A13. **Qualitative results** of GENMAC. GENMAC shows exhibit superior performances in controllability of generative numeracy, multiple objects with different attributes in compositionality.

Table A3. Our full example for Figure A13, with the prompt "A car driving right to left on the moon".

DESIGN stage

Design	Reasoning: The car is driving from right to left, so its x-coordinate should decrease while its y-coordinate remains constant. The moon's surface is flat and has low gravity, so the car's movement will be smooth and consistent. Frame 1: [{'id': 0, 'name': 'car', 'box': [400, 350, 100, 50]}] Frame 2: [{'id': 0, 'name': 'car', 'box': [320, 350, 100, 50]}] Frame 3: [{'id': 0, 'name': 'car', 'box': [240, 350, 100, 50]}] Frame 4: [{'id': 0, 'name': 'car', 'box': [160, 350, 100, 50]}] Frame 5: [{'id': 0, 'name': 'car', 'box': [80, 350, 100, 50]}] Frame 6: [{'id': 0, 'name': 'car', 'box': [0, 350, 100, 50]}] Background keyword: moon Generation suggestion: None New prompt: A car drives from right to left on the moon's surface.

REDESIGN stage **iter 1**

Verification	**verification agent** The alignment check reveals the following issues: 1. Quantity of Objects: There are two cars present in the video instead of one. 2. Motion Direction: The cars are not clearly shown moving from right to left. 3. Correctness of Attributes: The cars appear to be on the moon, which aligns with the prompt. Overall, there is a mismatch in the quantity of objects and the motion direction.
Suggestion	**suggestion agent** 1. Suggest corrections for the bounding boxes: - Remove one of the cars to correct the quantity issue. - Ensure the remaining car is clearly shown moving from right to left by adjusting its position across frames to depict motion. 2. Choose the suitable correction agent: B1. (correction agent for spatial dynamics)
Correction	**correction agent** The correction suggestion indicates that one of the cars should be removed to correct the quantity issue. The remaining car should be shown moving from right to left, which aligns with the previous bounding boxes. Here are the corrected bounding boxes: Corrected Bounding Boxes: - Frame 1: ['id': 0, 'name': 'car', 'box': [400, 350, 100, 50]] - Frame 2: ['id': 0, 'name': 'car', 'box': [320, 350, 100, 50]] - Frame 3: ['id': 0, 'name': 'car', 'box': [240, 350, 100, 50]] - Frame 4: ['id': 0, 'name': 'car', 'box': [160, 350, 100, 50]] - Frame 5: ['id': 0, 'name': 'car', 'box': [80, 350, 100, 50]] - Frame 6: ['id': 0, 'name': 'car', 'box': [0, 350, 100, 50]] Explanation: The bounding boxes remain the same as the previous ones, as they already depict the car moving from right to left. The suggestion to generation is to emphasize the movement of the single car across the frames.
Output	**output structuring agent** Reasoning: The sequence depicts a single car moving from right to left across the frames. Initially, the car is positioned on the right side of the frame and gradually moves to the left, maintaining a consistent y-coordinate, which aligns with the prompt of a car driving on the moon. Frame 1: [{'id': 0, 'name': 'car', 'box': [400, 350, 100, 50]}] Frame 2: [{'id': 0, 'name': 'car', 'box': [320, 350, 100, 50]}] Frame 3: [{'id': 0, 'name': 'car', 'box': [240, 350, 100, 50]}] Frame 4: [{'id': 0, 'name': 'car', 'box': [160, 350, 100, 50]}] Frame 5: [{'id': 0, 'name': 'car', 'box': [80, 350, 100, 50]}] Frame 6: [{'id': 0, 'name': 'car', 'box': [0, 350, 100, 50]}] Background keyword: moon Generation suggestion: emphasize id 0 New prompt: A car driving right to left on the moon.

iter 2

Verification	**verification agent** The alignment check reveals the following issues: Direction of Motion: The car is moving from left to right, which is opposite to the prompt's description of right to left. Existence of Specified Objects: The car and the moon surface are present, which aligns with the prompt. Quantity of Objects: The single car is correctly depicted. Correctness of Object Attributes: The car and the lunar surface appear as expected. Overall, the main issue is the direction of the car's movement. Alignment Check Result: The video does not fully align with the prompt due to the incorrect motion direction of the car.
Suggestion	**suggestion agent** 1. Suggest corrections for the bounding boxes: Adjust the motion path of the car to reflect a right-to-left direction instead of left-to-right. 2. Choose the suitable correction agent: B1. (correction agent for spatial dynamics)
Correction	**correction agent** Based on the correction suggestion, the bounding boxes need to be adjusted to reflect the car moving from right to left. Here are the corrected bounding boxes: **Corrected Bounding Boxes:** - **Frame 1:** ['id': 0, 'name': 'car', 'box': [400, 350, 100, 50]] - **Frame 2:** ['id': 0, 'name': 'car', 'box': [320, 350, 100, 50]] - **Frame 3:** ['id': 0, 'name': 'car', 'box': [240, 350, 100, 50]]- **Frame 4:** ['id': 0, 'name': 'car', 'box': [160, 350, 100, 50]] - **Frame 5:** ['id': 0, 'name': 'car', 'box': [80, 350, 100, 50]]- **Frame 6:** ['id': 0, 'name': 'car', 'box': [0, 350, 100, 50]] **Comparison and Suggestion:** The corrected bounding boxes are the same as the previous ones, indicating that the car is already moving from right to left as intended. **Suggestion to Generation:** Emphasize the car (id 0) to ensure it stands out against the moon background.
Output	**output structuring agent** Reasoning: The sequence depicts a car moving from right to left across the frames. The car's x-coordinate decreases consistently, indicating its movement from right to left as described in the prompt. The background is the moon, providing a unique setting for the car's journey. Frame 1: [{'id': 0, 'name': 'car', 'box': [400, 350, 100, 50]}] Frame 2: [{'id': 0, 'name': 'car', 'box': [320, 350, 100, 50]}] Frame 3: [{'id': 0, 'name': 'car', 'box': [240, 350, 100, 50]}] Frame 4: [{'id': 0, 'name': 'car', 'box': [160, 350, 100, 50]}] Frame 5: [{'id': 0, 'name': 'car', 'box': [80, 350, 100, 50]}] Frame 6: [{'id': 0, 'name': 'car', 'box': [0, 350, 100, 50]}] Background keyword: moon Generation Suggestion : emphasize id 0. New Prompt: A car driving right to left on the moon.

Table A4. Our full example for Figure 6 in the main paper, with the prompt "Rabbit police officer directs traffic".

DESIGN stage

Design	Reasoning: The rabbit police officer will likely be standing in one place, directing traffic, so its position will remain relatively stable across frames. Frame 1: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 2: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 3: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 4: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 5: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 6: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Background keyword: street New prompt: A rabbit police officer directing traffic on the street.

REDESIGN stage — iter 1

Verification	**verification agent** The alignment check reveals the following issues: 1. **Existence of Specified Objects**: The rabbit is present and dressed as a police officer. 2. **Quantity of Objects**: There is one rabbit police officer. 3. **Correctness of Object Attributes**: The rabbit is wearing a police uniform, which aligns with the prompt. 4. **Accuracy of Relationships**: The rabbit is not shown directing traffic in the frames provided. Overall, the video does not fully align with the prompt as the action of directing traffic is missing.
Suggestion	**suggestion agent** 1. **Suggest corrections for the bounding boxes**: - Add a bounding box to include a scene or action where the rabbit is directing traffic. This could involve positioning the rabbit with an arm raised or using a gesture to indicate traffic direction. 2. **Choose the suitable correction agent**: A. (correction agent for consistency) The focus is on maintaining the consistent attributes of the rabbit as a police officer while introducing the fixed spatial relationship of directing traffic.
Correction	**correction agent** **Corrected Bounding Boxes:** 1. **Frame 1:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] 2. **Frame 2:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] 3. **Frame 3:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] 4. **Frame 4:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] 5. **Frame 5:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] 6. **Frame 6:** - ['id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]] **Explanation:** - The bounding boxes remain the same as the previous ones since the suggestion primarily involves adding an action or gesture, which is not reflected in the bounding box dimensions. - The rabbit police officer's position is stable, and the box size is appropriate for visibility. **Suggestion to Generation:** - Emphasize the rabbit police officer's gesture or action of directing traffic within the existing bounding box.
Output	**output structuring agent** Reasoning: The video features a rabbit police officer consistently positioned in the same location across all frames, with no additional objects or changes in state. The prompt needs to be rephrased to include all objects, but since only one object is present, the original prompt is sufficient. Frame 1: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 2: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 3: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 4: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 5: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Frame 6: [{'id': 0, 'name': 'rabbit police officer', 'box': [206, 256, 100, 150]}] Background keyword: street Generation suggestion: emphasize id 0 New prompt: Rabbit police officer directs traffic.

iter 2

Verification	**verification agent** The video does not align well with the prompt. Detailed alignment check results: - The specified object, "rabbit police officer," is not clearly depicted as a police officer. There are no visible indicators (e.g., uniform, badge) that suggest the rabbit is a police officer. - The action "directs traffic" is not shown. The rabbit is simply standing on a path without any indication of directing traffic. Overall, the video lacks the necessary attributes and actions described in the prompt.
Corrections	**suggestion agent** (1) Suggest corrections for the bounding boxes: - Add a bounding box around the rabbit and include elements that indicate it is a police officer, such as a small uniform or badge. - Introduce additional elements or objects within the scene to depict traffic (e.g., small toy cars) and position them in a way that suggests the rabbit is directing them. - Ensure the rabbit's posture or gestures imply directing traffic, such as an arm raised or a whistle. (2) Choose the suitable correction agent: A. (correction agent for consistency)
Correction	**correction agent** Corrected Bounding Boxes: Frame 1: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Frame 2: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Frame 3: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Frame 4: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Frame 5: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Frame 6: Rabbit police officer: [200, 250, 112, 162], Toy car 1: [50, 400, 60, 30], Toy car 2: [400, 400, 60, 30] Differences and Suggestions: Differences: Additional bounding boxes for toy cars have been added to depict traffic, and the rabbit's bounding box has been slightly adjusted for better coverage. Suggestion to Generation: Emphasize the rabbit police officer's directing posture and ensure the toy cars are clearly visible in the scene.
Output	**output structuring agent** Reasoning: The scene involves a rabbit police officer consistently directing traffic, with two toy cars present throughout the sequence. The rabbit's position remains constant, while the toy cars are positioned at opposite ends of the frame, maintaining their positions to depict a traffic scenario. Frame 1: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Frame 2: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Frame 3: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Frame 4: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Frame 5: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Frame 6: [{'id': 0, 'name': 'rabbit police officer', 'box': [200, 250, 112, 162]}, {'id': 1, 'name': 'toy car 1', 'box': [50, 400, 60, 30]}, {'id': 2, 'name': 'toy car 2', 'box': [400, 400, 60, 30]}] Background keyword: street Generation suggestion: emphasize id 0 New prompt: A rabbit police officer directs traffic with two toy cars on the street.