Sync-R1: Bridging Understanding and Generation for Personalized Reasoning via Co-operative Reinforcement Learning

Zijun Shen[1,*] **Ruichuan An**[2,*,†] **Sihan Yang**[2,*] **Ziyu Guo**[3]
Hao Liang[2,4] **Ming Lu**[2] **Renrui Zhang**[3] **Wentao Zhang**[2,4,‡]

[1]Nanjing University [2]Peking University
[3]The Chinese University of Hong Kong [4]Zhongguancun Academy

[*]Equal contribution [†]Project leader [‡]Corresponding author

Abstract

Unified Multimodal Models (UMMs) excel in general tasks but struggle to bridge the gap between personalized understanding and generation. Prior works largely rely on implicit token-level alignment via supervised fine-tuning, which fails to fully capture the potential synergy between comprehension and creation. In this work, we propose Sync-R1, an end-to-end reinforcement learning framework that jointly optimizes personalized understanding and generation within a single, explicit reasoning loop. Through this unified feedback process, Sync-R1 enables personalized comprehension to guide content creation, while the resulting generation quality reciprocally refines understanding within an integrated reward landscape. To efficiently orchestrate this dual-task synergy, we introduce Sync-GRPO, a reinforcement learning method utilizing an ensemble reward system. Furthermore, we propose Dynamic Group Scaling (DGS), which adaptively filters low-potential trajectories to reduce gradient variance and accelerate convergence. To better reflect real-world complexity, we introduce UnifyBench++, featuring denser textual descriptions and richer user contexts. Experimental results demonstrate that Sync-R1 achieves state-of-the-art performance, showcasing superior cross-task reasoning and robust personalization without requiring complex cold-start procedures.

1 Introduction

The rapid evolution of Unified Vision-Language Models (ULM) [Wu et al., 2024a, Jiang et al., 2025, Li et al., 2025, Xie et al., 2025a, Deng et al., 2025] has established a powerful paradigm for general-purpose multimodal AI, yet the potential of unified models to execute complex, personalized reasoning remains largely under-explored. Real-world tasks typically demand a complex interplay between understanding user-specific contexts and generating personalized content. Existing approaches often rely on implicit token-level sharing or simplistic multi-task supervised fine-tuning (SFT) [Zhong et al., 2026], failing to capture deep contextual nuances [Nguyen et al., 2025]. These approaches either treat understanding and generation as disjoint objectives [Tian et al., 2025] or demand high-cost training pipelines that fail to capture the deep synergistic relationship between tasks. Consequently, two critical challenges remain: (1) How to establish an explicit mechanism where personalized understanding and generation mutually enhance one another [Ye et al., 2026], and (2) How to achieve such coordination efficiently within a unified framework.

To address these challenges, we propose **Sync-R1**, an end-to-end reinforcement learning framework that achieves explicit and synergistic co-optimization of personalized understanding and generation.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

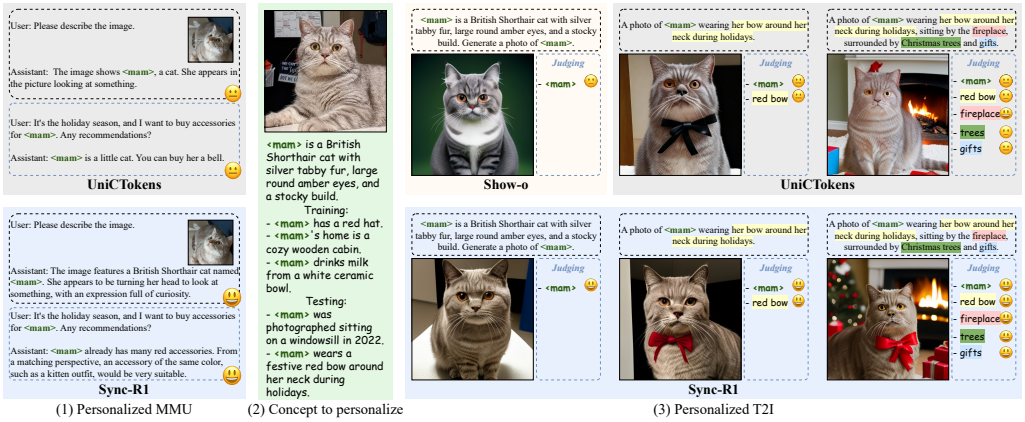


Figure 1: **Capability overview of Sync-R1.** Our framework jointly optimizes personalized understanding and generation within a unified reasoning trajectory via Sync-GRPO. By establishing an explicit synergistic loop, Sync-R1 leverages the reciprocal enhancement between comprehension and creation to achieve precise integration of personalized concepts. Beyond standard personalization, Sync-R1 excels in challenging scenarios, including dense-text and reasoning-based generation tasks with missing concept information.

Unlike previous methods that rely on latent embedding alignment [An et al., 2025a], we bridge the task gap by orchestrating a unified reasoning trajectory [Penha et al., 2025]: the model first extracts fine-grained conceptual information from user-provided context (*understanding*) to guide the subsequent creation of personalized content (*generation*). Crucially, this trajectory is optimized holistically under a composite reward function that evaluates both the fidelity of the understanding output and the quality of the final generation. This ensures that the feedback from generation substantially informs and improves the understanding process. Thus, we establish a reciprocal feedback loop where comprehension provides the semantic foundation for generation, and generation provides the objective signal to calibrate comprehension.

While the synergistic potential between understanding and generation is well-recognized, effectively coordinating these distinct reasoning objectives necessitates a sophisticated optimization strategy. To this end, we propose **Sync-GRPO**, a reinforcement learning method designed to jointly optimize both tasks within a unified loop. We advocate for RL over supervised fine-tuning (SFT) based on two considerations: **(1) Eliciting Latent Synergy:** Instead of redundantly re-teaching skills via SFT, RL is uniquely suited to elicit and coordinate the robust foundational capabilities already inherent in UMMs [Chen et al., 2025, Liang et al., 2025]. This approach enables the discovery of optimal integration strategies through adaptive self-improvement. **(2) Enhancing Coherent Reasoning:** Leveraging RL's proven efficacy in complex reasoning [Bai et al., 2022, Hu et al., 2025], we design rewards to explicitly incentivize logical alignment between understanding and generation. This steers the model toward coherent, contextually grounded personalization.

To address the prohibitive computational cost of multimodal RL, we further introduce **Dynamic Group Scaling (DGS)**. This adaptive sampling mechanism dynamically prunes low-potential trajectories in the initial stages of generation, reducing gradient variance and accelerating convergence while maintaining high sample quality for reward estimation. Moreover, to establish a more rigorous evaluation benchmark for personalized reasoning, we extend UnifyBench [An et al., 2025a] to **UnifyBench++**, enriching it with denser textual descriptions and more diverse user contexts [An et al., 2026]. Extensive experiments demonstrate that Sync-R1 achieves strong performance across the evaluated tasks, exhibiting robust cross-task synergy without requiring costly cold-start fine-tuning.

Our primary contributions are summarized as follows:

- We propose Sync-R1, a unified RL framework establishing an explicit reasoning loop to synergistically co-optimize personalized understanding and generation.
- We develop Sync-GRPO and DGS to effectively and efficiently coordinate the joint optimization of personalized understanding and generation tasks.

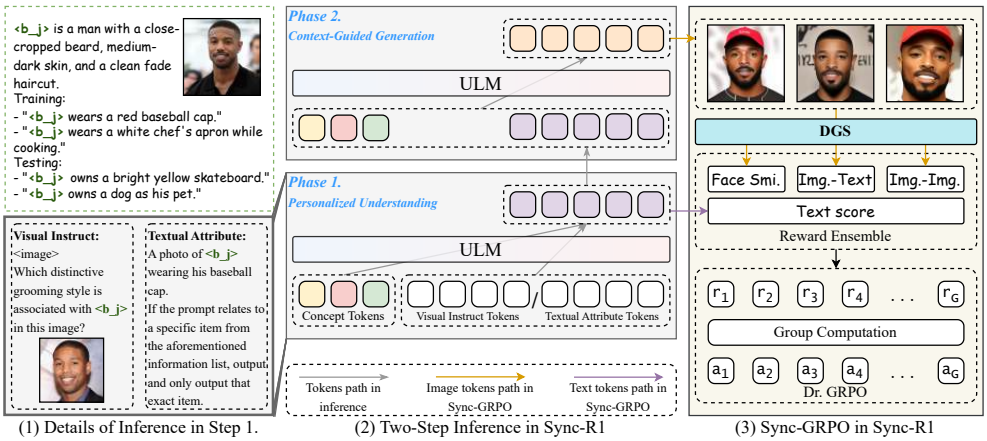


Figure 2: **Overview of Sync-R1.** (1) Inputs to the visual and textual reasoning pathways. (2) Intermediate reasoning outputs are incorporated into compound prompts to guide generation. (3) A shared trajectory-level advantage updates the reasoning and image-token policies, while DGS accelerates training through early trajectory screening.

- We introduce UnifyBench++, an enhanced benchmark for robust evaluation. Extensive experiments show that Sync-R1 sets a new state of the art, demonstrating superior cross-task synergy without costly cold-start procedures.

2 Related Work

Reinforcement Learning for Multimodal Tasks. Reinforcement Learning (RL) has established itself as a pivotal paradigm for enhancing model adaptability and complex task-specific reasoning [Yuan et al., 2025, Yue et al., 2025]. In visual generation, RL is increasingly explored to transcend the limitations of supervised fine-tuning. Early efforts integrated Chain-of-Thought (CoT) reasoning into the generation process [Chern et al., 2025, Pan et al., 2025a], while the advent of Group Relative Policy Optimization (GRPO) [DeepSeek-AI et al., 2025] has catalyzed efficient RL-driven visual synthesis [Tong et al., 2025, Han et al., 2025, Xue et al., 2025, Xiao et al., 2025]. However, applying RL to unified multimodal models [Mao et al., 2025, Zhang et al., 2025a]—specifically to foster explicit synergy between comprehension and creation—remains largely unexplored. Existing personalization methods, in contrast, often rely on isolated optimization or implicit cross-task transfer. To explicitly couple personalized understanding and generation, we propose **Sync-R1**, a reinforcement learning framework that jointly optimizes both tasks within a unified reasoning trajectory.

Personalized Models. The core objective of model personalization is to integrate concept-specific information into model outputs via diverse techniques [Zheng et al., 2026, Feng et al., 2026]. In the generative domain, recent approaches focus on recontextualization conditioned on text. For instance, DreamBooth [Ruiz et al., 2023] ensures subject authenticity in generated results, while Textual Inversion [Gal et al., 2022] optimizes special tokens using soft-prompt tuning. In the realm of LLMs, personalization has been achieved through dual-tower structures that endow models with user-aware capabilities [Pi et al., 2024]. For MLLMs, recent works [An et al., 2025b] enhance output relevance by combining user data with visual content via fine-tuning or Retrieval-Augmented Generation (RAG). Within Unified MLLMs, Chameleon [Team, 2025] employs "text embedding optimization + Transformer fine-tuning" for information injection, while Yo'Chameleon [Nguyen et al., 2025] utilizes soft prompts. UniCTokens [An et al., 2025a] was the pioneer in focusing on the mutual promotion between understanding and generation, significantly reducing sample requirements and expanding the scope of personalization. However, existing unified methods often suffer from cumbersome training protocols and suboptimal performance in complex semantic generation tasks, highlighting an urgent need for more efficient and robust solutions [Yang et al., 2025].

3 Method

To facilitate the explicit synergy between personalized understanding and generation, we propose Sync-R1 (as shown in Figure 2), an end-to-end GRPO reinforcement learning framework that establishes an explicit reasoning loop where comprehension and creation are jointly optimized within a unified trajectory. In this section, we first define the dual-task personalized reasoning process, then introduce the synergistic reward ensemble that jointly evaluates intermediate reasoning and final generation, followed by the Sync-GRPO optimization objective. Finally, we present Dynamic Group Scaling (DGS), which accelerates convergence through gradient variance reduction.

3.1 Explicit Synergistic Reasoning

Existing works [Nguyen et al., 2025, An et al., 2025a] typically achieve cross-task information sharing at the token level via multi-task supervised fine-tuning. However, this implicit alignment often creates semantic bottlenecks when processing complex, highly personalized contexts. To overcome this fundamental limitation, we decompose the personalized task into a structured two-phase explicit reasoning trajectory as detailed below:

Phase I: Personalized Understanding. We study concept-level personalization, adapting a unified multimodal model to a target concept using limited reference images and textual information. Each concept is represented by $c = (\mathcal{I}_c, \text{info}_c, \mathcal{E}_c)$. Reference images $\mathcal{I}_c$ and stable attributes info_c provide core knowledge, while $\mathcal{E}_c$ contains external `extra_info`. Personalized understanding requires recognizing and reasoning about the concept, while personalized generation requires preserving its identity and following the requested attributes. The understanding stage extracts relevant visual information or selects task-relevant attributes from this list through the two pathways shown in Figure 2(1): **(1) Visual Instruct Reasoning:** Given a reference image I and a descriptive query Q, the model distills a reasoning result IR that captures the fine-grained attributes of the concept: $I + Q \xrightarrow{\text{UMM}} IR_{vis}$. **(2) Textual Attribute Reasoning:** Given an external attribute list $\mathcal{E} =$ `extra_info` and a reasoning prompt IP, the model selects the attribute relevant to the requested task: $(\mathcal{E}, IP) \xrightarrow{\text{UMM}} IR_{\text{txt}}$. The selected attribute is subsequently incorporated into the compound prompt for generation.

Phase II: Context-Guided Generation. Based on the reasoning results IR from Phase I, we construct a Compound Prompt CP for each pathway to supervise the image generation process, ensuring a direct semantic flow from comprehension to creation. **(1) For Visual Instruct Reasoning:** We combine a basic prompt BP (e.g., "a photo of $\langle$sks$\rangle$", where $\langle$sks$\rangle$ is a learnable unique identifier initialized via supervised fine-tuning.) with the visual reasoning result IR_{vis} to form the compound prompt: $CP_{\text{vis}} = BP \oplus IR_{\text{vis}}$. Here, IR_{vis} enriches the generation with fine-grained visual attributes (e.g., "$\langle$sks$\rangle$ has a clean fade haircut"), explicitly guiding the generation. **(2) For Textual Attribute Reasoning:** We concatenate the original inference prompt IP (e.g., "a photo of $\langle$sks$\rangle$ wearing his baseball cap") with the textual reasoning result IR_{txt} (e.g., "$\langle$sks$\rangle$ wears a red baseball hat") to form the compound prompt: $CP_{\text{txt}} = IP \oplus IR_{\text{txt}}$. This allows the model to resolve ambiguities or missing details in IP by incorporating the explicitly reasoned attribute IR_{txt}. This two-phase decomposition forms an explicit reasoning loop, directly optimizable via reinforcement learning. By designing a tailored reward function that jointly assesses IR and the final generation, we seamlessly enable the synergistic co-optimization of both comprehension and creation.

3.2 Synergistic Reward Ensemble

To close the feedback loop between reasoning and generation, we define a unified reward function:

$$R_{total} = w_1 R_{\text{TIER}} + w_2 R_{\text{BER}} + w_3 R_{\text{DER}} + w_4 R_{\text{FER}} \tag{1}$$

The ensemble incorporates: (1) TIER (Text Inference Evaluation) via ERNIE 3.0 [Sun et al., 2021] to measure logical consistency; (2) BER (BLIP Evaluation [Li et al., 2023]) for cross-modal semantic alignment; (3) DER (DINOv2 Evaluation [Oquab et al., 2024]) for identity preservation of personalized concepts; and (4) FER (FaceNet Evaluation) for high-fidelity facial synthesis. Scaling coefficients w_i are calibrated via sensitivity analysis (see Appendix D).

3.3 Sync-GRPO

To optimize the explicit reasoning trajectory defined in Section 3.1, we introduce **Sync-GRPO**, an objective designed to close the feedback loop between understanding and generation. Our approach builds upon the Group Relative Policy Optimization (GRPO) framework [DeepSeek-AI et al., 2025], eliminating the value function dependency to enable joint optimization across discrete modalities. However, unlike pure text models, our multimodal backbone (Show-o [Xie et al., 2025b]) employs a discrete diffusion mechanism based on MaskGIT [Chang et al., 2022]. In this paradigm, image generation is formulated as a non-autoregressive mask-and-predict process. Starting from a fully masked state I_T, the model iteratively predicts and refines the entire set of image tokens over T denoising steps to reach the final image I_0:

$$P_\theta(I_0 \mid I_T, CP) \propto \prod_{t=1}^{T} \prod_{k=1}^{N} \pi_\theta(I_{t-1,k} \mid I_t, CP) \tag{2}$$

where N is the number of image tokens, and $\pi_\theta(I_{t-1,k} \mid I_t, CP)$ represents the probability distribution over all tokens at step t. While only masked tokens are updated in each step during inference, the reinforcement learning objective considers the predictions across all positions as meaningful policy decisions that contribute to the final image quality. We formally model the complete synergistic trajectory as $o_i = (IR_i, \{I_{i,t}\}_{t=0}^{T})$. To ensure stable optimization across varying task difficulties and prevent length-bias from degrading generative quality, we adopt the *Dr.GRPO* formulation [Liu et al., 2025a]. The probability ratio $D_{i,j}(\theta)$, which serves as the fundamental learning signal, is defined to span the entire loop, capturing both understanding and generation information:

$$D_{i,j}(\theta) = \begin{cases} \dfrac{\pi_\theta(IR_{i,j} \mid q, IR_{i,<j})}{\pi_{\theta_{\text{old}}}(IR_{i,j} \mid q, IR_{i,<j})}, j \in \text{Tokens in } IR_i \\ \dfrac{\pi_\theta(I_{i,t-1,k} \mid I_{i,t}, CP_i)}{\pi_{\theta_{\text{old}}}(I_{i,t-1,k} \mid I_{i,t}, CP_i)}, \begin{array}{l} j \in \text{Image Tokens} \\ \text{at step } t, \text{ index } k \end{array} \end{cases} \tag{3}$$

In this formulation, q is determined by the randomly selected pathway (Visual Instruct or Textual Attribute), and k denotes the spatial index within the image token grid at denoising step t. To balance the contribution of reasoning versus generation, we assign separate weights α_{text} and α_{image} to the two components. The final objective (detailed in Appendix C) maximizes the clipped advantage estimated from this joint ratio while penalizing the KL-divergence from a reference policy for stability. Sync-GRPO retains Dr.GRPO's advantage formulation and extends it to connected autoregressive reasoning and per-step image-generation decisions. The coupling operates in two directions: the intermediate reasoning output explicitly conditions image generation through the compound prompt, and generation-side rewards contribute to the advantage used to optimize the reasoning-token policy loss. The latter provides cross-stage credit assignment through policy optimization over the sampled reasoning tokens. Controlled ablations of these two mechanisms are reported in Appendix H.6.

3.4 Dynamic Group Scaling

Motivation. The performance of Sync-GRPO is intrinsically tied to the group size G, but a larger G demands prohibitively long training times. Inspired by selective sampling paradigms [Shrivastava et al., 2025], we propose Dynamic Group Scaling (DGS), which evaluates multiple trajectory candidates and retains only the most promising ones through early-stage screening. This strategy allows the model to concentrate its gradient computation on the most informative regions of the trajectory space, effectively achieving the variance-reduction benefits of an enlarged candidate pool at a fraction of the full-generation cost while maintaining high generation quality.

Method. As illustrated in Figure 3, we exploit the observation that semantic structures often emerge in early denoising stages [An et al., 2025c]. Specifically, at $t \approx 0.2T_{total}$, we compute a surrogate reward $\tilde{R}$ (e.g., the BLIP Evaluation Reward shown in Section 3.2). DGS iteratively samples trajectories until it obtains G successful candidates that satisfy the threshold criterion $\tilde{R}(o_i) > T$. By only completing the generation process for these G candidates, DGS ensures that the resulting gradients are derived from high-potential, semantically grounded samples.

Theoretical Analysis. The efficacy of DGS stems from its systematic reduction of gradient estimator variance. Let the terminal reward R and surrogate $\tilde{R}$ for each trajectory be i.i.d. and follow a bivariate normal distribution with correlation $\rho = \text{Corr}(R, \tilde{R}) > 0$.

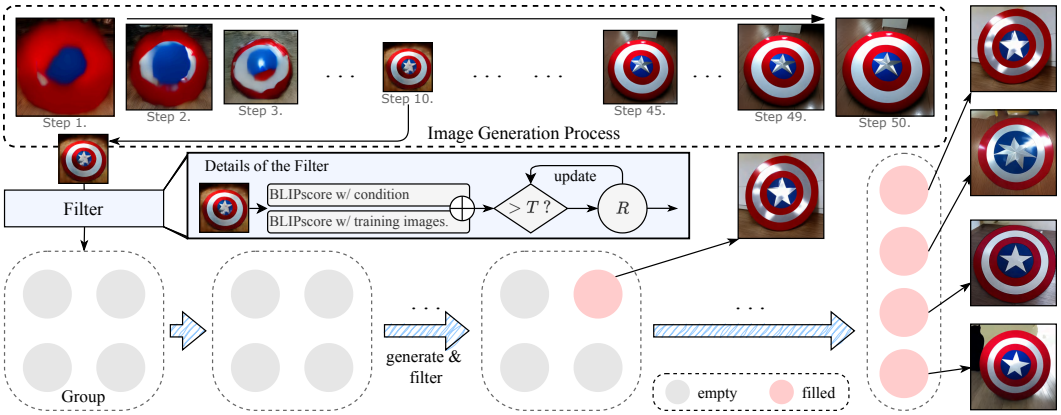


Figure 3: **Illustration of Dynamic Group Scaling (DGS).** To improve efficiency, we employ a preliminary assessment at the 10th step of the Show-o generation process. The generation continues only if the intermediate score exceeds a dynamically updated threshold. This threshold is adjusted adaptively to maintain the probability of high-quality generation selection at a target level, effectively filtering out suboptimal trajectories early.

Theorem 3.1 (Variance Reduction)**.** *Under a linearized policy gradient regime (omitting clipping and KL terms for analysis), we have:*

$$\mathrm{Tr}(\mathrm{Var}[\nabla_\theta \mathcal{J}]) \propto \mathrm{Var}[\hat{A}] \tag{4}$$

where $\nabla_\theta \mathcal{J}$ is the standard Sync-GRPO gradient using G random samples and:

$$\hat{A} = R - \mathrm{E}[R] \tag{5}$$

If we denote N as the total number of the evaluated trajectories, the trace of the gradient covariance for DGS-enhanced Sync-GRPO satisfies:

$$\frac{\mathrm{Tr}(\mathrm{Var}[\nabla_\theta \mathcal{J}^{(DGS)}])}{\mathrm{Tr}(\mathrm{Var}[\nabla_\theta \mathcal{J}])} \leq 1 - \rho^2(1 - \frac{G}{N}) \tag{6}$$

Theorem 3.1 implies that DGS suppresses gradient noise by a factor proportional to the filtration intensity $1 - \frac{G}{N}$ and the surrogate fidelity ρ. This leads to an amplification of the Stochastic Gradient Signal-to-Noise Ratio (SNR), given by $\mathrm{SNR}(\theta) = \|\mathbb{E}[\mathbf{g}]\|^2 / \mathrm{Tr}(\mathrm{Var}[\mathbf{g}])$ for a gradient estimator $\mathbf{g}$.

Corollary 3.2 (SNR Amplification)**.** *Based on Theorem 3.1, the SNR of the DGS update relates to the baseline as:*

$$\frac{\mathrm{SNR}^{(DGS)}(\theta)}{\mathrm{SNR}(\theta)} > \frac{1}{1 - \rho^2(1 - \frac{G}{N})}. \tag{7}$$

Higher SNR translates to more reliable update directions, accelerating traversal across the optimization manifold. Related proofs are provided in Appendix E.

Adaptive Threshold Control. To maintain a robust and stable selection ratio near the Target Pass Rate (TPR) as π_θ evolves during training, we implement a trend-aware closed-loop controller. Let PR_k be the pass rate at iteration k, and $\Delta_k = PR_k - \text{TPR}$. The threshold T evolves via the following adaptive dynamics:

$$\begin{cases} T_{k+1} & = T_k^{1+(\mu - \epsilon_0 \cdot \mathbb{I}[\tau_k \Delta_k < 0])\Delta_k} \\ \tau_{k+1} & = \eta \tau_k + (1-\eta)(T_{k+1} - T_k) \end{cases} \tag{8}$$

where μ controls sensitivity, η is momentum decay, and the indicator $\mathbb{I}[\tau_k \Delta_k < 0]$ reduces gain upon trend conflict. This adaptive control mechanism ensures that T robustly tracks the $(1 - \text{TPR})$-quantile of $\tilde{R}$, effectively filtering out stochastic noise while preserving high information density.

Table 1: **Quantitative results on Unifybench++.** TP = Text Prompt. IP = Image Prompt. Columns with * are newly extended from Unifybench. Best and second best performances are highlighted.

Type	Method	Model Size	Und.							Gen.							
			Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
			Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
Upper Bound	GPT-4o+TP	200B	0.742	0.746	0.819	0.784	0.863	0.611	0.699	0.306	0.684	0.374	0.697	0.357	0.776	0.403	0.692
	GPT-4o+IP	200B	0.788	0.745	0.854	0.751	0.784	0.594	0.658	0.309	0.787	0.382	0.672	0.376	0.804	0.381	0.796
	Real Images	-	-	-	-	-	-	-	-	-	0.833	-	-	-	-	-	-
Und. Only	Yo'LLaVA	13B	0.921	0.327	0.529	0.616	0.625	0.614	0.594	-	-	-	-	-	-	-	-
	MC-LLaVA	13B	0.924	0.297	0.511	0.623	0.636	0.606	0.583	-	-	-	-	-	-	-	-
	RAP-MLLM	13B	0.936	0.332	0.595	0.624	0.617	0.712	0.723	-	-	-	-	-	-	-	-
	Qwen2.5-VL + TP	3B	0.669	0.218	0.341	0.404	0.726	0.579	0.770		-	-	-	-	-	-	-
	Yo'LLaVA(Phi-1.5)	1.3B	0.769	0.225	0.484	0.493	0.493	0.511	0.498	-	-	-	-	-	-	-	-
Gen. Only	Text inversion	1.0B	-	-	-	-	-	-	-	0.248	0.632	0.322	0.538	0.294	0.673	0.351	0.633
	DreamBooth (SD)	1.0B	-	-	-	-	-	-	-	0.282	0.645	0.323	0.558	0.313	0.661	0.367	0.651
Unified Model	Chameleon+TP	7B	0.685	0.206	0.313	0.411	0.489	0.509	0.560	0.186	0.542	0.281	0.463	0.193	0.516	0.309	0.549
	Chameleon+IP	7B	0.497	0.216	0.374	0.446	0.497	0.411	0.532	0.165	0.514	0.266	0.449	0.190	0.532	0.299	0.503
	Show-o+TP	1.3B	0.562	0.217	0.337	0.462	0.412	0.507	0.579	0.263	0.663	0.299	0.558	0.235	0.679	0.341	0.660
	Yo'Chameleon	7B	0.764	0.231	0.399	0.470	0.511	0.506	0.582	0.235	0.697	0.289	0.601	0.273	0.704	0.321	0.702
	UniCTokens	1.3B	0.792	0.238	0.385	0.505	0.521	0.546	0.601	0.280	0.750	0.298	0.639	0.282	0.762	0.317	0.712
	Sync-R1	1.3B	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

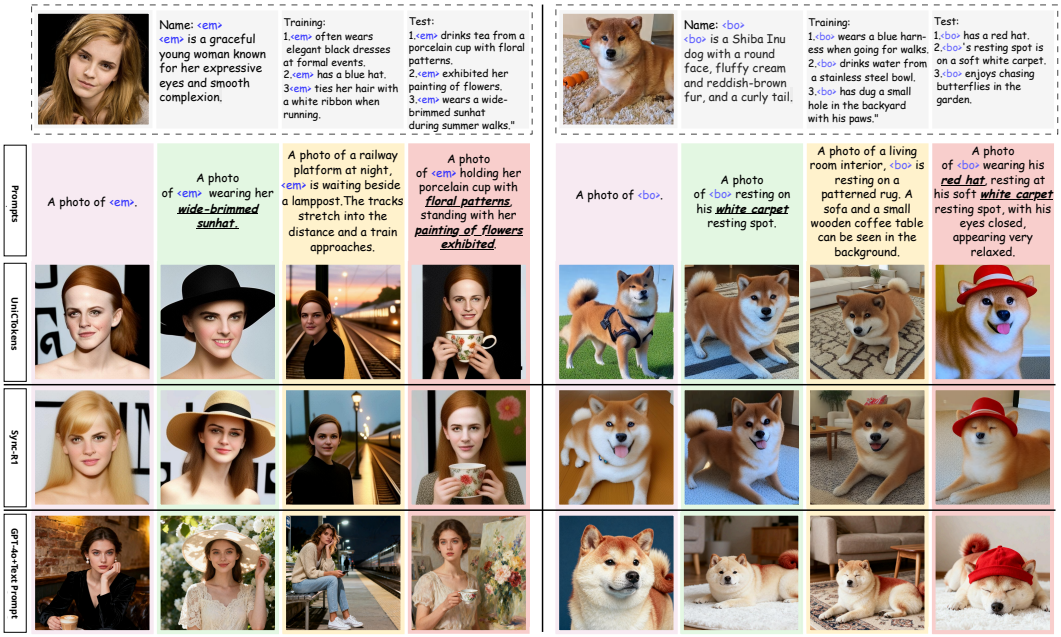


Figure 4: **Qualitative comparison.** We present a visual comparison between Sync-R1 and baseline methods. The underlined text highlights specific information that requires the model to infer from the personalized context correctly.

4 Experiment

In this section, we comprehensively evaluate the efficacy of our proposed Sync-R1 framework across a diverse range of complex, personalized understanding and generation tasks. Our experiments are carefully designed to investigate: (1) whether the synergistic loop improves fine-grained multimodal alignment; (2) the computational efficiency gains provided by the DGS mechanism; and (3) the framework's robustness across different initialization recipes.

Table 2: **Ablation study on the effectiveness of the two reasoning components and DGS method.**

Method	Und.							Gen.							
	Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
	Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
w/o Und.Rea.	0.808	0.224	0.427	0.511	0.519	0.549	0.621	0.301	0.762	0.323	0.637	0.289	0.765	0.321	0.727
w/o VIA	0.823	0.239	0.489	0.587	0.591	0.571	0.626	0.299	0.759	0.306	0.641	0.301	0.799	0.343	0.741
w/o TIA	0.855	0.222	0.431	0.572	0.588	0.597	0.646	0.305	0.757	0.319	0.639	0.291	0.773	0.329	0.728
w/o DGS	0.848	0.243	0.498	0.590	0.583	0.607	0.643	0.305	0.766	0.331	0.632	0.307	0.796	0.338	0.739
Sync-R1	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

Table 3: **Ablation study on different initialization methods.**

Method	Und.							Gen.							
	Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
	Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
Joint-SFT	0.712	0.224	0.387	0.489	0.482	0.529	0.577	0.264	0.680	0.261	0.607	0.266	0.724	0.288	0.653
Joint+Sync-GRPO	0.789	0.231	0.468	0.548	0.557	0.581	0.629	0.287	0.695	0.291	0.616	0.303	0.757	0.322	0.689
UniCTokens-SFT	0.792	0.238	0.385	0.505	0.521	0.546	0.601	0.280	0.750	0.298	0.639	0.282	0.762	0.317	0.712
Sync-R1 (w/o SFT)	0.322	0.227	0.365	0.289	0.191	0.264	0.322	0.259	0.673	0.278	0.607	0.248	0.613	0.301	0.584
Sync-R1	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

4.1 Experiment Setup

Implementation Details. We utilize Show-o (1.3B, 512×512) [Xie et al., 2025b] as the multimodal backbone. We adopt the multi-stage supervised fine-tuning protocol established in An et al. [2025a] to initialize the personalized tokens. In the RL stage, we set the group size $G = 9$ and maintain a balanced 1:1 ratio between Visual Instruct Reasoning and Textual Attribute Reasoning to promote stable policy optimization. All models are trained on $8\times$ NVIDIA H100 GPUs. Hyperparameters and SFT training protocols are provided in Appendix G for further technical details.

Baseline Comparisons. We benchmark Sync-R1 against a comprehensive suite of models across diverse paradigms to thoroughly assess its overall capabilities: (1) unified architectures, including UniCTokens [An et al., 2025a] (adopting its official training protocol) and Yo'Chameleon [Nguyen et al., 2025] (retrained at 7B scale with 1k images per concept); (2) specialized understanding-only models and generation-only models; and (3) GPT-4o, serving as an approximate upper bound. Further implementation details are provided in Appendix G.

Dataset. To evaluate high-order inferential logic and dense semantic synthesis, we augment UnifyBench into UnifyBench++. For understanding, we incorporate Reasoning and Dense Reasoning tasks to test concept disambiguation. For generation, we introduce Reasoning Generation alongside Dense Generation and Dense Personalized Generation to assess fine-grained alignment. This creates a rigorous stress-test for synergistic multimodal reasoning (see Appendix B).

Multimodal Evaluation Suite. For reasoning accuracy, we employ BLEU and GPT-4o-based semantic scoring to ensure precise textual understanding. For generative fidelity, we employ CLIP-T and GPT-4o to assess the visual alignment with dense prompts and inferred personalized concepts throughout the reasoning trajectory.

4.2 Main Results

Performance on UnifyBench. We first evaluate the foundational capabilities of Sync-R1 on standard personalized understanding tasks, including Recognition (Rec.), Visual Question Answering (VQA), and Question Answering (QA) [Alaluf et al., 2024]. As presented in Table 1, despite possessing only 1.3B parameters, Sync-R1 delivers exceptional performance, achieving an average improvement of **12.2%** over previous state-of-the-art (SOTA) methods across all understanding metrics. In terms of generation, we utilize the Pure Gen. metric to assess identity-preserving generation capabilities. Sync-R1 consistently outperforms other unified models, securing the top rank. Collectively, these results

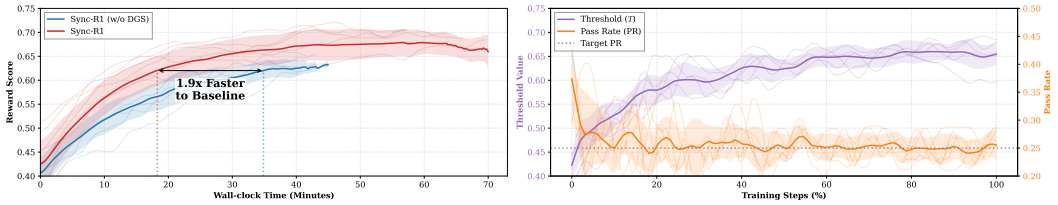


Figure 5: **Efficiency and dynamics of DGS. (a) Convergence efficiency: Sync-R1** (Red) reaches 98% of the baseline's peak performance approx. **1.9× faster** in wall-clock time, confirming that high sample efficiency outweighs the selection overhead. **(b) Adaptive dynamics:** The threshold T (Purple) dynamically adapts to pass rate fluctuations (Orange) to anchor the selection ratio at TPR, ensuring robust training stability across diverse concepts.

corroborate that our framework offers a highly resource-efficient solution for unifying personalized understanding and generation without compromising performance on established benchmarks.

Performance on UnifyBench++. The primary contribution of our evaluation lies in UnifyBench++, which is designed to stress-test the model's capacity for reasoning-intensive and information-dense scenarios. As shown in Table 1, Sync-R1 achieves SOTA results across all newly introduced metrics, validating the effectiveness of our Sync-GRPO strategy. **(1) Understanding Reasoning:** We observe substantial gains in the Dense Rea. metric compared to baselines. This strongly indicates that our model has transcended simple pattern matching and acquired a deeper, more granular understanding of personalized concepts embedded within complex contexts. **(2) Generation Reasoning:** In generation tasks requiring reasoning and high information density, our method demonstrates a significant leap over the baseline UniCTokens. Specifically, we achieve relative improvements of **14.1%** in Dense Gen., **10.0%** in Rea. Gen., and **8.8%** in Dense Rea. Gen. **(3) Qualitative Analysis:** As visualized in Figure 4, we evaluate fidelity across a complexity spectrum, ranging from Pure Generation to challenging Dense Reasoning Generation. Crucially, Sync-R1 demonstrates superior robustness in both explicit structural arrangement and implicit attribute inference. In Dense Generation, it maintains precise spatial relations under heavy semantic load (e.g., correctly positioning $\langle em \rangle$ *beside* the lamppost), whereas baselines often suffer from severe spatial dislocation. Furthermore, in complex Reasoning scenarios, Sync-R1 successfully retrieves and accurately visualizes implicit attributes (e.g., the painting of flowers) where previous methods succumb to catastrophic neglect of crucial details. This confirms that our framework enhances both the spatial coherence of complex scenes and the semantic depth of personalized understanding.

4.3 Ablation Study

In this section, we dissect the contribution of each component within Sync-R1, focusing on the necessity of the explicit synergistic loop and the efficiency gains from DGS.

Efficacy of the Explicit Synergistic Loop. To test whether personalized understanding and generation mutually reinforce each other, we selectively disable parts of the reasoning trajectory in Table 2. Removing Visual Instruct Reasoning sharply reduces MMU metrics such as Rec. and weakens generative fidelity, confirming the role of visual-attribute extraction in semantic guidance. Excluding Textual Attribute Reasoning mainly hurts logic-intensive metrics such as Rea. and Rea. Gen., showing its importance for resolving semantic ambiguities. Most critically, removing the entire understanding loop reduces the method to RL-tuned generation: Pure Gen. remains relatively stable, but user-specific personalization metrics collapse. This confirms our core premise that generation quality alone is insufficient; explicit understanding is essential for faithful user alignment. Additional directional ablations and inference-time interventions examine the two coupling mechanisms in Appendix H.6. A matched comparison with Standard GRPO from the same SFT checkpoint further shows gains across all reported subtasks, supporting the benefit of Sync-GRPO beyond generic RL post-training (Appendix H.7).

Efficiency and Dynamics of DGS. We analyze the training dynamics in Figure 5. As illustrated in Figure 5(a), DGS achieves a $1.9\times$ wall-clock acceleration, confirming that the benefits of gradient variance reduction significantly outweigh the selection overhead. The underlying mechanism is visualized in Figure 5(b), where the threshold (Purple) automatically trends upward in correlation

with the model's increasing competence. This acts as a homeostatic regulator, anchoring the pass rate (Orange) to the target ratio. By establishing this implicit self-pacing curriculum, DGS ensures robust stability and continuous learning from high-value trajectories without manual hyperparameter tuning.

Robustness to Initialization Strategies. Finally, we investigate the role of SFT initialization and whether Sync-R1 requires a specific complex cold-start recipe such as UniCTokens [An et al., 2025a]. Results in Table 3 show that applying Sync-GRPO directly to the base Show-o model without SFT initialization causes severe degradation, indicating that SFT provides an essential bootstrapping foundation. However, Sync-GRPO is not tied to a single elaborate SFT recipe: it consistently improves both the basic Joint-SFT and the stronger UniCTokens-SFT initialization across all evaluated metrics. In particular, Sync-GRPO on Joint-SFT yields a 17.9% improvement in Dense Rea. and an average 13.8% gain in VQA tasks, while the full Sync-R1 pipeline achieves the best overall performance. Thus, SFT supplies the necessary initialization, while Sync-GRPO unlocks higher-level gains in cross-modal alignment and reasoning-intensive personalization.

5 Conclusion

We present **Sync-R1**, a unified framework for synergistic co-optimization of personalized understanding and generation. By establishing an explicit reasoning loop via **Sync-GRPO**, Sync-R1 bridges comprehension and creation within a closed-loop reinforcement learning process, enabling the two tasks to mutually reinforce each other. This helps the model move beyond simple pattern matching toward deeper reasoning over dense user contexts and high-fidelity personalized generation. Experiments on **UnifyBench++** show that Sync-R1 significantly outperforms existing baselines, especially in reasoning-intensive and information-dense scenarios. In addition, **Dynamic Group Scaling (DGS)** mitigates the computational cost of multimodal RL by accelerating convergence through high-potential trajectory filtering. Overall, this work offers a robust and efficient RL-based paradigm for aligning multimodal understanding and generation in personalized systems.

Limitations

Sync-R1 personalizes a model to individual concepts using a small set of reference images and textual descriptions. We have not tested it in settings involving long-term user interactions or changing preferences. We also do not evaluate conflicting contextual information. The current method has no explicit mechanism for deciding which information to follow when the provided descriptions contradict each other.

Acknowledgments and Disclosure of Funding

Funding. This work used computing and research resources provided internally by our research group. No external funding or third-party support was received for this work.

Competing interests. The authors declare no competing financial interests related to this work.

References

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm, 2024a. URL https://arxiv.org/abs/2309.05519.

Jingjing Jiang, Chongjie Si, Jun Luo, Hanwang Zhang, and Chao Ma. Co-reinforcement learning for unified multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2505.17534.

Yanghao Li, Rui Qian, Bowen Pan, Haotian Zhang, Haoshuo Huang, Bowen Zhang, Jialing Tong, Haoxuan You, Xianzhi Du, Zhe Gan, Hyunjik Kim, Chao Jia, Zhenbang Wang, Yinfei Yang, Mingfei Gao, Zi-Yi Dou, Wenze Hu, Chang Gao, Dongxu Li, Philipp Dufter, Zirui Wang, Guoli Yin, Zhengdong Zhang, Chen Chen, Yang Zhao, Ruoming Pang, and Zhifeng Chen. Manzano: A simple and scalable unified multimodal model with a hybrid vision tokenizer, 2025. URL https://arxiv.org/abs/2509.16197.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025a.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining, 2025. URL https://arxiv.org/abs/2505.14683.

Yu Zhong, Tianwei Lin, Ruike Zhu, Yuqian Yuan, Haoyu Zheng, Liang Liang, Wenqiao Zhang, Feifei Shao, Haoyuan Li, Wanggui He, Hao Jiang, and Yueting Zhuang. Unified personalized understanding, generating and editing, 2026. URL https://arxiv.org/abs/2601.06965.

Thao Nguyen, Krishna Kumar Singh, Jing Shi, Trung Bui, Yong Jae Lee, and Yuheng Li. Yochameleon: Personalized vision and language generation, 2025. URL https://arxiv.org/abs/2504.20998.

Rui Tian, Mingfei Gao, Mingze Xu, Jiaming Hu, Jiasen Lu, Zuxuan Wu, Yinfei Yang, and Afshin Dehghan. Unigen: Enhanced training & test-time strategies for unified multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2505.14682.

Sen Ye, Mengde Xu, Shuyang Gu, Di He, Liwei Wang, and Han Hu. Understanding vs. generation: Navigating optimization dilemma in multimodal models, 2026. URL https://arxiv.org/abs/2602.15772.

Ruichuan An, Sihan Yang, Renrui Zhang, Zijun Shen, Ming Lu, Gaole Dai, Hao Liang, Ziyu Guo, Shilin Yan, Yulin Luo, Bocheng Zou, Chaoqun Yang, and Wentao Zhang. Unictokens: Boosting personalized understanding and generation via unified concept tokens, 2025a. URL https://arxiv.org/abs/2505.14671.

Gustavo Penha, Edoardo D'Amico, Marco De Nadai, Enrico Palumbo, Alexandre Tamborrino, Ali Vardasbi, Max Lefarov, Shawn Lin, Timothy Heath, Francesco Fabbri, and Hugues Bouchard. Semantic ids for joint generative search and recommendation, 2025. URL https://arxiv.org/abs/2508.10478.

Jierun Chen, Tiezheng Yu, Haoli Bai, Lewei Yao, Jiannan Wu, Kaican Li, Fei Mi, Chaofan Tao, Lei Zhu, Manyi Zhang, Xiaohui Li, Lu Hou, Lifeng Shang, and Qun Liu. The synergy dilemma of long-cot sft and rl: Investigating post-training techniques for reasoning vlms, 2025. URL https://arxiv.org/abs/2507.07562.

Qian Liang, Yujia Wu, Kuncheng Li, Jiwei Wei, Shiyuan He, Jinyu Guo, and Ning Xie. Mm-r1: Unleashing the power of unified multimodal large language models for personalized image generation. *arXiv preprint arXiv:2508.11433*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL https://arxiv.org/abs/2204.05862.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model, 2025. URL https://arxiv.org/abs/2503.24290.

Ruichuan An, Sihan Yang, Ziyu Guo, Wei Dai, Zijun Shen, Haodong Li, Renrui Zhang, Xinyu Wei, Guopeng Li, Wenshan Wu, and Wentao Zhang. Genius: Generative fluid intelligence evaluation suite, 2026. URL https://arxiv.org/abs/2602.11144.

Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. What's behind ppo's collapse in long-cot? value optimization holds the secret, 2025. URL https://arxiv.org/abs/2503.01491.

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, Xiangpeng Wei, Xiangyu Yu, Gaohong Liu, Juncai Liu, Lingjun Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Ru Zhang, Xin Liu, Mingxuan Wang, Yonghui Wu, and Lin Yan. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks, 2025. URL https://arxiv.org/abs/2504.05118.

Ethan Chern, Zhulin Hu, Steffi Chern, Siqi Kou, Jiadi Su, Yan Ma, Zhijie Deng, and Pengfei Liu. Thinking with generated images, 2025. URL https://arxiv.org/abs/2505.22525.

Jiadong Pan, Zhiyuan Ma, Kaiyan Zhang, Ning Ding, and Bowen Zhou. Self-reflective reinforcement learning for diffusion-based image reasoning generation, 2025a. URL https://arxiv.org/abs/2505.22407.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Chengzhuo Tong, Ziyu Guo, Renrui Zhang, Wenyu Shan, Xinyu Wei, Zhenghao Xing, Hongsheng Li, and Pheng-Ann Heng. Delving into rl for image generation with cot: A study on dpo vs. grpo, 2025. URL https://arxiv.org/abs/2505.17017.

Longzhen Han, Awes Mubarak, Almas Baimagambetov, Nikolaos Polatidis, and Thar Baker. A survey of generative categories and techniques in multimodal large language models, 2025. URL https://arxiv.org/abs/2506.10016.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. Dancegrpo: Unleashing grpo on visual generation, 2025. URL https://arxiv.org/abs/2505.07818.

Yicheng Xiao, Lin Song, Yukang Chen, Yingmin Luo, Yuxin Chen, Yukang Gan, Wei Huang, Xiu Li, Xiaojuan Qi, and Ying Shan. Mindomni: Unleashing reasoning generation in vision language models with rgpo, 2025. URL https://arxiv.org/abs/2505.13031.

Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning, 2025. URL https://arxiv.org/abs/2505.23380.

Yu Zhang, Yunqi Li, Yifan Yang, Rui Wang, Yuqing Yang, Dai Qi, Jianmin Bao, Dongdong Chen, Chong Luo, and Lili Qiu. Reasongen-r1: Cot for autoregressive image generation models through sft and rl, 2025a. URL https://arxiv.org/abs/2505.24875.

Yuanhong Zheng, Ruichuan An, Xiaopeng Lin, Yuxing Liu, Sihan Yang, Huanyu Zhang, Haodong Li, Qintong Zhang, Renrui Zhang, Guopeng Li, Yifan Zhang, Yuheng Li, and Wentao Zhang. Pearl: Personalized streaming video understanding model, 2026. URL https://arxiv.org/abs/2603.20422.

Junyu Feng, Binxiao Xu, Jiayi Chen, Mengyu Dai, Cenyang Wu, Haodong Li, Bohan Zeng, Yunliu Xie, Hao Liang, Ming Lu, and Wentao Zhang. M2a: Multimodal memory agent with dual-layer hybrid memory for long-term personalized interactions, 2026. URL https://arxiv.org/abs/2602.07624.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation, 2023. URL https://arxiv.org/abs/2208.12242.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. URL https://arxiv.org/abs/2208.01618.

Renjie Pi, Jianshu Zhang, Tianyang Han, Jipeng Zhang, Rui Pan, and Tong Zhang. Personalized visual instruction tuning, 2024. URL https://arxiv.org/abs/2410.07113.

Ruichuan An, Kai Zeng, Ming Lu, Sihan Yang, Renrui Zhang, Huitong Ji, Hao Liang, and Wentao Zhang. Concept-as-tree: A controllable synthetic data framework makes stronger personalized vlms, 2025b. URL https://arxiv.org/abs/2503.12999.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2025. URL https://arxiv.org/abs/2405.09818.

Sihan Yang, Huitong Ji, Shaolin Lu, Jiayi Chen, Binxiao Xu, Ming Lu, Yuanxing Zhang, Wenhui Dong, and Wentao Zhang. Small-large collaboration: Training-efficient concept personalization for large vlm using a meta personalized small vlm, 2025. URL https://arxiv.org/abs/2508.07260.

Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, Weixin Liu, Zhihua Wu, Weibao Gong, Jianzhong Liang, Zhizhou Shang, Peng Sun, Wei Liu, Xuan Ouyang, Dianhai Yu, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation, 2021. URL https://arxiv.org/abs/2107.02137.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. URL https://arxiv.org/abs/2301.12597.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation, 2025b. URL https://arxiv.org/abs/2408.12528.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer, 2022. URL https://arxiv.org/abs/2202.04200.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective, 2025a. URL https://arxiv.org/abs/2503.20783.

Vaishnavi Shrivastava, Ahmed Awadallah, Vidhisha Balachandran, Shivam Garg, Harkirat Behl, and Dimitris Papailiopoulos. Sample more to think less: Group filtered policy optimization for concise reasoning, 2025. URL https://arxiv.org/abs/2508.09726.

Ruichuan An, Sihan Yang, Ming Lu, Renrui Zhang, Kai Zeng, Yulin Luo, Jiajun Cao, Hao Liang, Ying Chen, Qi She, Shanghang Zhang, and Wentao Zhang. Mc-llava: Multi-concept personalized vision-language model, 2025c. URL https://arxiv.org/abs/2411.11706.

Yuval Alaluf, Elad Richardson, Sergey Tulyakov, Kfir Aberman, and Daniel Cohen-Or. Myvlm: Personalizing vlms for user-specific queries, 2024. URL https://arxiv.org/abs/2403.14599.

Xiangxiang Chu, Hailang Huang, Xiao Zhang, Fei Wei, and Yong Wang. Gpg: A simple and strong reinforcement learning baseline for model reasoning, 2025. URL https://arxiv.org/abs/2504.02546.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models, 2025. URL https://arxiv.org/abs/2503.06749.

Jiazhen Pan, Che Liu, Junde Wu, Fenglin Liu, Jiayuan Zhu, Hongwei Bran Li, Chen Chen, Cheng Ouyang, and Daniel Rueckert. Medvlm-r1: Incentivizing medical reasoning capability of vision-language models (vlms) via reinforcement learning, 2025b. URL https://arxiv.org/abs/2502.19634.

Hengguang Zhou, Xirui Li, Ruochen Wang, Minhao Cheng, Tianyi Zhou, and Cho-Jui Hsieh. R1-zero's "aha moment" in visual reasoning on a 2b non-sft model, 2025. URL https://arxiv.org/abs/2503.05132.

Yuqi Liu, Bohao Peng, Zhisheng Zhong, Zihao Yue, Fanbin Lu, Bei Yu, and Jiaya Jia. Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement, 2025b. URL https://arxiv.org/abs/2503.06520.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning, 2025c. URL https://arxiv.org/abs/2503.01785.

Xingjian Zhang, Siwei Wen, Wenjun Wu, and Lei Huang. Tinyllava-video-r1: Towards smaller lmms for video reasoning, 2025b. URL https://arxiv.org/abs/2504.09641.

Guangzhi Sun, Yudong Yang, Jimin Zhuang, Changli Tang, Yixuan Li, Wei Li, Zejun MA, and Chao Zhang. video-salmonn-o1: Reasoning-enhanced audio-visual large language model, 2025. URL https://arxiv.org/abs/2502.11775.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code, 2024. URL https://arxiv.org/abs/2403.07974.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems?, 2024a. URL https://arxiv.org/abs/2403.14624.

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Shicheng Li, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, Peng Gao, Chunyuan Li, and Hongsheng Li. Mavis: Mathematical visual instruction tuning with an automatic data engine, 2024b. URL https://arxiv.org/abs/2407.08739.

Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model, 2023. URL https://arxiv.org/abs/2307.08041.

Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, Xiangwen Kong, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Dreamllm: Synergistic multimodal comprehension and creation, 2024. URL https://arxiv.org/abs/2309.11499.

Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. Seed-x: Multimodal models with unified multi-granularity comprehension and generation, 2025. URL https://arxiv.org/abs/2404.14396.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation, 2024b. URL https://arxiv.org/abs/2410.13848.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model, 2024. URL https://arxiv.org/abs/2408.11039.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. Emu3: Next-token prediction is all you need, 2024. URL https://arxiv.org/abs/2409.18869.

Bohan Wang, Zhongqi Yue, Fengda Zhang, Shuo Chen, Li'an Bi, Junzhe Zhang, Xue Song, Kennard Yanting Chan, Jiachun Pan, Weijia Wu, Mingze Zhou, Wang Lin, Kaihang Pan, Saining Zhang, Liyu Jia, Wentao Hu, Wei Zhao, and Hanwang Zhang. Selftok: Discrete visual tokens of autoregression, by diffusion, and for reasoning, 2025. URL https://arxiv.org/abs/2505.07538.

Yulin Luo, Ruichuan An, Bocheng Zou, Yiming Tang, Jiaming Liu, and Shanghang Zhang. Llm as dataset analyst: Subpopulation structure discovery with large language model, 2024. URL https://arxiv.org/abs/2405.02363.

Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E. Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation, 2023. URL https://arxiv.org/abs/2305.16289.

Rongyao Fang, Shilin Yan, Zhaoyang Huang, Jingqiu Zhou, Hao Tian, Jifeng Dai, and Hongsheng Li. Instructseq: Unifying vision tasks with instruction-conditioned multi-modal sequence generation, 2023. URL https://arxiv.org/abs/2311.18835.

Shengbang Tong, David Fan, Jiachen Zhu, Yunyang Xiong, Xinlei Chen, Koustuv Sinha, Michael Rabbat, Yann LeCun, Saining Xie, and Zhuang Liu. Metamorph: Multimodal understanding and generation via instruction tuning, 2024. URL https://arxiv.org/abs/2412.14164.

Lijie Fan, Tianhong Li, Siyang Qin, Yuanzhen Li, Chen Sun, Michael Rubinstein, Deqing Sun, Kaiming He, and Yonglong Tian. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens, 2024. URL https://arxiv.org/abs/2410.13863.

Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, October 2016. ISSN 1558-2361. doi: 10.1109/lsp.2016.2603342. URL http://dx.doi.org/10.1109/LSP.2016.2603342.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 815–823. IEEE, June 2015. doi: 10.1109/cvpr.2015.7298682. URL http://dx.doi.org/10.1109/CVPR.2015.7298682.

Mei Wang and Weihong Deng. Deep face recognition: A survey. *Neurocomputing*, 429:215–244, 2021. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2020.10.081. URL https://www.sciencedirect.com/science/article/pii/S0925231220316945.

Thao Nguyen, Haotian Liu, Yuheng Li, Mu Cai, Utkarsh Ojha, and Yong Jae Lee. Yo'llava: Your personalized language and vision assistant, 2024. URL https://arxiv.org/abs/2406.09400.

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Xin Wang, Rachel Ward, Yue Wu, Dingli Yu, Cyril Zhang, and Yi Zhang. Phi-4 technical report, 2024. URL https://arxiv.org/abs/2412.08905.

Wei-Lin Chiang, Zhuohan Li, Ziqing Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6, 2023.

Haoran Hao, Jiaming Han, Changsheng Li, Yu-Feng Li, and Xiangyu Yue. Rap: Retrieval-augmented personalization for multimodal large language models, 2025. URL https://arxiv.org/abs/2410.13360.

A Other Related Work

Reinforcement Learning in Large Models. Within the LLM domain, recent studies [Chu et al., 2025] have made significant strides in addressing critical challenges such as long-chain reasoning, output coherence, and training efficiency. Following the introduction of the GRPO strategy and the demonstration of rule-based rewards in DeepSeek-R1 [DeepSeek-AI et al., 2025], there has been a surge of research applying these principles to Multimodal LLMs (MLLMs) [Huang et al., 2025, Pan et al., 2025b, Zhou et al., 2025]. This expansion spans diverse applications, including semantic segmentation [Liu et al., 2025b], object recognition [Liu et al., 2025c], video analysis [Zhang et al., 2025b, Sun et al., 2025], code generation [Austin et al., 2021, Jain et al., 2024], and mathematical problem-solving [Hendrycks et al., 2021, Zhang et al., 2024a,b]. Ample evidence suggests that RL not only enhances in-domain reasoning capabilities but also yields superior performance compared to Supervised Fine-Tuning (SFT) in out-of-distribution (OOD) scenarios.

Unifying Understanding and Generation. Ideally, a single model should seamlessly handle both understanding and generation tasks. To this end, numerous studies have explored unified architectures. For example, Ge et al. [2023], Dong et al. [2024], Ge et al. [2025] aggregate language-conditioned information to drive generation, while Janus [Wu et al., 2024b] models different modalities using distinct tokenizers. Show-o [Xie et al., 2025b] and Transfusion [Zhou et al., 2024] combine autoregressive and diffusion methods to process text and images, whereas Emu3 [Wang et al., 2024] and Selftok [Wang et al., 2025] unify multimodal data into discrete tokens for joint training. In personalization scenarios, the cross-modal potential of unified models inspires the pursuit of complementary advantages. While early attempts serially combined understanding and generation models [Luo et al., 2024, Dunlap et al., 2023], they faced challenges in end-to-end optimization. Subsequent efforts adopted shared representations to jointly optimize understanding and generation [Fang et al., 2023, Tong et al., 2024, Fan et al., 2024]. ULM-R1 [Jiang et al., 2025] jointly trains general understanding and generation without using the understanding output to condition generation. MM-R1 [Liang et al., 2025] uses understanding to guide zero-shot personalized generation. And our work, Sync-R1, explicitly couples personalized understanding and generation within a unified reasoning trajectory, jointly optimizing both stages with understanding and generation rewards.

B Details of Dataset

Building upon the foundation of UnifyBench [An et al., 2025a], we introduce **UnifyBench++**, an augmented benchmark designed to rigorously evaluate advanced reasoning and generation capabilities. As illustrated in Figure 6, each concept contains stable descriptive information in `info` and supplementary contextual attributes in `extra_info`, paired with reasoning prompts. Stable concept knowledge is encoded in the learned concept tokens, whereas `extra_info` serves as an external information list for the textual-attribute pathway. The italicized entries denote the attribute–prompt examples used for training, and the non-italicized entries denote evaluation examples. The evaluation attributes are supplied as external context when answering their associated prompts. Thus, the held-out split evaluates the ability to select and ground relevant attributes for examples not used for training. Furthermore, to assess robustness in high-complexity settings, we constructed three specialized subsets. Dense Generation (`Dense Gen.`) targets the generative domain, challenging the model to maintain fidelity under high information load. Dense Reasoning (`Dense Rea.`) focuses on the understanding domain, evaluating fine-grained comprehension capabilities in complex scenarios. Crucially, Dense Reasoning Generation (`Dense Rea. Gen.`) bridges all these modalities by assessing the model's ability to synthesize visual content contingent upon attributes inferred from the understanding process within a dense-prompt context.

C Theoretical Formulation of Sync-GRPO

Foundational Strategy. We build our optimization framework upon the **Dr.GRPO** strategy [Liu et al., 2025a], which mitigates critical optimization biases inherent in standard GRPO. Dr.GRPO fundamentally diverges from the standard paradigm through two synergistic modifications: First, it discards the variance normalization term (σ). Instead of $\hat{A}_i = (R_i - \mu)/\sigma$, it computes advantage via strict group centering: $\hat{A}_i = R_i - \frac{1}{G}\sum_{k=1}^{G} R_k$. This preserves the absolute magnitude of reward

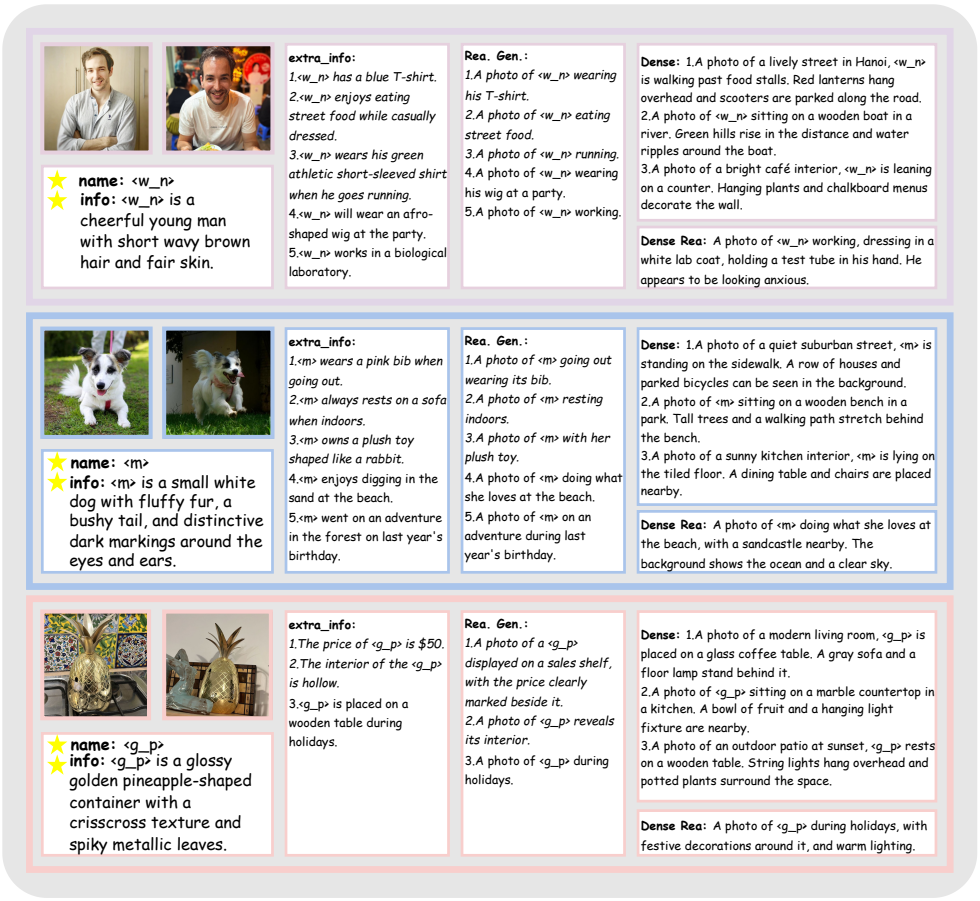


Figure 6: **Snapshot of UnifyBench++.** We present illustrative examples of concept entries within our constructed dataset. The *italicized segments* within the `extra_info` and `Rea. Gen.` columns denote the specific data subsets utilized for training Sync-R1, while the non-italicized text is reserved for evaluation.

differences, ensuring gradients accurately reflect semantic difficulty and preventing noise amplification in low-variance groups. Besides, it eliminates the token-level length normalization $(1/|o_i|)$ from the objective. By treating the generation trajectory as a holistic unit (via summation $\sum_{j=1}^{|o_i|}$ rather than averaging), Dr.GRPO effectively curtails the model's tendency to exploit reward structures through verbosity, ensuring concise alignment with prompt constraints. Dr.GRPO effectively curtails the model's inherent tendency to exploit underlying reward structures through excessive verbosity, ultimately ensuring concise alignment with complex user prompt constraints.

The Optimization Objective of Sync-GRPO. To adapt this foundation to our unified multimodal setting—where the model must simultaneously perform intermediate reasoning (Understanding) and visual synthesis (Generation)—we propose **Sync-GRPO**. This formulation uses modality-specific weighting coefficients to balance the distinct gradient contributions of the understanding and genera-

tion phases. The final optimization objective is formulated as:

$$\mathcal{J}_{\text{Sync-GRPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{G}\sum_{i=1}^{G}\sum_{j=1}^{|o_i|} \alpha_j \left(\min\left(D_{i,j}(\theta)\hat{A}_i, \text{clip}\left(D_{i,j}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_i \right) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}}) \right) \right] \tag{9}$$

Here, the task-specific weighting coefficient α_j is defined piecewise to modulate the learning focus across the sequence:

$$\alpha_j = \begin{cases} \alpha_{\text{text}}, & j \in \text{Tokens in } IR_i \\ \alpha_{\text{image}}, & j \in \text{Image Tokens} \end{cases} \tag{10}$$

where IR_i denotes the length of the intermediate reasoning segment (Understanding Task), and the subsequent tokens correspond to the generation phase. By tuning α_{text} and α_{image}, Sync-GRPO ensures robust joint optimization, preventing one modality from dominating the learning process while maintaining the rigorous constraints of Dr.GRPO.

D Details and Analysis of the Synergistic Reward Ensemble

The effectiveness of **Sync-GRPO** relies on a multi-faceted reward ensemble that provides dense feedback for both discrete reasoning and high-fidelity generation. This section first summarizes the functional role and activation scope of each reward component, then details their mathematical formulations, analyzes the calibration and sensitivity of the reward weights, and further examines reward dynamics and leave-one-out balance during training. These analyses show that the proposed ensemble provides balanced supervision across understanding, alignment, identity preservation, and facial fidelity, leading to stable optimization without relying on a single dominant reward signal.

D.1 Functional Roles of Reward Components

Table 4 summarizes the role, input, target behavior, and activation scope of each reward component in the Sync-GRPO trajectory. TIER supervises the understanding phase by encouraging the model to select and ground relevant personalized concept information. BER, DER, and FER supervise the generation phase from complementary perspectives: prompt-image semantic alignment, concept identity preservation, and fine-grained facial fidelity.

The reward weights are assigned at a coarse category level rather than tuned for each individual concept. Specifically, FER is activated only for human subjects, where facial identity cues are essential, while animals and objects use a larger share of BER and DER to emphasize semantic alignment and visual identity preservation. This design keeps the reward ensemble simple while adapting supervision to the main source of fidelity required by each category.

D.2 Reward Component Formulations

We employ four distinct reward experts to supervise the multimodal optimization process by providing fine-grained feedback, covering logical consistency, cross-modal alignment, structural identity, and facial fidelity.

Text Inference Evaluation Reward (TIER). To evaluate the logical consistency of the understanding phase, we utilize ERNIE 3.0 [Sun et al., 2021] to compute semantic embeddings. Let v_{IR} be the embedding of the model's intermediate reasoning output, and $\{v_{E_i}\}_{i=1}^{l}$ be the embeddings of l candidate conceptual information segments. We first compute a similarity vector $\mathbf{s} = [s_1, \ldots, s_l]$, where $s_i = \cos(v_{IR}, v_{E_i})$. The final TIER is derived from the Euclidean distance d between $\mathbf{s}$ and the binary ground-truth vector $\mathbf{y} \in \{0,1\}^l$:

$$R_{\text{TIER}} = \frac{1}{2}(2 - \|\mathbf{s} - \mathbf{y}\|_2) \tag{11}$$

Since $\|\mathbf{s} - \mathbf{y}\|_2$ is bounded within $[0, 2]$, R_{TIER} is normalized to $[0, 1]$.

Table 4: **Summary of the reward components in Sync-R1.** Each reward supervises a different aspect of the joint reasoning-generation trajectory. TIER is used for the understanding phase, whereas BER, DER, and FER evaluate generation fidelity. The weights are coarse category-level defaults rather than per-concept tuning.

Component	TIER	BER	DER	FER
Supervises	Understanding phase	Generation phase	Generation phase	Generation phase
Role	Understanding	Semantic Alignment	Identity Preservation	Facial Fidelity
Input	Intermediate reasoning output & candidate concept information	Text prompt & generated image	Generated image & reference image	Aligned face crops from generated and reference images
Target behavior	Encourage correct selection and grounding of relevant personalized concept information	Enforce prompt-image consistency and reduce semantic drift	Preserve concept-specific visual identity	Preserve subtle facial identity cues
Activated category	All categories	All categories	All categories	Human only
Weight (Human subjects)	0.4	0.2	0.2	0.2
Weight (Animals/Objects)	0.4	0.3	0.3	0.0

BLIP Evaluation Reward (BER). To ensure cross-modal semantic alignment, we leverage BLIP-2 [Li et al., 2023] as a vision-language judge. Let P be the text prompt and I_{gen} be the generated image. BER measures the similarity:

$$R_{\text{BER}} = \cos(\text{Embtext}(P), \text{Embimg}(I_{\text{gen}})) \tag{12}$$

This reward penalizes semantic drift, ensuring that the generated visual content strictly adheres to the provided prompt.

DINOv2 Evaluation Reward (DER). For structural identity preservation, we utilize DINOv2 [Oquab et al., 2024], which is sensitive to fine-grained textural and structural features. DER is calculated as the cosine similarity between the features of the generated image I_{gen} and the reference image I_{ref}:

$$R_{\text{DER}} = \cos(\Phi_{\text{DINO}}(I_{\text{gen}}), \Phi_{\text{DINO}}(I_{\text{ref}})) \tag{13}$$

This ensures that the personalized concept maintains its unique visual identity across different generated contexts.

FaceNet Evaluation Reward (FER). To achieve high-fidelity facial synthesis, we implement a dedicated facial enhancement reward. We first employ MTCNN [Zhang et al., 2016] for face alignment and cropping, followed by FaceNet [Schroff et al., 2015, Wang and Deng, 2021] to extract facial embeddings. The reward is defined as:

$$R_{\text{FER}} = \cos(\text{Embface}(I_{\text{gen}}), \text{Embface}(I_{\text{ref}})) \tag{14}$$

This specific supervisor ensures the preservation of subtle facial features essential for human-centric personalization.

Table 5: **Ablation study of reward ensembles.** We compare different combinations of TIER, BER, DER, and FER across all the tasks. FER is used only for human concepts in the full ensemble.

Reward Ensembles	Und.							Gen.							
	Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
	Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
TIER+BER	0.838	0.245	0.494	0.582	0.583	0.594	0.644	0.298	0.739	0.329	0.636	0.311	0.784	0.326	0.743
TIER+DER	0.849	0.239	0.487	0.593	0.588	0.594	0.639	0.297	0.733	0.323	0.638	0.306	0.779	0.321	0.738
TIER+BER+DER	0.852	0.252	0.491	0.585	0.591	0.603	0.631	0.306	0.748	0.327	0.640	0.318	0.787	0.331	0.741
TIER+BER+DER+FER	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

Table 6: **Sensitivity of reward-weight allocation.** We perturb the weight $w_{\text{Und.}}$ by $\pm 15\%$ around the default $0.40 : 0.60$ setting and redistribute the remaining generation weight uniformly. All relative changes remain below 1%, indicating robustness to moderate coefficient perturbations.

Weight Setting ($w_{\text{Und.}} : w_{\text{Gen.}}$)	Und.					Gen.			
	Rec.	Rea.*	Dense Rea.*	VQA	QA	Pure Gen.	Dense Gen.*	Rea. Gen.*	Dense Rea. Gen.*
	Weight	BLEU	GPT	Avg. Score	Avg. Score	Avg. Score	Avg. Score	Avg. Score	Avg. Score
+15% Und. (0.46:0.54)	0.853 (-0.698%)	0.252 (+0.800%)	0.499 (-0.795%)	0.599 (0.000%)	0.630 (+0.318%)	0.535 (-0.372%)	0.489 (-0.407%)	0.565 (+0.355%)	0.557 (+0.360%)
-15% Und. (0.34:0.66)	0.864 (+0.582%)	0.249 (-0.400%)	0.499 (-0.795%)	0.600 (+0.167%)	0.632 (+0.637%)	0.542 (+0.931%)	0.490 (-0.204%)	0.562 (-0.178%)	0.553 (-0.360%)
Sync-R1 (0.40:0.60)	0.859	0.250	0.503	0.599	0.628	0.537	0.491	0.563	0.555

D.3 Reward Ensemble Calibration and Sensitivity

The final unified reward combines one understanding-oriented reward with multiple generation-oriented rewards:

$$R_{\text{total}} = w_1 R_{\text{TIER}} + w_2 R_{\text{BER}} + w_3 R_{\text{DER}} + w_4 R_{\text{FER}}, \quad \sum_{i=1}^{4} w_i = 1. \tag{15}$$

Here, R_{TIER} supervises the personalized reasoning result, while R_{BER}, R_{DER}, and R_{FER} evaluate text-image alignment, identity preservation, and facial fidelity in the generated image, respectively.

Difficulty-aware allocation. Understanding tasks require discrete, concept-level reasoning and are therefore harder to optimize than standard generation rewards. We thus use an aggregate default ratio of $w_{\text{Und.}} : w_{\text{Gen.}} = 0.40 : 0.60$, where $w_{\text{Und.}} = w_1$ and $w_{\text{Gen.}} = w_2 + w_3 + w_4$. The generation budget is distributed over the applicable generation rewards:

$$W = \begin{cases} (0.4, 0.3, 0.3, 0.0), & \text{for Animals/Objects,} \\ (0.4, 0.2, 0.2, 0.2), & \text{for Human Subjects.} \end{cases} \tag{16}$$

Reward-ensemble ablation. Table 5 validates the complementary roles of the reward experts. Adding both BER and DER to TIER consistently improves the balance between prompt-level semantic alignment and identity preservation. The domain-aware full ensemble further brings the strongest overall performance: FER is activated only for human concepts, where facial fidelity is a critical personalization cue, while animals and objects retain the TIER+BER+DER configuration. This indicates that fine-grained facial supervision benefits human-centric generation without disrupting the broader understanding-generation balance.

Weight sensitivity. To test whether the proposed weighting strategy depends on fragile numerical tuning, we perturb the aggregate understanding weight by $\pm 15\%$ around the default setting and uniformly redistribute the remaining budget among the generation rewards. As shown in Table 6, both perturbed settings, $0.46 : 0.54$ and $0.34 : 0.66$, keep all relative changes below 1% compared with the default $0.40 : 0.60$ configuration. This demonstrates that Sync-R1 is robust to moderate coefficient perturbations.

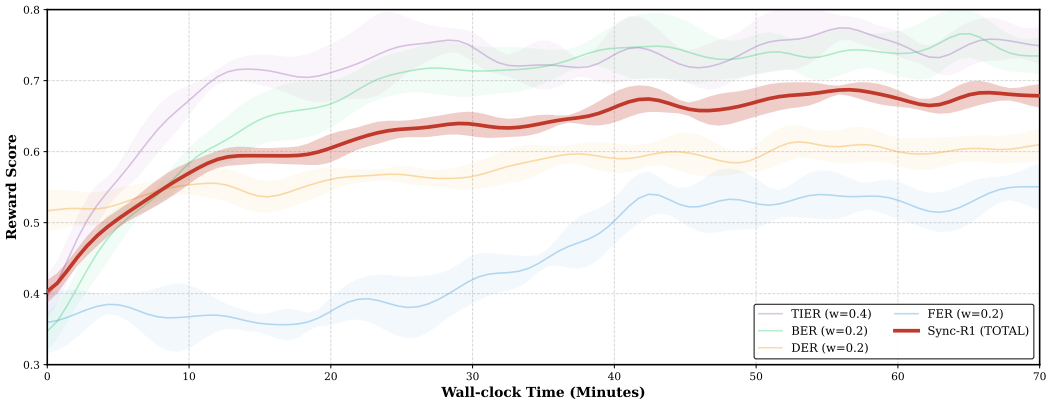


Figure 7: **Training dynamics of individual reward components.** We plot the moving-average trajectories of the reward experts during Sync-R1 training, including TIER, BER, DER, FER, and the aggregated total reward R_{total}. All reward components improve steadily throughout training without evidence of single-reward domination or collapse, suggesting that the joint reward optimization remains stable and does not exhibit obvious reward hacking behavior.

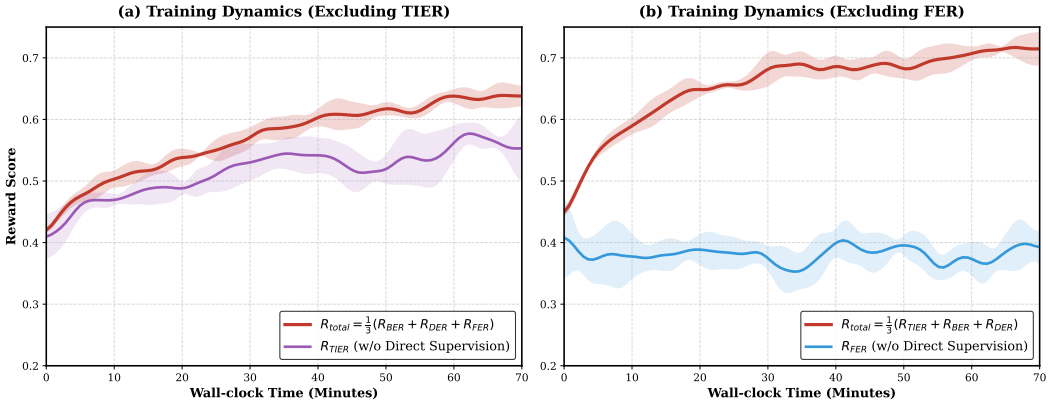


Figure 8: **Held-out reward dynamics under leave-one-out training.** We exclude one reward component from the training objective and monitor its held-out score during optimization. **(a)** When TIER is excluded from optimization, the held-out TIER still improves during training. **(b)** When FER is excluded, the held-out FER remains stable without collapse. This suggests limited reward conflict and no obvious single-reward exploitation.

D.4 Reward Dynamics and Leave-One-Out Balance Analysis

Sync-GRPO does not introduce a separate arbitration module among reward experts. Instead, the different objectives are integrated through a unified trajectory-level reward, where the understanding reward R_{TIER} is assigned an absolute weight of 0.4, larger than any individual generation-side reward. This design prevents the model from bypassing the reasoning stage and exploiting only easier visual reward signals, while still allowing generation-side feedback to refine the final image quality.

Training dynamics of reward components. We visualize reward trajectories on the human concept `<adrien_brody>` to examine whether the reward experts evolve coherently during Sync-GRPO training. As shown in Figure 7, all components and the aggregated Sync-R1 reward increase smoothly over wall-clock time, without one reward improving at the expense of another. This suggests that TIER, BER, DER, and FER provide compatible supervision for the unified reasoning-generation trajectory.

Leave-one-out reward balance. We further conduct leave-one-out training to test whether optimizing the remaining rewards harms a removed objective. Figure 8(a) shows that held-out R_{TIER}

still improves when direct TIER supervision is excluded, suggesting that generation-side rewards do not conflict with understanding. Figure 8(b) shows that held-out R_{FER} remains stable when FER is excluded, indicating that facial-fidelity supervision is complementary rather than competitive. Together, these results support balanced reward optimization without explicit arbitration.

E Theoretical Analysis and Proofs

In this section, we provide the detailed mathematical derivations supporting the effectiveness of Sync-GRPO and Dynamic Group Scaling (DGS). We proceed in a bottom-up manner: first deriving the gradient variance, then proving the variance reduction and SNR amplification properties of DGS, and finally establishing the control stability.

E.1 Gradient Derivation and Variance Decomposition

To justify the focus of DGS on minimizing reward variance, we first explicitly derive the analytical form of the gradient estimator $\hat{g}$ under the clipping mechanism and then decompose its variance.

Step 1: Derivation of the Gradient Estimator $\hat{g}$. Recall the sample-based objective function for Sync-GRPO:

$$\hat{\mathcal{J}}(\theta) = \frac{1}{BG}\sum_{b,i,j} \alpha_j \underbrace{\min\left(D_{i,j}\hat{A}_i, \text{clip}(D_{i,j}, 1-\varepsilon, 1+\varepsilon)\hat{A}_i\right)}_{\mathcal{L}_{i,j}^{\text{CLIP}}(\theta)} - \beta\mathbb{D}_{\text{KL}} \tag{17}$$

We compute the gradient $\nabla_\theta \mathcal{L}_{i,j}^{\text{CLIP}}(\theta)$ by analyzing the behavior of the $\min$ operator. The clipping function restricts $D_{i,j}$ to the interval $[1-\varepsilon, 1+\varepsilon]$. Differentiating with respect to θ:

- **Case 1: Active Region.** The probability ratio is within the trust region, or the update moves $D_{i,j}$ towards the region. In this case, $\min(\cdot) = D_{i,j}\hat{A}_i$. Using the log-derivative trick $\nabla_\theta D_{i,j} = D_{i,j}\nabla_\theta \log \pi_\theta(o_{i,j})$, the gradient is:

$$\nabla_\theta(D_{i,j}\hat{A}_i) = \hat{A}_i \cdot D_{i,j}\nabla_\theta \log \pi_\theta(o_{i,j})$$

- **Case 2: Clipped Region.** The term is clipped to $(1 \pm \varepsilon)\hat{A}_i$. Since $(1 \pm \varepsilon)\hat{A}_i$ is a constant with respect to the current policy parameters θ (locally), its gradient is zero:

$$\nabla_\theta((1 \pm \varepsilon)\hat{A}_i) = 0$$

Let $\mathbb{I}_{i,j}^{\text{active}}$ be the indicator function for the active region. The total policy gradient estimator G_{PG} is:

$$G_{\text{PG}} = \frac{1}{BG}\sum_{b,i,j} \mathbb{I}_{i,j}^{\text{active}} \cdot \alpha_j D_{i,j}\hat{A}_i \nabla_\theta \log \pi_\theta(o_{i,j}) \tag{18}$$

Step 2: Variance Decomposition. Based on Step 1, the total gradient variance is:

$$\text{Var}(\hat{g}) = \text{Var}(G_{\text{PG}} - \beta\nabla\mathbb{D}_{\text{KL}}) \tag{19}$$

Since the KL-divergence term depends only on the policy distribution (which is deterministic given θ) and does not involve the stochastic sampling of rewards, its variance is negligible compared to the Monte Carlo estimator G_{PG}. Thus, we focus on analyzing $\text{Var}(G_{\text{PG}})$.

Step 3: Variance Analysis of the Policy Gradient Term. Consistent with the *linearized policy gradient* regime assumed in Theorem 3.1 of the main text, we omit the clipping mechanism (i.e., setting $\mathbb{I}_{i,j}^{\text{active}} = 1$) for this variance analysis. Let the unclipped gradient contribution of a single sample be:

$$\mathbf{x}_{b,i,j} = \underbrace{(\alpha_j D_{i,j}\nabla_\theta \log \pi_\theta(o_{i,j}))}_{\mathbf{u}_{b,i,j}} \cdot \hat{A}_i \tag{20}$$

We assume that the fluctuations in $\mathbf{u}_{b,i,j}$ are statistically independent of the fluctuations in the advantage estimate $\hat{A}_i$.

Applying the variance of a product for independent variables, and noting that the advantage function is typically centered (i.e., $\mathbb{E}[\hat{A}_i] \approx 0$ due to the baseline), the variance simplifies as follows:

$$\mathrm{Var}(\mathbf{x}_{b,i,j}) \approx \mathbb{E}[\|\mathbf{u}_{b,i,j}\|^2] \cdot \mathrm{Var}(\hat{A}_i) \tag{21}$$

This indicates that the variance of the gradient update is the product of the expected squared norm of $\mathbf{u}_{b,i,j}$ and the variance of the advantage. Consequently, the trace of the variance for the aggregated estimator is:

$$\mathrm{Tr}(\mathrm{Var}[G_{\mathrm{PG}}]) = \frac{1}{(BG)^2} \sum_{b,i,j} \mathbb{E}[\|\mathbf{u}_{b,i,j}\|^2] \cdot \mathrm{Var}(\hat{A}_i) \tag{22}$$

Conclusion. Since $\hat{A}_i = R_i - \text{mean}\left(\{R_i\}_{i=1}^G\right)$ and the baseline mean $\left(\{R_i\}_{i=1}^G\right)$ is deterministic within the group, we have $\mathrm{Var}(\hat{A}_i) = \mathrm{Var}(R_i)$. Based on the assumption in the main text that the rewards R are independent and identically distributed (i.i.d.), we obtain:

$$\mathrm{Tr}(\mathrm{Var}[\hat{g}]) \propto \mathrm{Var}[\hat{A}] \tag{23}$$

This derivation formally proves that reducing the variance of the reward distribution (via DGS) directly minimizes the variance of the optimization update.

E.2 Proof of Variance Reduction (Theorem 3.1)

Assumption. Let the terminal reward R and surrogate $\tilde{R}$ follow a bivariate normal distribution:

$$\begin{pmatrix} R \\ \tilde{R} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_R \\ \mu_{\tilde{R}} \end{pmatrix}, \begin{pmatrix} \sigma_R^2 & \rho\sigma_R\sigma_{\tilde{R}} \\ \rho\sigma_R\sigma_{\tilde{R}} & \sigma_{\tilde{R}}^2 \end{pmatrix} \right) \tag{24}$$

with $\rho > 0$. Let $\mathcal{E} = \{\tilde{R} > T\}$ be the selection event with $P(\mathcal{E}) = G/N$.

Proof. We verify the variance of the selected rewards using the Law of Total Variance:

$$\mathrm{Var}[R \mid \mathcal{E}] = \mathbb{E}[\mathrm{Var}(R \mid \tilde{R}) \mid \mathcal{E}] + \mathrm{Var}(\mathbb{E}[R \mid \tilde{R}] \mid \mathcal{E}) \tag{25}$$

The conditional variance $\mathrm{Var}(R \mid \tilde{R}) = (1-\rho^2)\sigma_R^2$ is constant. Thus, the first term $\mathbb{E}[\mathrm{Var}(R \mid \tilde{R}) \mid \mathcal{E}] = (1-\rho^2)\sigma_R^2$.

For the second term, the conditional expectation is linear: $\mathbb{E}[R \mid \tilde{R}] = \mu_R + \rho\frac{\sigma_R}{\sigma_{\tilde{R}}}(\tilde{R} - \mu_{\tilde{R}})$. Let Y be the random variable $\tilde{R}$ conditioned on $\tilde{R} > T$ (a truncated normal distribution). The variance of this term is:

$$\mathrm{Var}(\mathbb{E}[R \mid \tilde{R}] \mid \mathcal{E}) = \left(\rho\frac{\sigma_R}{\sigma_{\tilde{R}}}\right)^2 \mathrm{Var}(Y) \tag{26}$$

For a standard normal distribution, the variance of the truncated top p-quantile satisfies $Var(Z|Z > z_p) \leq p$ for $p \leq 0.5$. This follows from the fact that the conditional variance increases with p and equals p when $p = 1$. More precisely, for a normal variable $\tilde{R}$ with mean $\mu_{\tilde{R}}$ and variance $\sigma_{\tilde{R}}^2$, we have:

$$\mathrm{Var}(Y) = \sigma_{\tilde{R}}^2 \cdot \mathrm{Var}\left(\frac{\tilde{R} - \mu_{\tilde{R}}}{\sigma_{\tilde{R}}} \middle| \frac{\tilde{R} - \mu_{\tilde{R}}}{\sigma_{\tilde{R}}} > \frac{T - \mu_{\tilde{R}}}{\sigma_{\tilde{R}}} \right) \leq \sigma_{\tilde{R}}^2 \cdot \frac{G}{N} \tag{27}$$

where the inequality holds when $G/N \leq 0.5$ (which is satisfied in our setting with $\frac{G}{N} \approx TPR = 0.25 < \frac{1}{2}$). Specifically, $\mathrm{Var}(Y) \leq \frac{G}{N}\sigma_{\tilde{R}}^2$. Substituting this bound:

$$\mathrm{Var}(\mathbb{E}[R \mid \tilde{R}] \mid \mathcal{E}) \leq \rho^2 \frac{\sigma_R^2}{\sigma_{\tilde{R}}^2} \cdot \frac{G}{N}\sigma_{\tilde{R}}^2 = \rho^2\sigma_R^2\frac{G}{N} \tag{28}$$

Synthesis:

$$\mathrm{Var}[R \mid \mathcal{E}] \leq (1-\rho^2)\sigma_R^2 + \rho^2\frac{G}{N}\sigma_R^2 \tag{29}$$

$$= \sigma_R^2\left(1 - \rho^2\left(1 - \frac{G}{N}\right)\right) \tag{30}$$

This proves the inequality in Theorem 3.1. □

E.3 Proof of SNR Amplification (Corollary 3.2)

The Signal-to-Noise Ratio (SNR) of the gradient estimator is $\mathrm{SNR} = \frac{\|\mathbb{E}[\hat{g}]\|^2}{\mathrm{Tr}(\mathrm{Var}[\hat{g}])}$.

Numerator (Signal): The expected gradient aligns with the expected reward of the selected samples. Under the bivariate normal assumption, $\mathbb{E}[R \mid \mathcal{E}]$ can be expressed exactly using the **Inverse Mills Ratio** $\lambda(\alpha) = \frac{\phi(\alpha)}{1-\Phi(\alpha)} > 0$:

$$\mathbb{E}[R \mid \mathcal{E}] = \mu_R + \rho\sigma_R\lambda\left(\frac{T - \mu_{\tilde{R}}}{\sigma_{\tilde{R}}}\right) \tag{31}$$

Since $\lambda(\cdot)$ is strictly positive, $\mathbb{E}[R \mid \mathcal{E}] > \mu_R$. Thus, the signal magnitude $\|\mathbb{E}[\hat{g}_{\text{DGS}}]\|^2$ is strictly larger than the baseline $\|\mathbb{E}[\hat{g}_{\text{Base}}]\|^2$.

Denominator (Noise): From Appendix E.2, the variance is reduced by $\gamma = 1 - \rho^2(1 - G/N) < 1$.

Result:

$$\mathrm{SNR}_{\text{DGS}} \propto \frac{(\mu_R + \delta)^2}{\gamma\sigma_R^2} > \frac{\mu_R^2}{\sigma_R^2} \cdot \frac{1}{\gamma} = \frac{1}{1 - \rho^2(1 - \frac{G}{N})}\mathrm{SNR}_{\text{Base}} \tag{32}$$

This confirms significant SNR amplification. □

E.4 Stability of Adaptive Threshold Control

We analyze the stability of the trend-aware controller defined in Equation 8. Let $F_{\tilde{R}}(t)$ denote the cumulative distribution function (CDF) of the surrogate reward $\tilde{R}$ at iteration k. The pass rate is $PR_k = 1 - F_{\tilde{R}}(T_k)$. Our goal is for T_k to converge to the value T^* satisfying $PR(T^*) = \text{TPR}$.

Linearized Dynamics. Assuming $F_{\tilde{R}}$ is differentiable with density $f_{\tilde{R}}$, we can linearize around the equilibrium T^*. For small deviations $\delta_k = T_k - T^*$, we have:

$$PR_k \approx \text{TPR} - f_{\tilde{R}}(T^*)\delta_k + \mathcal{O}(\delta_k^2) \tag{33}$$

Thus $\Delta_k = PR_k - \text{TPR} \approx -f_{\tilde{R}}(T^*)\delta_k$.

Taking logarithms of the update rule (ignoring the damping term for simplicity):

$$\ln T_{k+1} = (1 + \mu\Delta_k)\ln T_k \tag{34}$$

Define $x_k = \ln T_k - \ln T^*$. Linearizing yields:

$$x_{k+1} \approx x_k - \mu f_{\tilde{R}}(T^*)T^*x_k = (1 - \mu f_{\tilde{R}}(T^*)T^*)x_k \tag{35}$$

The system is locally stable when $|1 - \mu f_{\tilde{R}}(T^*)T^*| < 1$, i.e., $0 < \mu < \frac{2}{f_{\tilde{R}}(T^*)T^*}$.

Role of Trend-Aware Damping. The full update includes the indicator $\mathbb{I}[\tau_k\Delta_k < 0]$, which reduces the effective gain when the momentum τ_k conflicts with the error direction Δ_k. This adaptive gain scheduling prevents oscillatory divergence when the gradient direction reverses due to batch noise. The momentum term τ_k itself acts as a low-pass filter on threshold changes, smoothing transient fluctuations across the optimization steps.

Global Behavior. While the linear analysis guarantees local stability, the controller's global performance relies on the quasi-stationarity of the reward distribution $F_{\tilde{R}}$. As the policy π_θ improves, the distribution shifts slowly relative to the controller's adaptation rate. The chosen parameters (μ, η, ϵ_0) ensure that the threshold T_k tracks the moving $(1 - \text{TPR})$-quantile, maintaining a consistent flow of high-quality gradients throughout training.

F Empirical Analysis of Dynamic Group Scaling

The theoretical analysis in Appendix E shows that Dynamic Group Scaling (DGS) reduces gradient variance by selecting high-potential trajectories according to an early surrogate reward. We further examine this mechanism empirically through runtime breakdown, truncation-step selection, selection-strategy comparison, and visualization of discarded trajectories.

Table 7: **Wall-clock latency per rejected candidate in DGS.** The intermediate surrogate evaluation adds only 0.32s after the first 10 denoising steps, but allows DGS to skip 9.03s of subsequent decoding and final reward evaluation for low-potential trajectories. All latencies are averaged across 10 samples per concept.

Operation Phase	**Wall-clock Latency**	**Status in DGS (for rejected samples)**
1. Initial Decoding (Steps 0–9)	~ 2.12 s	Executed
2. Intermediate VQ & BLIP Scoring	$\sim$ **0.32 s**	**Executed (Overhead)**
3. Remaining Decoding (Steps 10–49) & Final VQ	~ 8.92 s	Bypassed **(Saved)**
4. Full Reward Eval (BER, DER, FER)	~ 0.11 s	Bypassed **(Saved)**

Table 8: **Effect of truncation step on DGS efficiency and generation quality.** We vary the step used to estimate the surrogate reward $\tilde{R}$. Earlier truncation improves filtering potential but weakens correlation with the terminal reward, while later truncation improves surrogate fidelity at the cost of acceleration. Pearson r measures the correlation between $\tilde{R}$ and the final reward.

Step	**Ratio**	**Pure Gen.**	**Dense Gen.***	**Rea. Gen.***	**Dense Rea. Gen.***	**Avg. Gen.**	**Acceleration Ratio**	**Pearson r**
5	0.10	0.529	0.473	0.535	0.507	0.511	0.9	0.34
10	0.20	0.537	0.491	0.563	0.555	0.536	1.9	0.79
15	0.30	0.540	0.488	0.569	0.556	0.538	1.3	0.82
20	0.40	0.542	0.495	0.568	0.559	0.541	1.1	0.81

Runtime breakdown. We first quantify where the wall-clock acceleration of DGS comes from. For each candidate trajectory, DGS performs early decoding up to step 10 and computes an intermediate surrogate reward $\tilde{R}$ using VQ decoding and BLIP scoring. Candidates below the adaptive threshold are rejected before completing the remaining denoising process and terminal generation-side reward evaluation. As shown in Table 7, the intermediate surrogate evaluation introduces only $\sim$0.32s of overhead, while a rejected trajectory bypasses $\sim$8.92s of remaining decoding and $\sim$0.11s of final reward computation. This confirms that the selection overhead is small relative to the computation saved by filtering low-potential trajectories early.

Surrogate-reward truncation step. DGS relies on a truncated generation prefix to estimate the surrogate reward $\tilde{R}$ before deciding whether a trajectory should be fully evaluated. The truncation step therefore controls a trade-off between selection reliability and computational savings. As shown in Table 8, filtering too early at step 5 yields a weak correlation between $\tilde{R}$ and the final terminal reward, which hurts generation quality and removes the practical speed advantage. Later truncation steps provide slightly stronger reward fidelity, but they consume more generation computation before filtering and therefore reduce acceleration. The adopted setting at step 10, corresponding to $t \approx 0.2T_{\text{total}}$, achieves the best overall compromise: it preserves near-best average generation quality while yielding the strongest wall-clock acceleration.

Selection strategy comparison. We further examine whether DGS benefits from high-potential trajectory selection rather than merely an expanded candidate pool. As shown in Table 9, Bottom-k and the Top/Bottom Mix underperform DGS, with Bottom-k failing to reach the target reward within the training budget. In contrast, Sync-GRPO+DGS preserves or improves the baseline quality while achieving clear acceleration, indicating that high-potential selection is more effective than random retention or mixing low-reward samples.

Visualization of discarded trajectories. Finally, we visualize discarded and retained trajectories to examine what DGS filters out. As shown in Figure 9, discarded samples typically exhibit either severe structural collapse or very low semantic fidelity to the target concept and prompt. In contrast, retained low-quality trajectories remain structurally plausible and semantically related, but miss fine-grained details. This suggests that DGS primarily filters trajectories unlikely to provide stable corrective signals, rather than systematically removing informative hard cases.

Table 9: **Ablation of DGS selection strategies under a fixed compute budget.** All methods retain $k = 9$ trajectories from an expanded candidate pool of $N = 36$ on a representative subset containing one human, one animal, and one object. Acceleration is reported relative to Sync-GRPO. *Ref.* denotes an oracle reference setting, and *Not reached* indicates failure to reach 98% of the baseline peak reward within the maximum training budget.

Selection Strategy	Rea.*	Dense Rea.*	Dense Gen.*	Rea. Gen.*	Dense Rea. Gen.*	Acceleration Ratio
Sync-GRPO	0.269	0.531	0.486	0.571	0.578	1.0×
Full-N Oracle	0.286	0.537	0.527	0.611	0.599	Ref.
Random-k	0.270	0.533	0.483	0.578	0.577	0.9×
Bottom-k	0.264	0.489	0.469	0.555	0.539	Not reached
Top/Bottom Mix	0.271	0.524	0.477	0.569	0.582	0.7×
Sync-GRPO+DGS	0.281	0.536	0.503	0.587	0.583	1.8×



Figure 9: **What does DGS actually discard?** For three representative concepts, we compare the reference image, a trajectory discarded by DGS, a retained low-quality trajectory, and a retained high-quality trajectory. Discarded samples typically show either severe structural collapse or very low semantic fidelity, whereas retained low-quality samples remain structurally plausible but lack fine-grained details. This suggests that DGS mainly filters trajectories unlikely to provide stable corrective signals.

G Experimental Implementation and Additional Analysis

Hyperparameters and Training Details. We first complete the first two-stage training of UniCTokens [An et al., 2025a], setting the number of learnable tokens as $K = 16$ and $M = 8$, and training for 15 epochs in each stage. To manage memory efficiency, the batch size is configured to 4 for understanding tasks and 1 for generation tasks. In the subsequent Sync-GRPO stage, we set the group size to $G = 9$ and maintain a balanced 1:1 mixing ratio between Visual Instruct Reasoning and Textual Attribute Reasoning. Detailed hyperparameters are summarized in Table 10.

Baselines. To evaluate the efficacy of our proposed framework, we compare against a diverse set of state-of-the-art baselines, including our direct predecessor, unified systems, and specialized models:

Table 10: Sync-R1 training hyperparameters.

Name	Value	Name	Value
Group Size G	9	Learning rate lr	1e-6
β	0.01	Classifier-Free Guidance Scale	5
Batchsize	1	Max Gradient Norm	1.0
Training Steps	100	Image Resolution $h \times w$	512×512
α_{text}	0.4	α_{image}	0.6
TPR	0.25	μ	0.12
η	0.8	ε_0	2

- **UniCTokens** [An et al., 2025a]: As our foundational architecture, this method employs a multi-stage training protocol. We compare against the original model trained via its standard pipeline to isolate and verify the performance gains contributed by our proposed Sync-GRPO stage.
- **Yo'chameleon** [Nguyen et al., 2025]: A recent unified model capable of personalized understanding and generation. We retrain this model following the specifications in its original study, utilizing a 7B-parameter base model and 1,000 training images per concept to ensure fair comparison.
- **Yo'LLaVA** [Nguyen et al., 2024]: A unified framework addressing personalization via VLMs. Following the original protocol, we train Yo'LLaVA with varying backbone scales (Phi-1.5 [Abdin et al., 2024], 1.3B and Vicuna [Chiang et al., 2023], 13B) to ensure a comprehensive comparison.
- **MC-LLaVA** [An et al., 2025c]: Designed to enhance multi-concept personalization, this model utilizes paired textual and visual prompts. It stands as a robust baseline for personalized understanding tasks involving complex concept interactions across various challenging visual domains.
- **RAP-MLLM** [Hao et al., 2025]: A retrieval-augmented approach that builds upon RAP-LLaVA. We construct a database for each concept following the RAP-MLLM framework and post-train on a 260K-sample dataset.
- **Textual Inversion** [Gal et al., 2022]: A specialized generative technique that projects visual concepts into the text embedding space, enabling the generation of personalized images via standard text-to-image pipelines.
- **DreamBooth** [Ruiz et al., 2023]: A leading fine-tuning method that adapts generative models to user-specific concepts using a small set of reference images. To rigorously assess performance, we evaluate DreamBooth in both low-data (10 images) and high-data (3,000 images) settings.
- **GPT-4o**: We evaluate GPT-4o on our benchmark to establish an empirical upper bound.

Evaluation Metrics. We adopt a rigorous evaluation protocol covering both personalized understanding and generation tasks. For tasks including personalized recognition, VQA, and standard QA, as well as the fundamental generation Pure Gen. metric in UnifyBench, we strictly follow the calculation protocols defined in UniCTokens to ensure direct comparability. For advanced understanding metrics in UnifyBench++, we adopt distinct evaluation protocols for reasoning tasks. **Rea.** is quantified via the BLEU score, calculated between the model's generated rationale and the ground-truth `extra_info`. For **Dense Rea.**, we employ GPT-4o as an automated judge to assess the logical coherence and factual accuracy of the generated dense reasoning chains. For generation metrics in UnifyBench++, we assess generation quality focusing on visual fidelity and text-image alignment. Visual fidelity is measured across all tasks using CLIP-I similarity between the generated output and the user-provided reference concept image. For prompt alignment, we differentiate by task density: (1) Non-dense tasks (**Rea. Gen.**) utilize CLIP-T scores. Crucially, we substitute abstract reasoning placeholders in the prompt with specific descriptive details before computation to ensure ground-truth alignment. (2) Dense tasks (**Dense Gen.** and **Dense Rea. Gen.**) employ a fine-grained dense alignment score, where the prompt is segmented by punctuation. GPT-4o validates whether each segment is accurately represented in the image, yielding an average validity ratio in $[0, 1]$ to provide a robust quantitative assessment.

Additional Results and Analysis. To validate the robustness of our method beyond our primary benchmarks, we conducted assessments on established datasets specifically designed for pure personalized understanding and generation.

Understanding Performance. Table 11 (left) presents the results on the Yo'LLaVA [Nguyen et al., 2024] and MC-LLaVA [An et al., 2025c] datasets. Note that since our current training paradigm

Table 11: **Performance on personalized understanding and generation benchmarks.** TP = Text Prompt. IP = Image Prompt.

Type	Method	Yo'LLaVA			MC-LLaVA		
		Rec	VQA	QA	Rec	VQA	QA
		Weight	Acc	Acc	Weight	BLEU	Acc
Und. Only	LLaVA+TP	0.807	0.921	0.815	0.608	0.412	0.609
	Yo'LLaVA	0.932	0.918	0.897	0.829	0.657	0.691
	MC-LLaVA	0.953	0.935	0.924	0.901	0.692	0.736
	Qwen2.5-VL+TP	0.685	0.861	0.697	0.634	0.436	0.549
	Yo'LLaVA(Phi-1.5)	0.805	0.627	0.713	0.726	0.525	0.591
Unified Model	Chameleon+TP	0.715	0.537	0.702	0.649	0.408	0.675
	Yo'Chameleon	0.832	0.591	0.734	0.753	0.610	0.658
	Show-o+TP	0.704	0.526	0.605	0.589	0.574	0.482
	UniCTokens	0.865	0.602	0.725	0.742	0.617	0.692
	Sync-R1	0.926	0.663	0.737	0.773	0.671	0.704

Type	Method	Training Images	DreamBench		Yo'LLaVA
			CLIP - I	CLIP - T	CLIP - I
Gen. Only	Real Images	-	0.873	-	0.864
	DreamBooth	∼10	0.692	0.297	0.645
	DreamBooth	∼3000	0.815	0.293	0.788
	Text inversion	∼10	0.675	0.284	0.632
Unified Model	Chameleon+TP	-	0.612	0.168	0.579
	Chameleon+IP	-	0.594	0.171	0.499
	Show-o+TP	-	0.678	0.235	0.652
	Yo'Chameleon	∼1,000	0.807	0.212	0.771
	UniCTokens	∼10	0.788	0.299	0.806
	Sync-R1	∼10	0.808	0.323	0.822

Table 12: **Reliability of LLM-based evaluation.** For task groups originally evaluated by GPT-4o, we additionally report Gemini-2.5-Pro and human scores on sampled outputs. The rankings are highly consistent across judges, and Sync-R1 remains the strongest method under all three evaluations.

Method	Und.									Gen.					
	Dense Rea.*			VQA			QA			Dense Gen.*			Dense Rea. Gen.*		
	GEMINI	GPT	HUMAN	GEMINI	GPT	HUMAN	GEMINI	GPT	HUMAN	GEMINI	GPT	HUMAN	GEMINI	GPT	HUMAN
Yo'LLaVA(Phi-1.5)	0.519	0.489	0.703	0.514	0.497	0.492	0.492	0.504	0.618	-	-	-	-	-	-
Yo'Chameleon	0.429	0.406	0.621	0.541	0.518	0.511	0.592	0.589	0.703	0.344	0.294	0.317	0.304	0.327	0.309
UniCTokens	0.433	0.392	0.608	0.552	0.529	0.523	0.606	0.608	0.721	0.361	0.303	0.327	0.299	0.322	0.302
Sync-R1	0.547	0.512	0.728	0.637	0.614	0.609	0.653	0.659	0.773	0.409	0.345	0.370	0.337	0.361	0.344

focuses on single-concept personalization, we restricted the MC-LLaVA evaluation to the single-concept subset to ensure methodological validity. Our approach consistently outperforms existing state-of-the-art unified models. Notably, Sync-R1 surpasses the previous best method, UniCTokens, by an average margin of **5.3%** across all understanding tasks, demonstrating the efficacy of our reasoning-enhanced optimization in establishing robust multimodal concept alignment.

Generation Performance. We further conducted a comprehensive evaluation on Dreambench [Ruiz et al., 2023] and the generation split of the Yo'LLaVA dataset. Our method exhibits superior performance compared to UniCTokens across all metrics. Most significantly, under identical training data constraints, our unified model outperforms the specialized generative baseline, DreamBooth. This indicates that Sync-GRPO successfully bridges the gap between understanding and generation, enabling the synthesis of context-aware images without compromising model versatility.

H Additional Empirical Analysis and Ablation Studies

H.1 Reliability of LLM-based Evaluation

We further validate the reliability of our LLM-based evaluation protocol in Table 12. For task groups originally evaluated by GPT-4o, we randomly sample 200 instances from each group with more than 200 samples, and additionally collect Gemini-2.5-Pro and human scores on the same outputs.

The rankings are highly consistent across judges. GPT-4o produces exactly the same method ranking as human evaluation on all five overlapping task groups, with Spearman rank correlation $\rho = 1.0$ in each case. Gemini-2.5-Pro also shows strong agreement with human judgments, with $\rho \geq 0.8$. Under all three judges, Sync-R1 remains the best-performing method. These results suggest that our conclusions are robust to the choice of judge and are not an artifact of GPT-4o-based evaluation.

H.2 Comparison with Larger-Capacity Unified Models

To further examine the competitiveness of Sync-R1 against recent larger unified models, we compare it with 7B-scale Chameleon, Janus-Pro, and BAGEL variants on UnifyBench++ in Table 13. When

Table 13: **Comparison with larger-capacity contemporary baselines on Unifybench++.** We compare Sync-R1 with larger 7B-scale baselines, including Chameleon, Janus-Pro, and BAGEL. TP/IP denote text-prompt/image-prompt conditioning when supported by the baseline. Columns with * are newly extended from Unifybench. Best and second best results are highlighted; ties are marked with the same highlight.

Method	Model Size	Und.							Gen.							
		Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
		Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
Chameleon+TP	7B	0.685	0.206	0.313	0.411	0.489	0.509	0.560	0.186	0.542	0.281	0.463	0.193	0.516	0.309	0.549
Chameleon+IP	7B	0.497	0.216	0.374	0.446	0.497	0.411	0.532	0.165	0.514	0.266	0.449	0.190	0.532	0.299	0.503
Janus-Pro+TP	7B	0.822	0.297	0.550	0.492	0.629	0.444	0.650	0.249	0.627	0.387	0.607	0.243	0.650	0.254	0.642
Bagel+TP	7B	0.856	0.278	0.500	0.391	0.641	0.366	0.600	0.284	0.643	0.317	0.628	0.263	0.658	0.349	0.638
Bagel+IP	7B	0.674	0.242	0.550	0.377	0.609	0.380	0.515	0.293	0.690	0.391	0.595	0.285	0.666	0.371	0.630
Sync-R1	1.3B	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

Table 14: **Generalization to a larger unified multimodal backbone.** We apply Sync-R1 to both Show-o-1.3B and Janus-Pro-7B. For Janus-Pro, Sync-R1 denotes Sync-GRPO without DGS, since Janus-Pro uses autoregressive image decoding rather than iterative denoising. Best results within each backbone are highlighted.

Model	Method	Und.							Gen.							
		Rec.	Rea.*	Dense Rea.*	VQA		QA		Pure Gen.		Dense Gen.*		Rea. Gen.*		Dense Rea. Gen.*	
		Weight	BLEU	GPT	BLEU	GPT	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I	CLIP-T	CLIP-I	GPT	CLIP-I
Janus-Pro 7B	Text Prompt	0.822	0.297	0.550	0.492	0.629	0.444	0.650	0.249	0.627	0.387	0.607	0.243	0.650	0.254	0.642
	SFT	0.877	0.323	0.617	0.541	0.773	0.558	0.674	0.285	0.744	0.401	0.685	0.287	0.803	0.306	0.683
	Sync-R1	0.902	0.344	0.621	0.629	0.792	0.622	0.737	0.291	0.809	0.396	0.702	0.331	0.817	0.342	0.779
Show-o 1.3B	Text Prompt	0.562	0.217	0.337	0.462	0.412	0.507	0.579	0.263	0.663	0.299	0.558	0.235	0.679	0.341	0.660
	SFT	0.792	0.238	0.385	0.505	0.521	0.546	0.601	0.280	0.750	0.298	0.639	0.282	0.762	0.317	0.712
	Sync-R1	0.859	0.250	0.503	0.592	0.606	0.604	0.652	0.308	0.765	0.337	0.645	0.324	0.801	0.353	0.756

supported by the baseline, we report both text-prompt and image-prompt conditioning results. Since Janus-Pro does not support image generation conditioned on both text and image inputs, we only report its text-prompt results.

Although Sync-R1 uses a much smaller 1.3B backbone, it achieves the best performance on 10 out of 15 metrics. The gains are especially consistent on QA and generation-side metrics, indicating that the improvement from Sync-GRPO remains competitive even when compared with larger-capacity contemporary unified models.

H.3 Generalization to a Larger Unified Backbone

To examine whether the proposed RL paradigm is tied to a specific backbone, we further apply Sync-R1 to Janus-Pro-7B, a larger tokenized unified multimodal model. Since Janus-Pro adopts autoregressive image decoding rather than iterative denoising, we evaluate Sync-GRPO without the DGS component for this backbone.

As shown in Table 14, Sync-R1 improves over the matched Janus-Pro SFT baseline on 14 out of 15 metrics. The gains are especially clear on reasoning-intensive generation metrics, e.g., Rea. Gen. CLIP-T improves from 0.287 to 0.331, and Dense Rea. Gen. CLIP-I improves from 0.683 to 0.779. These results indicate that the benefits of Sync-GRPO are not specific to Show-o or the 1.3B scale, but can transfer to larger unified multimodal backbones.

H.4 Sensitivity to the Text-Image Balancing Coefficients

We further study the sensitivity of Sync-GRPO to the relative weighting between the text-side and image-side policy optimization terms. As shown in Table 15, assigning too much weight to the image-side term, e.g., $\alpha_{\text{text}} : \alpha_{\text{image}} = 0.2 : 0.8$, weakens reasoning-intensive metrics such as Rea. and Dense Rea. In contrast, overly emphasizing the text-side term, e.g., $0.8 : 0.2$, noticeably reduces generation-side alignment, especially on Rea. Gen. and Dense Rea. Gen. The adopted

Table 15: **Ablation on the $\alpha_{\text{text}} : \alpha_{\text{image}}$ balancing coefficients.** We vary the relative weights of the text-side and image-side policy optimization terms while keeping all other settings fixed.

$\alpha_{\text{text}} : \alpha_{\text{image}}$	Rea.*	Dense Rea.*	Rea. Gen.*		Dense Rea. Gen.*	
	BLEU	GPT	CLIP-T	CLIP-I	GPT	CLIP-I
0.2:0.8	0.221	0.439	0.271	0.797	0.338	0.752
0.4:0.6	**0.250**	**0.503**	**0.324**	**0.801**	**0.353**	**0.756**
0.6:0.4	0.256	0.507	0.319	0.791	0.361	0.747
0.8:0.2	0.243	0.501	0.295	0.732	0.354	0.718

Table 16: **Catastrophic forgetting evaluation.** We compare the base model, UniCTokens-SFT, and Sync-R1 on general-purpose understanding and generation benchmarks. COCO-FID is computed on 1k randomly selected samples.

Method	GenEval	MMMU	POPE	VQAv2 (COCO)	COCO-FID
Base Model	0.67	26.7	80.0	0.72	8.97
UniCTokens-SFT	0.66	26.7	79.8	0.70	8.98
Sync-R1	0.65	26.5	79.8	0.72	9.01

setting $\alpha_{\text{text}} : \alpha_{\text{image}} = 0.4 : 0.6$ provides the best overall balance and achieves the strongest or near-strongest performance across both reasoning and generation metrics.

H.5 Catastrophic Forgetting Evaluation

To examine whether the personalized RL stage compromises the general-purpose capabilities of the base unified model, we evaluate Sync-R1 on a set of standard understanding and generation benchmarks, including GenEval, MMMU, POPE, VQAv2 (COCO), and COCO-FID. For COCO-FID, we randomly sample 1k images for evaluation.

As shown in Table 16, Sync-R1 remains largely unchanged compared with both the base model and UniCTokens-SFT across all benchmarks. These results suggest that the personalized Sync-GRPO stage does not introduce meaningful catastrophic forgetting, while preserving the gains on personalized reasoning and generation reported in the main experiments.

H.6 Controlled Analysis of the Coupling Mechanisms

Directional training ablations. We compare Full coupling with three controlled variants to examine the two coupling directions. Full coupling uses the intermediate reasoning output to condition generation and includes generation rewards in the advantage for reasoning-token updates. *Decoupled RL* disables both IR $\rightarrow$ Gen. and Gen. Reward $\rightarrow$ IR, while retaining RL optimization for both stages. *w/o IR for Gen.* removes the reasoning output from the compound prompt while retaining the shared trajectory-level reward. *w/o Gen. Reward for Und.* retains reasoning-based conditioning but excludes generation rewards from the advantage used in the reasoning-token loss, leaving the image-token objective unchanged. All four settings use the same initialization and training configuration apart from these interventions, with DGS disabled.

As shown in Table 17, both one-way variants outperform *Decoupled RL*, while Full coupling achieves the highest scores on all four tasks. Compared with Full coupling, *w/o IR for Gen.* lowers both generation scores, while *w/o Gen. Reward for Und.* lowers both understanding scores. These results support the complementary roles of forward conditioning and generation-reward feedback.

Inference-time reasoning interventions. We further examine whether generation benefits from the content of the intermediate reasoning output. Using the same trained Sync-R1 checkpoint without retraining, we vary the conditioning information appended to the generation prompt. *No IR* omits the reasoning output; *Random IR* replaces it with an unrelated reasoning output randomly selected

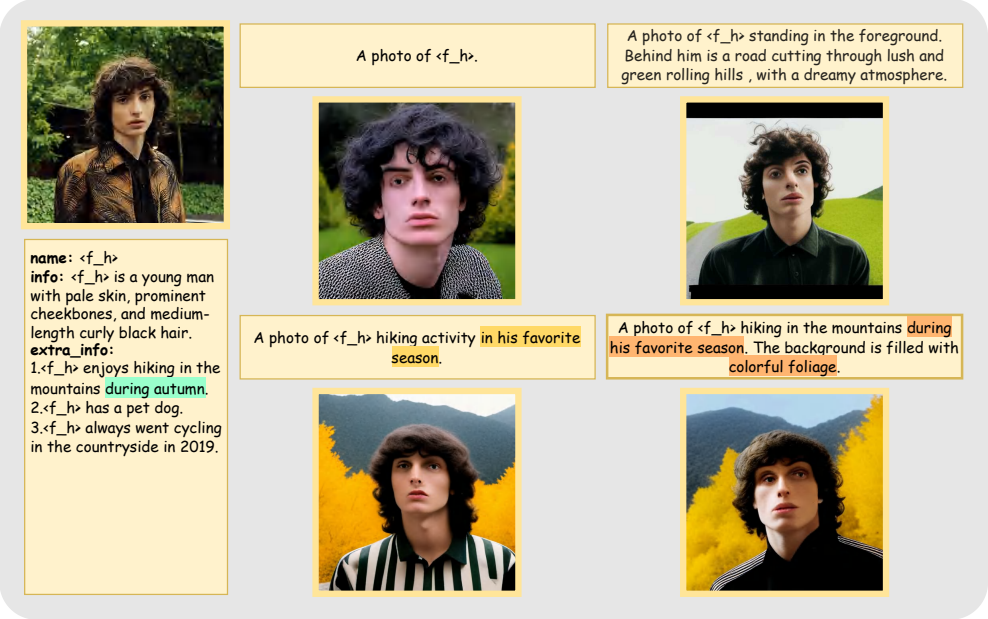


Figure 10: **Qualitative visualization of concept ⟨f_h⟩.**



Figure 11: **Qualitative visualization of concept ⟨butin⟩.**

from another sample; *Direct Context* supplies the entire `extra_info` list directly in place of the intermediate reasoning output; and *Generated IR* uses the model's original reasoning output.

As shown in Table 18, Generated IR outperforms both No IR and Random IR on all four metrics. Replacing the reasoning output with unrelated content fails to recover these gains, supporting the importance of task-relevant reasoning content in guiding personalized generation. Direct Context achieves a Dense Rea. Gen. GPT score close to Generated IR, but yields substantially lower Rea. Gen. CLIP-T. This suggests that selecting task-relevant attributes provides more effective conditioning than directly supplying the full list.

Table 17: Directional ablations of the coupling mechanisms. The two coupling columns indicate reasoning-to-generation conditioning and generation-reward feedback to the reasoning-token loss. Rea. and Dense Rea. report BLEU and GPT scores, respectively. Each generation score is the arithmetic mean of its alignment and CLIP-I metrics.

Method	IR → Gen.	Gen. Reward → IR	Rea.	Dense Rea.	Rea. Gen.	Dense Rea. Gen.
Decoupled RL	No	No	0.234	0.467	0.522	0.518
w/o IR for Gen.	No	Yes	0.240	0.488	0.536	0.527
w/o Gen. Reward for Und.	Yes	No	0.236	0.475	0.544	0.531
Full coupling	Yes	Yes	**0.243**	**0.498**	**0.552**	**0.539**

Table 18: Effect of conditioning information on personalized generation. All conditions use the same trained Sync-R1 checkpoint.

	Rea. Gen.		Dense Rea. Gen.	
Reasoning condition	CLIP-T	CLIP-I	GPT	CLIP-I
No IR	0.271	0.792	0.304	0.751
Random IR	0.273	0.788	0.298	0.748
Direct Context	0.201	0.790	0.346	0.744
Generated IR	**0.324**	**0.801**	**0.353**	**0.756**

H.7 Comparison of Post-Training Strategies

To distinguish the benefits of Sync-GRPO from those of generic RL post-training, we compare UniCTokens-SFT, Standard GRPO, and Sync-GRPO. Both RL methods start from the same UniCTokens-SFT checkpoint and use the same training data, reward ensemble, group size, number of training steps, and generation settings. We additionally include *IR-SFT + Gen. RL*. Since the compound generation prompts lack paired target images, this hybrid baseline trains reasoning with SFT and generation with image-side RL rewards. Generation conditions on model-generated IR during rollout, and image rewards do not update reasoning tokens. It uses the same initialization and training configuration as Sync-GRPO. DGS is disabled for all methods.

Table 19 shows that Standard GRPO improves most task scores over UniCTokens-SFT, demonstrating the benefit of RL post-training. Sync-GRPO further improves all nine reported task scores over Standard GRPO, with particularly clear gains in Dense Rea. and VQA. Sync-GRPO also outperforms IR-SFT + Gen. RL on all nine task scores, supporting the coupled RL formulation over supervised reasoning combined with generation-only RL. This comparison evaluates the combined effect of the Sync-GRPO optimization design.

H.8 Robustness to Incomplete and Noisy Context

We evaluate the full Sync-R1 model by perturbing only its `extra_info` list. *Original* uses the complete list; *Missing* randomly removes 50% of its entries; and *Noisy* appends irrelevant attributes equal to 50% of the original list length.

Table 20 shows modest performance declines under irrelevant distractors. Removing attributes causes larger drops in understanding, consistent with the potential absence of information required by the request, while generation averages show smaller declines.

I Extended Qualitative Results

In this section, we provide additional visual evidence to demonstrate the personalized generation capabilities of **Sync-R1**. We specifically showcase results for two distinct concepts, ⟨f_h⟩ and ⟨butin⟩. As highlighted in Figures 10 and 11, segments requiring high-level inferential reasoning are explicitly marked.

Table 19: Comparison of post-training strategies. Scores are averaged across the evaluation metrics within each subtask.

Method	Understanding					Generation			
	Rec.	Rea.	Dense Rea.	VQA	QA	Pure Gen.	Dense Gen.	Rea. Gen.	Dense Rea. Gen.
UniCTokens-SFT	0.792	0.238	0.385	0.513	0.574	0.515	0.469	0.522	0.515
Standard GRPO	0.826	0.237	0.456	0.534	0.612	0.534	0.474	0.527	0.529
IR-SFT + Gen. RL	0.834	0.235	0.460	0.561	0.618	0.531	0.478	0.534	0.518
Sync-GRPO (w/o DGS)	**0.848**	**0.243**	**0.498**	**0.587**	**0.625**	**0.536**	**0.482**	**0.552**	**0.539**

Table 20: Robustness to incomplete and noisy context. Generation scores average all original metrics.

Context	Rea.	Dense Rea.	Rea. Gen.	Dense Rea. Gen.
Original	0.250	0.503	0.563	0.555
Missing	0.132	0.276	0.543	0.536
Noisy	0.235	0.469	0.550	0.542

The visualizations reveal two core strengths of our approach: (1) Reasoning-to-Image Fidelity: The model successfully decodes implicit conceptual requirements into concrete visual attributes, effectively bridging the gap between abstract reasoning and pixel synthesis. (2) Dense Prompt Adherence: Our method exhibits robust alignment even when presented with dense, multi-attribute textual descriptions, ensuring that every segmented instruction is accurately manifested in the final output. These results further validate that the Sync-GRPO stage significantly enhances the model's ability to handle complex, knowledge-driven personalization tasks across diverse visual domains.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Section 1 state the main contributions, including Sync-R1, Sync-GRPO, DGS, and UnifyBench++. These claims are supported by the method in Section 3, the experiments in Sections 4–4.2, and appendix analyses in Appendices D–F.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Limitations section discusses the scope of concept-level personalization and unresolved conflicting context. Section 4.3 discusses dependence on SFT initialization, and Appendix H.8 evaluates incomplete and noisy context.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The main theoretical assumptions for DGS are stated in Section 3.4. Detailed derivations, variance-reduction proof, SNR analysis, and adaptive-threshold stability analysis are provided in Appendix E.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper discloses the model backbone, training protocol, reward design, benchmark construction, baselines, evaluation metrics, and hyperparameters in Sections 3–4 and Appendices B–G.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code and data are not yet publicly available. However, the paper provides implementation details, dataset construction procedures, evaluation protocols, and hyperparameters in Appendices B and G to support reproducibility. We commit to publicly releasing the constructed dataset upon publication.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training settings, datasets, baselines, metrics, and hyperparameters are described in Section 4.1 and Appendix G. Additional reward and DGS details are provided in Appendices D and F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The paper reports extensive ablations, sensitivity studies, robustness analyses, and evaluator-consistency checks, but does not include formal error bars, confidence intervals, or statistical significance tests for the main experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the GPU configuration in Section 4.1 and provides wall-clock acceleration and runtime breakdown analyses for DGS in Section 4.3 and Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is presented as a research study on personalized multimodal understanding and generation, and we are not aware of any deviation from the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: The paper discusses the technical benefits of personalized multimodal reasoning and generation, but the current draft does not include a dedicated discussion of potential negative societal impacts such as privacy risks or misuse in identity-sensitive image generation.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The current submission does not release a high-risk pretrained model, image generator, or scraped dataset artifact. If such assets are released later, appropriate responsible-release safeguards should be documented.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The paper cites the existing models, datasets, and evaluation tools used in the experiments, but the current draft does not explicitly list licenses and terms of use for all existing assets.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The paper introduces UnifyBench++ and documents its construction, task structure, and evaluation protocol in Appendix B, with additional evaluation details in Appendix G.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not conduct a crowdsourced user study or participant-facing human-subject experiment. Human judgments are used only as auxiliary evaluator-reliability checks in Appendix H.1.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or human-subject experiments requiring IRB or equivalent review. The auxiliary human judgments are used only for evaluation-reliability comparison in Appendix H.1.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The paper uses GPT-4o as part of the evaluation protocol for dense reasoning and alignment assessment, as described in Appendix G. Additional reliability checks for LLM-based evaluation are reported in Appendix H.1.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.