TOOLARTIST: TOOL-USING UNIFIED MULTIMODAL MODELS FOR AGENTIC IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image (T2I) models can produce visually compelling images, yet they remain limited on open-world tasks that require complex semantic understanding, multi-step reasoning, and the integration of external world knowledge. Existing efforts introduce agent capabilities into image generation, but they either prescribe a fixed workflow or place only a subset of the open-world image generation process under agent control. Consequently, reasoning, tool invocation, and image generation are not coordinated by a single policy. We propose ToolArtist, a fully **agentic image generation** model obtained by post-training a Unified Multimodal Model (UMM). ToolArtist dynamically orchestrates reasoning, external tool use, and native image generation within one unified policy. During Supervised Fine-Tuning (SFT), we equip a teacher agent with search tools alongside an image-generation tool. We then convert the collected trajectories into a UMM compatible format, where the image-generation tool is concealed while the resulting generated images are retained. During Reinforcement Learning (RL), we develop an agentic RL infrastructure for UMMs and introduce Reason-Act-Draw GRPO (RAD-GRPO), which uses complementary intent and quality rewards to jointly optimize the model. Experiments show that placing the entire open-world image-generation process under an agent policy consistently outperforms approaches with fixed pipelines or only partially agent-controlled components. We release the training data and the complete post-training infrastructure.

1 INTRODUCTION



Figure 1: Open-world images generated by **ToolArtist**, spanning cultural knowledge, specific IPs, history, and geography.

Text-to-image (T2I) models (Greenberg, 2025; Podell et al., 2023; Chen et al., 2023; Li et al., 2024; Team et al., 2025) can generate images with remarkable structure, detail and aesthetic quality. Nevertheless, these advances fail to enable reliable open-world image generation. For requests requiring complex semantic understanding, multi-hop reasoning, and knowledge of long-tail concepts,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

strict factual constraints or highly time-sensitive information, models often output visually plausible yet factually erroneous images. Recent benchmarks (Han et al., 2025; Zhang et al., 2025b; Huang et al., 2025a;b; Niu et al., 2026; Fu et al., 2024; Zhang et al., 2025a), including WISE (Niu et al., 2026) and WorldGenBench (Zhang et al., 2025a), reveal this capability gap. Critical information is frequently omitted from user prompts and cannot be reliably stored within the model's static parameters.

To overcome this challenge, contemporary works endow image-generation systems with agentic capabilities to gather supplementary information via real-world interaction, either intrinsically or with help from auxiliary agents (Wang et al., 2026; Feng et al., 2026; Chen et al., 2026a; Wu et al., 2023; Yang et al., 2024; Shalev-Arkushin et al., 2026; Zhang et al., 2026; Ye et al., 2026; Jiang et al., 2026; Chen et al., 2026b; 2022). They broadly follow two paradigms. The first organizes prompt understanding, search, evidence aggregation, and image synthesis into a **predefined fixed** pipeline (Chen et al., 2026a). The second depends on a search agent (Zhao et al., 2026; Song et al., 2025) to gather evidence and optimize the user instruction into a grounded generation prompt, which is then passed to an **external** image generator (Feng et al., 2026; Chen et al., 2026b; Jiang et al., 2026). These approaches demonstrate that introducing agentic capabilities can substantially improve open-world image generation quality.

However, they incorporate only part of the agentic capability required by open-world image generation. In the prompt optimization paradigm, the learned agent terminates at generation and delegates synthesis to a separate generator. In the pipeline paradigm, the order and role of search and generation are prescribed in advance. Under both paradigms, image generation is not an action freely chosen and initiated by the model. Consequently, the model does not learn the complete decision process from identifying a knowledge gap, through selecting and using external tools, to producing the final visual output.

Fully agentic image generation requires tool use and image generation to be autonomous actions of the same policy.

We propose **ToolArtist**, a fully agentic image-generation model obtained by post-training a Unified Multimodal Model (UMM) (Chen et al., 2025b; Wang et al., 2024; Cui et al., 2025; Team, 2026; Chen et al., 2025a; Deng et al., 2025). Given a user request, ToolArtist reasons about missing information, decides whether and how to invoke external tools, incorporates the returned textual and visual evidence into its evolving multimodal context, and natively generates and revises the requested image. Image generation is therefore a part of the agent policy.

We develop a post-training strategy to acquire this capability. During SFT, we equip a teacher agent with text-search and image-search tools together with an image-generation tool, allowing it to autonomously construct complete trajectories. We then convert the collected trajectories into a UMM-compatible format: the explicit image-generation tool is concealed, while each generated image is retained and represented as native visual-caption and visual-token spans. This conversion transfers the teacher's tool-using behavior into a form in which generation is performed by the UMM itself. Using 7,132 high-quality trajectories, SFT establishes the model's basic ability to reason, invoke search tools, use retrieved evidence, and generate images within one policy.

During RL, we construct a complete UMM agentic RL infrastructure and introduce RAD-GRPO to optimize full agentic image-generation trajectories online. RAD-GRPO incorporates two complementary signals. ❶ Intent Reward evaluates whether the final generation description accurately and sufficiently translates the user request and acquired evidence into an executable visual intent. ❷ Quality Reward evaluates whether the generated image is faithful to that intent and the original request while maintaining high visual quality. Their combination propagates outcome-level feedback across reasoning, tool use, evidence utilization, and native image generation, further unlocking the agentic capabilities initialized by SFT.

We evaluate ToolArtist on WISE and WorldGenBench-Humanities. The results show that our fully agentic formulation outperforms open-source UMMs and image-generation methods that incorporate only partial agentic capabilities. We release the training data and the complete SFT and RL infrastructure to facilitate future research on agentic image generation.

Our contributions are as follows:

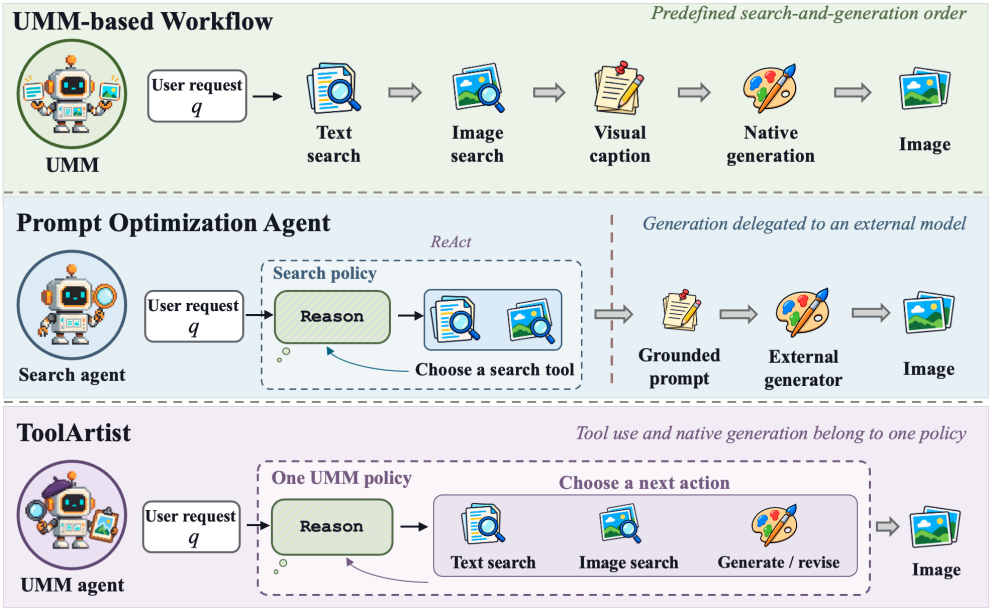


Figure 2: **Agentic image generation paradigms. Top:** A UMM searches and generates in a predefined workflow. **Middle:** A search agent optimizes the prompt for an external generator. **Bottom:** ToolArtist learns one UMM policy that decides when to search and natively generate or revise images.

1. Fully Agentic Image Generation Paradigm. ToolArtist, a fully agentic image-generation model that unifies reasoning, external tool utilization, and native visual generation under a single UMM policy.

2. Post-training Strategy for UMMs. We design a post-training strategy to endow UMMs with agentic image generation capabilities, achieving better performance than approaches with only partial agentic functionalities.

3. Open-source Data and Complete Infrastructure. We construct a dataset of 7k high-quality SFT trajectories and build a full training infrastructure supporting agentic SFT and RL for UMMs.

2 Preliminary

2.1 Open-World Image Generation

Conventional T2I generation is typically formulated as a closed-world conditional generation problem. Given a user instruction q, a generator G directly samples an image $\mathcal{I}$ from a prompt-conditioned distribution. This formulation assumes that all information required for generation is either explicitly provided in q or implicitly stored in the parameters of G. However, this assumption breaks down when the target image depends on external information that must be acquired from the world. We term this setting **open-world image generation**. Let $\mathcal{W}$ denote the world-knowledge space and $\mathcal{Z} \subseteq \mathcal{W}$ the task-relevant textual and visual evidence. Rather than being available in advance, $\mathcal{Z}$ must be actively discovered from the environment:

$$\mathcal{I} \sim G(\cdot \mid q, \mathcal{Z}) \qquad \mathcal{Z} \subseteq \mathcal{W}.$$

2.2 Unified Multimodal Model: Emu3.5

UMMs represent both language and images as tokens and model them within a single multimodal sequence. We build ToolArtist on Emu3.5 (Cui et al., 2025), a native autoregressive UMM trained by unified next-token prediction over interleaved vision–language data. Let $\mathcal{V}_{\text{text}}$ and $\mathcal{V}_{\text{visual}}$ denote its textual and visual token vocabularies. An interleaved sequence $\mathbf{x} = (x_1, \ldots, x_L)$ contains tokens

from $\mathcal{V}_{\text{text}} \cup \mathcal{V}_{\text{visual}}$ and is modeled as

$$p_\theta(\mathbf{x}) = \prod_{j=1}^{L} p_\theta(x_j \mid x_{<j}).$$

Under this formulation, user instructions, reasoning traces, and tool calls are represented as textual tokens, while input images, images returned by tools, and model-generated images are represented as visual-token spans. Emu3.5 can therefore process multimodal observations and natively produce images within the same autoregressive context, making it suitable as a unified policy over reasoning, acting, and drawing.

2.3 Agentic Image Generation

Following the agent interaction pattern introduced by Yao et al. (2023), we formulate agentic image generation as an iterative interaction between a UMM and the open-world environment. We instantiate ToolArtist as

$$A_\theta = (\pi_\theta, \mathcal{T}, \mathcal{W}),$$

where π_θ is the UMM, $\mathcal{T} = \{\texttt{TextSearch}, \texttt{ImageSearch}\}$ is the set of tools, and $\mathcal{W}$ is the open-world environment. Upon receiving q, the policy performs a variable number of interaction rounds. At round t, it conditions on the complete multimodal history $\mathcal{H}_{t-1}$, produces a reasoning span r_t, and then selects an action a_t: either calling a tool from $\mathcal{A}_{\text{tools}}$ or generating an image through $\mathcal{A}_{\text{draw}}$:

$$(r_t, a_t) \sim \pi_\theta(\cdot \mid \mathcal{H}_{t-1}) \qquad a_t \in \mathcal{A}_{\text{tools}} \cup \mathcal{A}_{\text{draw}}.$$

The reasoning span identifies missing knowledge, assesses the currently available evidence, and determines whether the next step should acquire external information or generate an image.

Tool calling When $a_t \in \mathcal{A}_{\text{tools}}$, the policy action is a command $a_t = (n_t, u_t)$, where $n_t \in \mathcal{T}$ specifies the tool and u_t is the query. `TextSearch` retrieves factual knowledge, whereas `ImageSearch` retrieves visual references. The environment executes the command and returns a multimodal observation

$$o_t \sim \mathcal{W}(\cdot \mid a_t).$$

The observation, including the returned text, images, and source-aware summaries, is appended to the history. Conditioned on this new evidence, the policy may continue searching, reformulate the query, switch tools, or proceed to drawing.

Native Image Generation When $a_t \in \mathcal{A}_{\text{draw}}$, the policy action consists of a visual-caption span g_t followed by a visual-token span v_t:

$$a_t = (g_t, v_t).$$

The caption g_t consolidates the user request and acquired evidence into an executable generation intent, while v_t represents the generated image. Unlike a prompt-optimization agent, ToolArtist does not delegate this step to an external image generator: both g_t and v_t are produced by the UMM policy itself. A generated image remains in the multimodal history, so drawing does not necessarily terminate the interaction. The policy may inspect its current result, identify missing or incorrect content, invoke additional search tools, and generate a revised image.

Agentic Trajectory The outcome of round t differs according to the selected action:

$$h_t = \begin{cases} [r_t, a_t, o_t] & a_t \in \mathcal{A}_{\text{tools}}, \\ [r_t, a_t] & a_t \in \mathcal{A}_{\text{draw}}. \end{cases}$$

Starting from $\mathcal{H}_0 = q$, the multimodal history is updated by $\mathcal{H}_t = [\mathcal{H}_{t-1}, h_t]$. A complete trajectory with T rounds is therefore

$$\mathcal{H}_T = [q, h_1, h_2, \ldots, h_T]. \tag{1}$$

The occurrence, ordering, and number of search and drawing actions are not specified by a fixed workflow. The policy determines the interaction path from the evolving context and terminates only when it decides that the latest drawing action constitutes the final answer.

Agentic Image Generation Masking The serialized trajectory contains tokens generated autoregressively by the UMM together with the user input and environment observations inserted as context.

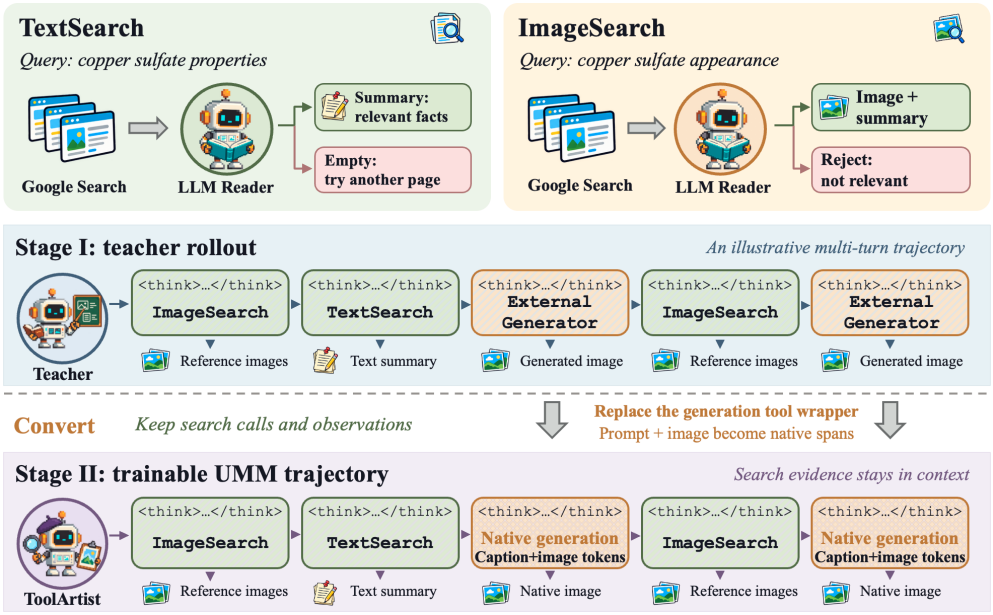


Figure 3: Data synthesis for post-training. **Top:** Search tools return textual and visual evidence. **Bottom:** Teacher rollouts are converted into native multimodal trajectories for UMM supervision.

Let $\mathbf{T}$ denote the collection of token instances belonging to the reasoning spans r_t and action spans a_t. Let $\mathbf{O}$ contain the conditioning instruction q and all environment-returned observations o_t:

$$\mathbf{T} = \bigcup_t (r_t \cup a_t), \qquad \mathbf{O} = q \cup \bigcup_t o_t.$$

Here, the collections refer to token instances at their serialized positions rather than vocabulary identities. For the j-th serialized token y_j, the policy-support mask is

$$M_j = \mathbb{I}[y_j \in \mathbf{T}].$$

Tokens in $\mathbf{O}$ remain available for conditional modeling but are excluded from the policy objective. Consequently, training directly optimizes the complete set of reasoning, searching, and drawing behaviors produced by the UMM policy.

3 Training

We post-train the UMM in two stages. The first stage is SFT, which itself consists of two data-processing steps: a teacher agent performs multi-turn rollouts with text search, image search, and an image-generation tool to produce raw interaction trajectories; a converter then removes the dependence on the image-generation tool and turns these trajectories into training samples for the UMM. In the second stage, RL, we use **Reason-Act-Draw GRPO (RAD-GRPO)** to directly optimize the complete policy through interaction with the open-world environment, using both intent-level and quality-level rewards.

3.1 Data Synthesis and Conversion

As shown in Figure 3, when constructing the SFT data, the teacher agent follows user instructions through multi-turn interaction and calls `TextSearch`, `ImageSearch`, and an external image generator (we use `gemini-3-pro-image-preview`). The first two tools gather factual evidence and reference images, while the external image generator actually produces images during synthesis.

Stage I: rollout. The raw sample produced by the teacher agent is a complete multi-turn trajectory, including reasoning, tool calls, tool responses, and the final externally generated image. Unlike a static "search first, draw later" pipeline, the rollout here is alternating: the model can reason first, then call a tool, then keep reasoning or search again until it reaches the final generation intent.
The search tools are also enhanced for open-world image generation: `TextSearch` uses Google

Search together with an LLM Reader to retrieve, summarize, and fall back across candidate pages; `ImageSearch` first filters out pages that cannot be downloaded, uses the LLM Reader to judge whether the content is relevant, and finally returns directly usable reference images together with structured summaries. This makes the evidence more stable and easier to use in later conversion and training.

Stage II: convert. In the convert stage, the raw trajectories are organized into unified token-level supervision samples. For the search part, the text and images returned by the tools are kept in the context. For the external image-generation step, the converter no longer treats it as a separate tool output; instead, it rewrites it into a native multimodal generation format: the prompt used to call the image-generation tool is written as a visual-caption span, followed by the corresponding image tokens. In other words, every external image generation in the teacher trajectory is converted into the form in which the model itself completes image generation in the same autoregressive trajectory. If a trajectory fails before the final image, or if image loading, tokenization, or context-length checks fail, the whole sample is filtered out.

3.2 Supervised Fine-Tuning

Let $\mathcal{D}_{\mathrm{SFT}} = \{x_i, y_i\}_{i=1}^{N}$ denote the dataset obtained after rollout and convert, where $x_i = q_i$ is the conditioning input and y_i is the corresponding multimodal policy trajectory. Each example is serialized as $\mathcal{H}_i = [x_i; y_i] = (u_{i,1}, \ldots, u_{i,L_i})$, where L_i denotes the valid token-sequence length. Let $\mathcal{T}_i$ denote the set of policy-generated tokens in example i, as defined by the agentic image generation masking formulation in the preliminary section. Define the indicator $M_{i,j} \triangleq \mathbb{I}[u_{i,j} \in \mathcal{T}_i]$.

The SFT objective is:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x_i,y_i)\sim\mathcal{D}_{\mathrm{SFT}}}\left[\frac{\sum_{j=1}^{L_i} M_{i,j}\log P_\theta\left(u_{i,j} \mid u_{i,<j}\right)}{\sum_{j=1}^{L_i} M_{i,j}}\right].$$

This objective supervises only the textual and visual spans generated by the policy, while excluding the conditioning prefix and environment-returned observations.

3.3 Reason–Act–Draw GRPO

SFT provides a stable initialization, but it remains limited by the coverage of the synthetic data and therefore struggles to discover better agentic image generation strategies. We further apply RAD-GRPO to directly optimize the complete policy on real inference trajectories.

For each input q, the rollout policy interacts with the open-world environment and samples a group of B trajectories:

$$\mathcal{H}_i \sim \pi_{\theta_{\mathrm{old}}}(\cdot \mid q) \otimes \mathcal{W} \qquad i = 1, \ldots, B.$$

Each trajectory may freely interleave reasoning, search, and visual generation. Let T_i denote the termination step of trajectory $\mathcal{H}_i$. Its final visual-generation action is represented as (g_{T_i}, v_{T_i}), where g_{T_i} and v_{T_i} denote the final visual-caption span and the final visual-token span, respectively.

Dual Reward. RAD-GRPO does not only look at the final image. It looks at both the generation intent and the final image quality.

- **Intent reward R_i^{I}.** This reward checks whether the final visual-caption span g_{T_i} turns the user request and the acquired evidence into a sufficient, accurate, and executable generation description. In other words, it measures whether this prompt would be enough for an ideal generator to produce the right image. If the trajectory has no valid final caption, we set this term to 0.

- **Quality reward R_i^{Q}.** This reward checks whether the image decoded from v_{T_i} really satisfies the user request and matches the final caption. In code, it is judged by a world-knowledge reward model with four dimensions: faithfulness, visual correctness, text accuracy, and aesthetics, combined with weights 0.1, 0.4, 0.4, and 0.1. If the task does not require readable text, text accuracy falls back to 0.5.

The two terms are combined into a main reward:

$$R_i = \alpha R_i^{\mathrm{I}} + (1-\alpha) R_i^{\mathrm{Q}}, \qquad \alpha \in [0,1].$$

The default is $\alpha = 0.5$. This means the caption term constrains whether the generation intent is good, while the image term constrains whether the final image is actually correct.

Auxiliary rewards. We add a **format reward** for complete outputs, valid tool calls, and the required think/caption/image structure; a **draw signal** encouraging native image generation; a **length penalty** discouraging overly long trajectories; and a **no-draw penalty** for trajectories that never generate an image, preventing collapse into a search-only policy.

Let $\widetilde{R}_i$ denote the final reward of trajectory $\mathcal{H}_i$, combining the dual and auxiliary rewards. For each instruction q, we sample a group of B trajectories from the old policy and optimize the following token-normalized GRPO objective:

$$\begin{aligned}\mathcal{J}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{\mathcal{H}_i\}_{i=1}^{B}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \\ \left[\frac{1}{\sum_{i=1}^{B}\sum_{j=1}^{L_i} M_{i,j}} \sum_{i=1}^{B}\sum_{j=1}^{L_i} M_{i,j} \right. \\ \left. \times \left(\min\left(\rho_{i,j}(\theta)\widehat{A}_i, \text{clip}(\rho_{i,j}(\theta), 1-\epsilon, 1+\epsilon)\widehat{A}_i\right) - \beta \widehat{D}_{i,j}^{\text{KL}} \right) \right].\end{aligned} \tag{2}$$

Here, $M_{i,j}$ is the policy-support mask defined in the preliminary section: it includes tokens generated in reasoning and action spans, while excluding the user instruction and environment-returned observations. The importance ratio and group-relative advantage are

$$\rho_{i,j}(\theta) = \frac{\pi_\theta(u_{i,j} \mid u_{i,<j})}{\pi_{\theta_{\text{old}}}(u_{i,j} \mid u_{i,<j})}, \qquad \widehat{A}_i = \frac{\widetilde{R}_i - \text{mean}(\{\widetilde{R}_b\}_{b=1}^{B})}{\text{std}(\{\widetilde{R}_b\}_{b=1}^{B}) + \varepsilon}. \tag{3}$$

The same trajectory-level advantage $\widehat{A}_i$ is applied to all policy-generated tokens in $\mathcal{H}_i$. Therefore, the final intent and image-quality feedback jointly optimize the complete sequence of reasoning, tool use, and native image generation. The KL term $\widehat{D}_{i,j}^{\text{KL}}$ regularizes the updated policy toward the frozen SFT reference policy, and β controls its strength.

4 Experiments

4.1 Evaluation Setting

Bench. WISE (Niu et al., 2026) measures world-knowledge-grounded generation across culture, time, space, biology, physics, and chemistry using WiScore, which combines image–text consistency, realism, and aesthetics. WorldGenBench-Humanities (WGB) (Zhang et al., 2025a) tests cultural, historical, and geographic grounding across seven continents using the Knowledge Checklist Score (KCS), the fraction of prompt-specific knowledge requirements satisfied by an image. The WGB columns AF, AN, AS, EU, NA, OC, and SA denote Africa, Antarctica, Asia, Europe, North America, Oceania, and South America, respectively; Avg. reports the overall benchmark score.

Baselines. We compare frontier proprietary generators; general image-generation models such as FLUX, SDXL, and LongCat-Image (Greenberg, 2025; Podell et al., 2023; Team et al., 2025); unified multimodal models, including VILA-U, Janus-Pro, Emu3.5, Hunyuan-Image 3.0, BLIP3-o, and BAGEL (Wu et al., 2025; Chen et al., 2025b; Cui et al., 2025; Team, 2026; Chen et al., 2025a; Deng et al., 2025); and agentic image-generation methods such as Unify-Agent and GenSearcher (Chen et al., 2026a; Feng et al., 2026).

4.2 Main Results

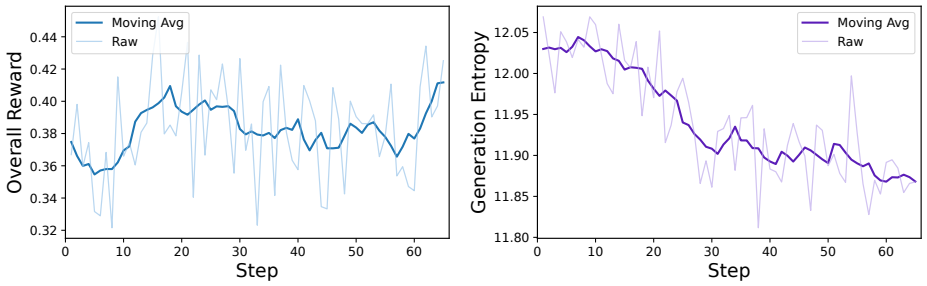
As shown in Table 1, ToolArtist achieves the highest WISE overall score among the reported non-proprietary models (0.79), with strong results in Physics (0.81) and Chemistry (0.79). Proprietary models remain ahead overall, particularly in Time and Space. On WorldGenBench-Humanities, ToolArtist obtains an average KCS of 22.10, compared with 16.80 for LongCat-Image, 15.58 for Unify-Agent, and 13.66 for GenSearcher. Among the reported non-proprietary models, ToolArtist leads in Africa, Antarctica, Asia, Europe, and Oceania; LongCat-Image leads in North America, and Unify-Agent in South America.

Table 1: **Main results on WISE and WorldGenBench-Humanities.** Bold and underlined scores mark the best and second-best reported non-proprietary results in each column; ties share the same marker.

Method	WISE							WorldGenBench-Humanities							
	Cul.	Time	Space	Bio.	Phys.	Chem	Avg.	AF	AN	AS	EU	NA	OC	SA	Avg.
Frontier Proprietary Models															
Nano Banana-Pro	0.89	0.80	0.89	0.88	0.86	0.85	0.87	27.75	23.80	28.82	29.77	28.27	30.65	25.83	28.62
Nano Banana	0.89	0.87	0.95	0.89	0.89	0.79	0.89	29.91	17.69	31.10	28.93	29.41	31.37	26.53	29.67
General Image-Generation Models															
FLUX.1-schnell	0.39	0.44	0.50	0.31	0.44	0.26	0.40	11.31	8.61	13.52	11.79	10.84	12.38	12.96	12.00
SD-XL-base-0.9	0.43	0.48	0.47	0.44	0.45	0.27	0.43	10.89	9.09	11.47	10.14	9.94	9.54	10.74	10.55
SD-3.5-large	0.44	0.50	0.58	0.44	0.52	0.31	0.46	11.82	13.24	13.43	12.72	11.46	11.91	15.57	12.57
FLUX.1-dev	0.48	0.58	0.62	0.42	0.51	0.35	0.50	8.43	11.30	10.15	10.59	8.23	8.43	9.63	9.36
LongCat-Image	0.66	0.61	0.72	0.66	0.72	0.49	0.65	15.63	16.30	15.52	16.59	**21.52**	12.56	21.67	16.80
Unified Multimodal Models															
VILA-u-7B-256	0.26	0.33	0.37	0.35	0.39	0.23	0.31	6.19	3.74	6.73	5.42	5.23	3.67	4.58	5.62
Janus-Pro-7B	0.30	0.37	0.49	0.36	0.42	0.26	0.35	6.87	5.45	8.57	9.22	5.28	5.24	8.46	7.41
Emu3.5	0.73	0.39	0.33	0.25	0.36	0.33	0.49	13.19	9.44	12.77	14.75	6.98	18.45	17.78	13.17
Hunyuan-Image-3.0	0.58	0.57	0.70	0.56	0.63	0.31	0.57	11.58	10.93	12.57	14.70	12.71	9.44	17.64	12.76
BLIP3o-8B	0.49	0.51	0.63	0.54	0.63	0.37	0.52	13.58	14.81	14.71	12.70	11.53	11.77	21.67	13.63
BAGEL+CoT	0.76	0.69	0.75	0.65	0.75	0.58	0.70	8.53	13.61	14.62	12.97	12.94	12.14	10.14	12.09
Agentic Image Generation Models															
Unify-Agent	0.82	**0.75**	0.74	0.72	0.73	0.70	0.77	13.88	13.80	15.43	15.65	15.61	14.78	**24.31**	15.58
GenSearcher	0.80	0.71	**0.82**	**0.76**	0.74	0.75	0.77	12.40	9.35	18.46	14.71	10.33	10.16	12.64	13.66
ToolArtist	**0.87**	0.62	0.75	0.75	**0.81**	**0.79**	**0.79**	**24.44**	**17.13**	**26.19**	**18.69**	19.52	**20.97**	19.44	**22.10**

5 Analysis

5.1 Effectiveness of RAD-GRPO



(a) Overall reward. **(b)** Entropy of actor model.

Figure 4: RAD-GRPO training dynamics.

Reward Dynamics. Figure 4(a) shows noisy rewards from online search and joint caption–image evaluation, but an improving moving average. After an initial dip, reward rises rapidly at Steps 10–20, fluctuates around 0.37–0.40 mid-training, and reaches approximately 0.41 near the end, indicating improved search-to-image trajectories.

Policy Entropy Dynamics. In panel (b), actor entropy gradually decreases from approximately 12.03 to 11.87–11.91 and stabilizes in the latter half of training. Together with rising reward, this suggests concentration on higher-reward reasoning, search, and generation behaviors, without a sudden decline or evident entropy collapse.

Effect of RL. RAD-GRPO raises WISE Overall from 0.74 to 0.79 (Table 2), improving all six categories, especially Chemistry (+0.07) and Physics (+0.06).

Table 2: **Effect of RAD-GRPO on WISE.** Green numbers denote absolute gains over the SFT checkpoint.

Setting	Cul.	Time	Space	Bio.	Phys.	Chem.	Avg.
SFT	0.83	0.58	0.71	0.70	0.75	0.72	0.74
SFT+RL	0.87 (+0.04)	0.62 (+0.04)	0.75 (+0.04)	0.75 (+0.05)	0.81 (+0.06)	0.79 (+0.07)	0.79 (+0.05)

5.2 Effect of Source-Aware Image Search Summaries

Each image-search reference includes an LLM summary of its source webpage to distinguish visually plausible but semantically incorrect results. Removing only these summaries reduces WISE Overall from 0.79 to 0.61 (Table 3). Biology shows the largest drop, from 0.75 to 0.25, highlighting the importance of source context for fine-grained factual identity.

Table 3: **Ablation of source-aware summaries in `image_search`.** Red numbers denote drops from the full toolchain.

Setting	Cul.	Time	Space	Bio.	Phys.	Chem.	Avg.
Full	0.87	0.62	0.75	0.75	0.81	0.79	0.79
w/o summary	0.78 (-.09)	0.54 (-.08)	0.54 (-.21)	0.25 (-.50)	0.64 (-.17)	0.50 (-.29)	0.61 (-.18)

We provide additional experimental details and results in the appendix, including comparisons with agentic image-generation models (Section A.1), case studies (Section A.2), SFT data distribution and trajectory examples (Section A.3), and the complete prompts for the teacher agent, ToolArtist, and reader (Section A.4).

6 Related Work

Unified Multimodal Model. Specialized diffusion and flow-based generators achieve strong visual quality (Podell et al., 2023; Chen et al., 2023; Li et al., 2024; Greenberg, 2025; Team et al., 2025), while UMMs unify understanding and generation. Janus-Pro uses separate visual pathways with a shared autoregressive backbone (Chen et al., 2025b); and BAGEL learns understanding, generation, and manipulation through interleaved pretraining (Deng et al., 2025). Emu3.5 scales native interleaved vision–language modeling and strengthens reasoning and generation through post-training (Cui et al., 2025). ToolArtist post-trains Emu3.5 to choose when to reason, search, and draw within one policy.

Tool-Using Agents and Agentic Image Generation. Yao et al. (2023) introduce ReAct, which interleaves reasoning and environment actions. Image-generation systems use external evidence (Chen et al., 2022), planning, refinement, or tool orchestration (Wu et al., 2023; Yang et al., 2024; Shalev-Arkushin et al., 2026; Ye et al., 2025; Jiang et al., 2026; Chen et al., 2026b; Ye et al., 2026; Zhang et al., 2026). Unify-Agent places understanding, multimodal search, recaptioning, and synthesis in a structured UMM pipeline (Chen et al., 2026a). GenSearcher trains multi-hop text and image retrieval with SFT and RL, then delegates synthesis to a separate generator (Feng et al., 2026). ToolArtist instead learns search and native generation jointly as actions of one policy.

Reinforcement Learning for Visual Generation. GRPO estimates relative advantages from groups of outputs without a learned critic (Shao et al., 2024). Liu et al. (2026) introduce UniGRPO, which jointly optimizes a single reasoning-then-generation round, while Nie et al. (2026) optimize multimodal text–image outputs with interleaved GRPO.

7 Conclusion

In conclusion, ToolArtist unifies reasoning, external tool use, and native image generation under a single UMM policy, providing a fully agentic approach to open-world image generation. To support reproducibility and future research, we open source the complete pipeline from synthetic data generation through Emu SFT and RL training at the https://anonymous.4open.science/r/EMU-Agentic-PostTrain-E4FF.

AI USE STATEMENT

We used generative AI to synthesize SFT data and LLM judges to compute rewards during RL and assess model outputs during evaluation. Codex assisted with manuscript writing and polishing, as well as code repository organization.

REFERENCES

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, Le Xue, Caiming Xiong, and Ran Xu. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset, 2025a. URL https://arxiv.org/abs/2505.09568.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. URL https://arxiv.org/abs/2310.00426.

Shuang Chen, Quanxin Shou, Hangting Chen, Yucheng Zhou, Kaituo Feng, Wenbo Hu, Yi-Fan Zhang, Yunlong Lin, Wenxuan Huang, Mingyang Song, Dasen Dai, Bolin Jiang, Manyuan Zhang, Shi-Xue Zhang, Zhengkai Jiang, Lucas Wang, Zhao Zhong, Yu Cheng, and Nanyun Peng. Unify-agent: A unified multimodal agent for world-grounded image synthesis, 2026a. URL https://arxiv.org/abs/2603.29620.

Sixiang Chen, Zhaohu Xing, Tian Ye, Xinyu Geng, Yunlong Lin, Jianyu Lai, Xuanhua He, Fuxiang Zhai, Jialin Gao, and Lei Zhu. Genevolve: Self-evolving image generation agents via tool-orchestrated visual experience distillation, 2026b. URL https://arxiv.org/abs/2605.21605.

Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W. Cohen. Re-imagen: Retrieval-augmented text-to-image generator, 2022. URL https://arxiv.org/abs/2209.14491.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling, 2025b. URL https://arxiv.org/abs/2501.17811.

Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, Yueze Wang, Chengyuan Wang, Fan Zhang, Yingli Zhao, Ting Pan, Xianduo Li, Zecheng Hao, Wenxuan Ma, Zhuo Chen, Yulong Ao, Tiejun Huang, Zhongyuan Wang, and Xinlong Wang. Emu3.5: Native multimodal models are world learners, 2025. URL https://arxiv.org/abs/2510.26583.

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.

Kaituo Feng, Manyuan Zhang, Shuang Chen, Yunlong Lin, Kaixuan Fan, Yilei Jiang, Hongyu Li, Dian Zheng, Chenyang Wang, and Xiangyu Yue. Gen-searcher: Reinforcing agentic search for image generation, 2026. URL https://arxiv.org/abs/2603.28767.

Xingyu Fu, Muyu He, Yujie Lu, William Yang Wang, and Dan Roth. Commonsense-t2i challenge: Can text-to-image generation models understand commonsense?, 2024. URL https://arxiv.org/abs/2406.07546.

Or Greenberg. Demystifying flux architecture, 2025. URL https://arxiv.org/abs/2507.09595.

Tianyang Han, Junhao Su, Junjie Hu, Peizhen Yang, Hengyu Shi, Junfeng Luo, and Jialin Gao. Beyond words and pixels: A benchmark for implicit world knowledge reasoning in generative models, 2025. URL https://arxiv.org/abs/2511.18271.

Hsin-Ping Huang, Xinyi Wang, Yonatan Bitton, Hagai Taitelbaum, Gaurav Singh Tomar, Ming-Wei Chang, Xuhui Jia, Kelvin C. K. Chan, Hexiang Hu, Yu-Chuan Su, and Ming-Hsuan Yang. Kitten: A knowledge-intensive evaluation of image generation on visual entities, 2025a. URL https://arxiv.org/abs/2410.11824.

Ziwei Huang, Wanggui He, Quanyu Long, Yandi Wang, Haoyuan Li, Zhelun Yu, Fangxun Shu, Weilong Dai, Hao Jiang, Fei Wu, and Leilei Gan. T2i-factualbench: Benchmarking the factuality of text-to-image models with knowledge-intensive concepts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27501–27524. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.acl-long.1334. URL http://dx.doi.org/10.18653/v1/2025.acl-long.1334.

Kaixun Jiang, Yuzheng Wang, Junjie Zhou, Pandeng Li, Zhihang Liu, Chen-Wei Xie, Zhaoyu Chen, Yun Zheng, and Wenqiang Zhang. Genagent: Scaling text-to-image generation via agentic multimodal reasoning, 2026. URL https://arxiv.org/abs/2601.18543.

Daiqing Li, Aleks Kamko, Ehsan Akhgari, Ali Sabet, Linmiao Xu, and Suhail Doshi. Playground v2.5: Three insights towards enhancing aesthetic quality in text-to-image generation, 2024. URL https://arxiv.org/abs/2402.17245.

Jie Liu, Zilyu Ye, Linxiao Yuan, Shenhan Zhu, Yu Gao, Jie Wu, Kunchang Li, Xionghui Wang, Xiaonan Nie, Weilin Huang, and Wanli Ouyang. Unigrpo: Unified policy optimization for reasoning-driven visual generation, 2026. URL https://arxiv.org/abs/2603.23500.

Ming Nie, Chunwei Wang, Jianhua Han, Hang Xu, and Li Zhang. Towards unified multimodal interleaved generation via group relative policy optimization, 2026. URL https://arxiv.org/abs/2603.09538.

Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Fanqing Meng, Kunpeng Ning, Bin Zhu, and Li Yuan. Wise: A world knowledge-informed semantic evaluation for text-to-image generation, 2026. URL https://arxiv.org/abs/2503.07265.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis, 2023. URL https://arxiv.org/abs/2307.01952.

Rotem Shalev-Arkushin, Rinon Gal, Amit H. Bermano, and Ohad Fried. Imagerag: Dynamic image retrieval for reference-guided image generation, 2026. URL https://arxiv.org/abs/2502.09411.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Huatong Song, Jinhao Jiang, Wenqing Tian, Zhipeng Chen, Yuhuan Wu, Jiahao Zhao, Yingqian Min, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher++: Incentivizing the dynamic knowledge acquisition of llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2505.17005.

Meituan LongCat Team, Hanghang Ma, Haoxian Tan, Jiale Huang, Junqiang Wu, Jun-Yan He, Lishuai Gao, Songlin Xiao, Xiaoming Wei, Xiaoqi Ma, Xunliang Cai, Yayong Guan, and Jie Hu. Longcat-image technical report, 2025. URL https://arxiv.org/abs/2512.07584.

Tencent Hunyuan Foundation Model Team. Hunyuanimage 3.0 technical report, 2026. URL https://arxiv.org/abs/2509.23951.

Haozhe Wang, Weijia Feng, Jinpeng Yu, Che Liu, Ping Nie, Fangzhen Lin, Jiaming Liu, Ruihua Huang, Jimmy Lin, Wenhu Chen, and Cong Wei. Search beyond what can be taught: Evolving the knowledge boundary in agentic visual generation, 2026. URL https://arxiv.org/abs/2607.05382.

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.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models, 2023. URL https://arxiv.org/abs/2303.04671.

Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, Song Han, and Yao Lu. Vila-u: a unified foundation model integrating visual understanding and generation, 2025. URL https://arxiv.org/abs/2409.04429.

Zhengyuan Yang, Jianfeng Wang, Linjie Li, Kevin Lin, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. Idea2img: Iterative self-refinement with gpt-4v(ision) for automatic image design and generation, 2024. URL https://arxiv.org/abs/2310.08541.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2023. URL https://arxiv.org/abs/2210.03629.

Junyan Ye, Jun He, Zilong Huang, Dongzhi Jiang, Xuan Yang, Rui Chen, and Weijia Li. Genclaw: Code-driven agentic image generation, 2026. URL https://arxiv.org/abs/2605.30248.

Wen Ye, Zhaocheng Liu, Yuwei Gui, Tingyu Yuan, Yunyue Su, Bowen Fang, Chaoyang Zhao, Qiang Liu, and Liang Wang. Genpilot: A multi-agent system for test-time prompt optimization in image generation, 2025. URL https://arxiv.org/abs/2510.07217.

Daoan Zhang, Che Jiang, Ruoshi Xu, Biaoxiang Chen, Zijian Jin, Yutian Lu, Jianguo Zhang, Liang Yong, Jiebo Luo, and Shengda Luo. Worldgenbench: A world-knowledge-integrated benchmark for reasoning-driven text-to-image generation, 2025a. URL https://arxiv.org/abs/2505.01490.

Wenchao Zhang, Jiahe Tian, Runze He, Jizhong Han, Jiao Dai, Miaomiao Feng, Wei Mi, and Xiaodan Zhang. Align beyond prompts: Evaluating world knowledge alignment in text-to-image generation, 2025b. URL https://arxiv.org/abs/2505.18730.

Zekai Zhang, Jiahao Li, Jie Zhang, Kaiyuan Gao, Kun Yan, Lihan Jiang, Ningyuan Tang, Shengming Yin, Tianhe Wu, Xiaoyue Chen, Xiao Xu, Yan Shu, Yanran Zhang, Yixian Xu, Yuxiang Chen, Zhendong Wang, Zihao Liu, Zikai Zhou, Huishuai Zhang, Dongyan Zhao, and Chenfei Wu. Qwen-image-agent: Bridging the context gap in real-world image generation, 2026. URL https://arxiv.org/abs/2606.26907.

Jiahao Zhao, Shaoxuan Xu, Zhongxiang Sun, Fengqi Zhu, Jingyang Ou, Yuling Shi, Chongxuan Li, Xiao Zhang, and Jun Xu. Dllm-searcher: Adapting diffusion large language model for search agents, 2026. URL https://arxiv.org/abs/2602.07035.

A APPENDIX

Appendix overview. The supplementary material is organized into four sections:

Contents		**Page**
A.1	Comparison with Agentic Image Generation Models	13
A.2	Case Studies	18
A.3	SFT Data Distribution and Examples	21
A.4	Prompt Displays	28

A.1 COMPARISON WITH AGENTIC IMAGE GENERATION MODELS

We compare ToolArtist with Unify-Agent and GenSearcher on five selected WorldGenBench-Humanities cases. Each case includes the evaluation prompt, generated images, and a checklist-level assessment of cultural, historical, and geographic grounding.

A.1.1 SOCOTRA: RESIN COLLECTION

Figure 5 presents a WorldGenBench-Humanities case about dragon's-blood resin collection on Socotra Island in 1955. The full evaluation prompt is:

WorldGenBench-Humanities Prompt

```
A dragon's blood tallow collector on Socotra Island is cleaning a
split in the trunk in 1955.  The sea breeze sends a salty scent,
and in the distance the cries of sheep.  These ancient trees have
witnessed countless generations, and the collector's skills have been
passed down from generation to generation.
```

The prompt requires both cultural details and geographic grounding: a traditional resin collector, appropriate tools and clothing, dragon-blood trees, local livestock, storage containers, and the coastline of Socotra. Our RL checkpoint obtains KCS 0.625, while Unify-Agent obtains 0.000 and GenSearcher obtains 0.250. Our result satisfies five checklist points. First, the collector is shown using traditional resin-harvesting implements, including a wooden scraping tool and a clay-like container. Second, the figure wears loose traditional clothing and a head covering that fit the tropical island setting, rather than modern casual clothing. Third, grazing goats are visible in the scene, matching the local livestock requirement. Fourth, storage containers are placed near the collector, making the resin-collection activity concrete rather than only implied. Finally, the image includes a visible coastline and ocean, grounding the scene on Socotra Island. The remaining missed points are the limestone-mountain landscape, endemic island vegetation such as agave or desert thistle, and traditional stone houses.



Figure 5: A case on WorldGenBench-Humanities. Our model better grounds the Socotra resin-collection scene in traditional tools, clothing, storage containers, grazing goats, and coastline cues.

Table 4: **Checklist evaluation for the Socotra case.** Each row is one score point from the benchmark checklist. Blue rows mark the points satisfied by our model. ✓ means satisfied and ✗ means not satisfied.

Checklist item	Ours	Unify-Agent	GenSearcher
Traditional resin-harvesting tools, such as wooden scrapers and clay pots	✓	✗	✗
Lightweight traditional clothing appropriate for the tropical climate	✓	✗	✗
Socotra landscape with limestone mountains	✗	✗	✗
Visible grazing goats, the main livestock in the area	✓	✗	✓
Typical island vegetation such as agave and desert thistle	✗	✗	✗
Traditional stone houses visible in the distance	✗	✗	✗
Storage containers for collected resin, possibly local-material products	✓	✗	✗
Visible coastline and Indian Ocean	✓	✗	✓
KCS	0.625	0.000	0.250

A.1.2 WHITECHAPEL: VICTORIAN STREET SCENE

Figure 6 presents a WorldGenBench-Humanities case about a working woman in Whitechapel, East London, in 1889. The full evaluation prompt is:

WorldGenBench-Humanities Prompt

```
On an overcast November day in 1889, a working woman in the
Whitechapel district of East London was walking home.  A thick fog
shrouded the narrow streets, and the gas streetlights glowed dimly.
She picks up her pace as she walks past a crowded room of low-cost
apartments, and the smell of factory smoke fills the air.  It had been
only a year since the Jack the Ripper case, and the neighborhood was
still shrouded in fear.
```

The prompt requires both Victorian East End social cues and atmospheric grounding: cobblestone streets, patched working-class dress, brick townhouse apartments, cast-iron gas streetlights, coal smog, street occupations, police patrols, soot-darkened facades, street sanitation details, and faded curtains. Our RL checkpoint obtains KCS 0.600, while Unify-Agent obtains 0.300 and GenSearcher obtains 0.300. Our result satisfies six checklist points. First, the road is paved with round cobblestones. Second, the woman wears a patched dark dress, capturing the working-class poverty cue. Third, the street is lined with brick Victorian townhouse apartments. Fourth, the gas streetlights use the expected cast-iron posts and glass shades. Fifth, tawny coal smoke is visible in the air. Sixth, soot-darkened facades make the industrial pollution concrete. The remaining missed points are newspaper boys, police patrols with sirens and batons, horse manure and garbage, and faded curtains.



Figure 6: Case study on WorldGenBench-Humanities. Our model better grounds the Whitechapel scene in patched working-class dress, coal-smog atmosphere, and soot-darkened Victorian facades while preserving the shared street, housing, and gaslight cues.

Table 5: **Checklist evaluation for the Whitechapel case.** Each row is one score point from the benchmark checklist. Blue rows mark the points satisfied by our model. ✓ means satisfied and ✗ means not satisfied.

Checklist item	Ours	Unify-Agent	GenSearcher
The streets are paved with round cobblestones	✓	✓	✓
Female workers wear patched and repaired dark-colored dresses	✓	✗	✗
The street is lined with brick Victorian townhouse apartments	✓	✓	✓
The gas streetlight is cast iron with a glass shade on top	✓	✓	✓
Tawny smoke floats in the air	✓	✗	✗
Newspaper boys selling newspapers on the street	✗	✗	✗
Visible police patrols equipped with sirens and batons	✗	✗	✗
Building facades blackened by soot	✓	✗	✗
Horse manure and garbage in the streets	✗	✗	✗
Faded curtains hang from the windows	✗	✗	✗
KCS	0.600	0.300	0.300

A.1.3 ANTOFAGASTA: SALTPETER PORT

Figure 7 presents a WorldGenBench-Humanities case about a saltpeter miner at the port of Antofagasta during the Chilean Civil War in February 1891. The full evaluation prompt is:

WorldGenBench-Humanities Prompt

```
In February 1891, at the height of the Chilean Civil War, an aging
saltpeter miner stood in the port of Antofagasta and watched the last
shipment of saltpeter being loaded.  As a businessman who supported
President Balmaceda, he was well aware that the Congressional Army was
about to occupy this important port.  In the early morning chill, he
watched harbor workers come and go, and in the distance came the sound
of a train whistle.
```

The prompt requires port-infrastructure details and regional historical grounding: Victorian iron cranes, labor clothing, steam locomotives and tracks, saltpeter packed as white crystals in burlap bags, British-influenced formal attire for the mine owner, colonial masonry harbor buildings, steam freighters, the Atacama desert landscape, morning coastal fog, and British merchant-bank signage. Our RL checkpoint obtains KCS 0.500, while Unify-Agent obtains 0.300 and GenSearcher obtains 0.200. Our result satisfies five checklist points. First, the port workers wear rough shirts, loose trousers, and sun hats. Second, the harbor buildings use a colonial masonry style. Third, a steam freighter is visible in the distance. Fourth, the surrounding terrain is sparse and desert-like, matching Antofagasta's Atacama setting. Finally, the sky appears misty in the early morning. The remaining missed points are Victorian iron cranes, visible steam locomotives and tracks, white crystalline saltpeter in burlap bags, the mine owner's British formal attire and pocket watch, and a British merchant-bank sign on the pier.



Figure 7: Case study on WorldGenBench-Humanities. Our model better grounds the Antofagasta port scene in period labor clothing, colonial masonry harbor buildings, a distant steam freighter, desert surroundings, and morning fog.

Table 6: **Checklist evaluation for the Antofagasta case.** Each row is one score point from the benchmark checklist. Blue rows mark the points satisfied by our model. ✓ means satisfied and ✗ means not satisfied.

Checklist item	Ours	Unify-Agent	GenSearcher
Harbor docks show typical Victorian iron cranes	✗	✗	✗
Workers wear rough shirts, baggy pants, and sun hats	✓	✗	✗
Steam locomotives and railroad tracks are visible in the scene	✗	✗	✗
Saltpeter shipments appear as white crystals packed in burlap bags	✗	✗	✗
The mine owner wears British formal attire and a pocket watch	✗	✗	✗
Harbor buildings are colonial-style masonry structures	✓	✓	✗
A steam freighter at anchor can be seen in the distance	✓	✓	✓
The port surroundings show a typical desert landscape	✓	✓	✗
The sky appears misty in the early morning	✓	✗	✓
A British merchant-bank sign is visible on the pier	✗	✗	✗
KCS	0.500	0.300	0.200

A.1.4 MOOREA: NONI HARVESTING

Figure 8 presents a WorldGenBench-Humanities case about elderly Granny Maria picking noni fruit in the Paopao Valley on Moorea in July 1975. The full evaluation prompt is:

WorldGenBench-Humanities Prompt

```
In July 1975, in the Paopao Valley on the island of Moorea, elderly
Granny Maria was picking noni fruit.  The ripe noni fruits from this
orchard, which has been passed down from generation to generation,
give off a special odor.  She is going to make the fruits into a
traditional medicine for the upcoming Harvest Festival celebrations.
```

The prompt requires both agricultural activity and Polynesian island grounding: volcanic valley terrain, traditional floral clothing, ripe white noni fruit, a woven rattan basket, breadfruit and banana trees, a local flower garland, stone orchard fences, red laterite soil, traditional storage sheds, and a distant Pacific view. Our RL checkpoint obtains KCS 0.400, while Unify-Agent obtains 0.100 and GenSearcher obtains 0.000. Our result satisfies four checklist points. First, the background shows steep volcanic island valley terrain. Second, white ripe noni fruit is scattered on the ground. Third, the character wears a local flower garland. Finally, the distant view includes the Pacific Ocean, grounding the scene on Moorea. The remaining missed points are traditional floral dress, a woven rattan basket, breadfruit and banana trees, stone fences, red laterite soil, and traditional storage sheds.



Figure 8: Case study on WorldGenBench-Humanities. Our model better grounds the Moorea noni-harvesting scene in volcanic valley terrain, scattered ripe noni fruit, a local flower garland, and a distant Pacific view.

Table 7: **Checklist evaluation for the Moorea case.** Each row is one score point from the benchmark checklist. Blue rows mark the points satisfied by our model. ✓ means satisfied and ✗ means not satisfied.

Checklist item	Ours	Unify-Agent	GenSearcher
Scenery reflects typical volcanic island valley terrain	✓	✓	✗
Characters wear traditional Polynesian flower dresses	✗	✗	✗
The ground is strewn with white ripe noni fruit	✓	✗	✗
The character carries a woven rattan basket	✗	✗	✗
Breadfruit and banana trees are visible in the scene	✗	✗	✗
The character wears a local flower garland	✓	✗	✗
A stone fence divides the orchard	✗	✗	✗
The ground shows laterite formed by weathered volcanic rocks	✗	✗	✗
Traditional storage sheds are visible in the scene	✗	✗	✗
The Pacific Ocean is visible in the distance	✓	✗	✗
KCS	0.400	0.100	0.000

A.1.5 SANTO DOMINGO: SUGAR WORKSHOP

Figure 9 presents a WorldGenBench-Humanities case about a sugar worker in an ancestral workshop in Santo Domingo in 1961. The full evaluation prompt is:

WorldGenBench-Humanities Prompt

```
On a hot afternoon in 1961, Luis, a sugar worker in the old town of
Santo Domingo, is busy in his ancestral sugar workshop.  The air is
filled with the sweet smell of fresh sugar cane from the Caripo Valley.
Suddenly, there is a noise in the distance, as news of the recent fall
of the Trujillo regime reaches the old neighborhood.
```

The prompt requires both traditional sugar-production details and Santo Domingo's historical setting: a wooden sugar-cane press, sweat-stained white cotton work clothing, Spanish colonial limestone walls, bagasse on the floor, earthenware vats, arched doors and windows, Trujillo-period leader portraits, steam from the sugar cooker, Caribbean-style buildings outside the window, and bamboo sieves and barrels. Our RL checkpoint obtains KCS 0.400, while Unify-Agent obtains 0.200 and GenSearcher obtains 0.100. Our result satisfies four checklist points. First, the workshop includes a traditional wooden sugar-cane press. Second, earthenware vats used for collecting cane sugar are visible. Third, steam rises from the sugar cooker, making the production process concrete. Finally, Caribbean-style buildings appear outside the window, grounding the scene in Santo Domingo. The remaining missed points are the sweat-stained white cotton shirt, Spanish colonial limestone walls, bagasse on the floor, arched doors and windows, Trujillo-period leader portraits, and bamboo sieves and barrels.



Figure 9: Case study on WorldGenBench-Humanities. Our model better grounds the Santo Domingo sugar-workshop scene in the wooden cane press, earthenware vats, visible steam from the sugar cooker, and Caribbean-style exterior buildings.

Table 8: **Checklist evaluation for the Santo Domingo case.** Each row is one score point from the benchmark checklist. Blue rows mark the points satisfied by our model. ✓ means satisfied and ✗ means not satisfied.

Checklist item	Ours	Unify-Agent	GenSearcher
The workshop has a traditional wooden sugar-cane press	✓	✓	✓
The worker wears a sweat-stained white cotton short-sleeved shirt	✗	✗	✗
The walls use typical Spanish colonial limestone construction	✗	✗	✗
Bagasse is scattered on the ground	✗	✗	✗
Earthenware vats used to collect cane sugar are visible	✓	✗	✗
Windows and doors use a typical Spanish arched design	✗	✗	✗
Trujillo-period leader portraits are visible on the walls	✗	✗	✗
Steam can be seen rising from the sugar cooker	✓	✗	✗
Typical Caribbean-style buildings can be seen outside the window	✓	✓	✗
Bamboo sieves and barrels are placed in the workshop	✗	✗	✗
KCS	0.400	0.200	0.100

A.2 Case Studies

The following trajectories illustrate how ToolArtist reasons, retrieves textual and visual evidence, and generates images.

A.2.1 Thomas Edison

Case Study: Thomas Edison

Query:
Thomas Edison's greatest invention.

Step 1. Reasoning + Text Search:
I first need to verify which invention is most widely regarded as Thomas Edison's greatest contribution, ensuring factual accuracy for subsequent image retrieval and generation.

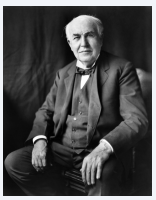
Thomas Edison most important invention
greatest invention of Thomas Edison

Summary:
The practical incandescent light bulb is Edison's most iconic invention. He improved earlier designs by developing a durable filament and the electric power system required for practical use. The phonograph, however, is often described as his first great invention and remained his personal favorite.

Step 2. Reasoning + Image Search:
Because the phonograph is commonly described as Edison's first great invention while the light bulb is his most famous contribution, the final image should represent both interpretations.

Thomas Edison official portrait
Edison phonograph original 1877 invention
Edison practical incandescent light bulb 1879

Summary:


[IMAGE 1]



[IMAGE 2]


[IMAGE 3]

Step 3. Reasoning + Caption:
To reflect both interpretations, the generated image should depict Edison beside the phonograph and the practical incandescent light bulb inside his laboratory.

Step 4. Generating:
Vintage 1900s realistic historical photograph of Thomas Edison, use [IMAGE 1] as reference for his exact facial features, clothing and appearance, standing beside his two most iconic inventions: the Edison phonograph, using [IMAGE 2] as reference for its original design, on his left, and the practical incandescent light bulb, using [IMAGE 3] as reference for its shape, on his right, set in the interior of his West Orange laboratory, warm ambient lighting, authentic 1900s photo aesthetic, high detail, 4K resolution.



A.2.2 WHITECHAPEL, LONDON

This trajectory corresponds to the Whitechapel comparison in Section A.1.2. The system prompt is omitted. Reasoning text is shown without <think> tags, and tool calls are displayed as tool names and query lists rather than raw JSON/XML wrappers. Tool observations, the final generation caption, and the generated image are retained.

Case Study: Whitechapel, London

```
Query:
User: On an overcast November day in 1889, a working woman in the Whitechapel district of East
    London was walking home. A thick fog shrouded the narrow streets, and the gas streetlights
    glowed dimly. She picks up her pace as she walks past a crowded room of low-cost apartments,
    and the smell of factory smoke fills the air. It had been only a year since the Jack the Ripper
     case, and the neighborhood was still shrouded in fear.

Step 1. Reasoning + Image Search:
First, I need to get accurate reference images for the key elements of the scene: 1889 Whitechapel
    overcast foggy street with gas lamps, late 19th century East End working woman clothing, 1880s
    Whitechapel low-cost apartment building exterior, and Jack the Ripper related period atmosphere
     elements to ensure historical accuracy.

image_search:
[
  "1889 Whitechapel East London overcast foggy gas street",
  "1880s British East End working class woman clothing",
  "1880s Whitechapel low cost apartment building exterior",
  "Jack the Ripper Whitechapel 1888 period atmosphere"
]
```

```
Summary:
query : 1889 Whitechapel East London overcast foggy gas street
Result : <desp>NO_RELEVANT_INFO

(The retrieved document is a review of Christine L. Corton's London Fog: The Biography. It
    discusses the history of London fog, but does not specifically mention 1889, Whitechapel, East
    London, or gas streetlights.)</desp>[IMAGE 1] [reference_image]

query : 1880s British East End working class woman clothing
Result : <desp>Victorian working-class women's clothing (based on Victorian Working Women by
    Michael Hiley, drawing on Arthur Munby's archives)

General contrast with upper-class dress:
- Working women's clothes were worn to rags and rarely preserved in museums; the lace, silks,
    crinolines, and bustles associated with "Victorian fashion" belonged only to the wealthy
    minority.

Work-specific clothing:
- Mine workers (1840s): Women dragged mine carts wearing trousers with belts around waists and
    chains passing between their legs; often bare-chested or half-naked underground.
- Pit brow / surface mine workers (Wigan, 1860s): Wore trousers -- photographed in studio portraits
     in their work clothes. Not universal across all mines; some women at other mines wore skirts.
- Fisherwomen / limpet collectors (Yorkshire, 1860s): Wore skirts and petticoats tucked up into
    their belts (front and back) to expose legs up to the thigh while wading.
- Mudlarks: Waded through Thames refuse; legs exposed up to the thigh.
- Housemaids (early 1860s): Dressed in their "best" for photographs, but hands visibly worn and
    roughened from labor.
- Milkwomen (London, 1864-1872): Street dress while carrying 80-100 lbs of milk.
- Washerwomen: Developed muscular arms from heavy wet laundry work.

Class & modesty norms:
- In Victorian London, a "respectable" woman exposing her legs (even a glimpse of calf/stocking)
    was read as a signal of prostitution. Working-class women's modesty was not considered a social
     concern -- fisherwomen and mudlarks routinely exposed their legs without the same stigma
    applying.
- Trousers on women caused public indignation ("Bloomer girls"), yet working women wore them out of
     practical necessity.

Key sources:
- Victorian Working Women: Portraits from Life, Michael Hiley
- Arthur Munby's diaries, sketches, and photo collection (now on microfilm at Trinity College,
    Cambridge)
- Henry Mayhew, London Labour and the London Poor (1851-1861) -- detailed descriptions of street
    workers' dress and lives</desp>[IMAGE 2] [reference_image]
query : 1880s Whitechapel low cost apartment building exterior
Result : <desp>Rowton House, Whitechapel -- Low-Cost Lodging Building (opened 1902, not 1880s)

- Location: 81 Fieldgate Street, Whitechapel, London
- Built: Opened 1902 (part of a series of six Rowton Houses built 1892-1905)
- Purpose: Low-cost accommodation ("hotels for working men") for single working men; response to
    London's housing crisis of the 1880s
- Architect: Harry Bell Measures
- Exterior description:
  - 7 storeys, rising 60 feet; distinctive roofline; local landmark
  - Pressed Leicester facing bricks (Fletton bricks on inward faces); concrete-clad steel
    construction
  - Rows of narrow slit windows to cubicle floors (deliberately ungrouped to give each sleeper a
    window)
  - Semi-circular windows on first two floors (dayrooms)
  - Relieved with gables, turrets, and pink terracotta dressings
  - Large projecting entrance porch flanked with octagonal finials, also in terracotta
  - Two-parallelogram plan with inner courtyard open at one end for air/light circulation
- Style: Characteristic red-brick institutional block; "new and easily recognisable style" in
    London
- Capacity: 816 cubicles
- Cost per night: 6d (slightly above common lodging houses at ~4d)

Note: The building itself opened in 1902, but Rowton Houses were conceived as a direct response to
    the 1880s housing crisis. The first in the series (Vauxhall) opened 1892.</desp>[IMAGE 3] [
    reference_image]
```

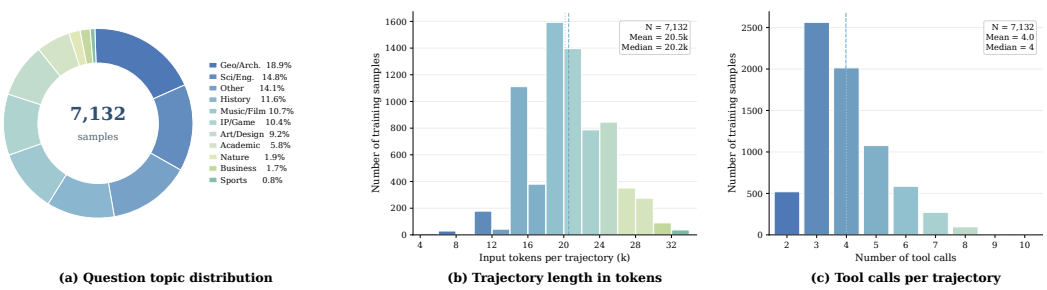
```
query : Jack the Ripper Whitechapel 1888 period atmosphere
Result : <desp>Whitechapel in 1888 -- Period Atmosphere

Social & Economic Conditions
- Victorian era; London was wealthy capital of a powerful empire but deeply unequal
- Whitechapel was one of the poorest districts in East End London
- Charles Booth's 1880s poverty map marked Whitechapel predominantly black and blue ("very poor" /
    "semi-criminal")
- Overcrowded: some families sharing a single room; houses badly ventilated and damp
- Jobs for unskilled workers poorly paid and inconsistent
- Intense poverty drove adults and children into alcohol abuse, sex work, and crime

Population & Community
- London's wealth drew migrants from across Britain, Ireland, and Europe -- housing severely
    strained
- Large Jewish community; majority worked in the tailoring industry
- Thousands of Jewish refugees fleeing Eastern European persecution were settling in Whitechapel
    and Spitalfields at the time of the murders -- triggering antisemitic suspicion

Fear & Public Atmosphere
- Murders spread fear among Londoners and shocked the nation
- Frenzied newspaper coverage; police refused to share investigation details, fuelling exaggerated
    theories
- Lord Cranbrook, 2 October 1888: "More murders at Whitechapel, strange and horrible. The
    newspapers reek with blood"
- Locals protested at Leman Street police station demanding the killer be caught
- 1888 cartoon mocked Metropolitan Police's failure to solve the crime

Media & Legend
- First transatlantic telegraph cable (laid 1866) made the story international
- Killings became "as much legend as fact" -- part of the classic image of a dark, foggy Victorian
    East End</desp>[IMAGE 4] [reference_image]

Step 2. Generating:
<draw_caption>Overcast November day 1889, East London Whitechapel, reference [IMAGE 1] for the 1880
    s Victorian street atmosphere with thick fog, dim glowing gas street lamps, cobblestone streets
    , old brick buildings. A working class woman in 1880s British East End clothing, reference [
    IMAGE 2] for her tattered simple work dress and headscarf, walking briskly on the sidewalk,
    picking up her pace as she passes the crowded low-cost apartments building from [IMAGE 3]. The
    air is thick with the smell of factory smoke, muted cool grey and brown color palette,
    historical period accurate, cinematic lighting, photorealistic, 19th century street photography
     style</draw_caption>
[reference_image]
```



A.3 SFT Data Distribution and Examples

A.3.1 Data Distribution

Figure 10 summarizes the retained SFT corpus. The final corpus contains 7,132 trajectories after conversion and filtering. The topic distribution is diverse: the largest category is Geo/Architecture (18.9%), followed by Sci/Engineering (14.8%), Other (14.1%), History (11.6%), Music/Film (10.7%), IP/Game (10.4%), Art/Design (9.2%), Academic (5.8%), Nature (1.9%), Business (1.7%), and Sports (0.8%). Trajectory length is centered around long multimodal contexts, with a mean of 20.5k and a median of 20.2k input tokens. Tool use is also compact: the mean and median number of tool calls are both 4.0. This distribution suggests that the SFT corpus is not a collection of one-step image prompts; it mainly consists of multi-turn search-and-generation trajectories with substantial retrieved evidence and visual context.



(a) Question topic distribution (b) Trajectory length in tokens (c) Tool calls per trajectory

Figure 10: Distribution of retained SFT trajectories. The figure reports topic categories, trajectory token length, and number of tool calls.

A.3.2 SFT Trajectory Examples

The following SFT rollout starts at the user question. The system prompt is omitted. The original text is kept, while image placeholders are replaced by the actual reference and generated images.

SFT Example: WGS 84 vs GRS 80 Raw Rollout

```
[message 1 | role=user]
User: A technical diagram comparing the reference ellipsoids of 'WGS
84' and 'GRS 80'; the illustration highlights the difference in the
semi-minor axis. A data plaque lists the exact length of the
semi-minor axis in meters for WGS 84 and the name of the international
organization that maintains the standard; both the length and the
organization name must be correct.
[message 2 | role=assistant]
<think> First, I need to obtain the accurate exact semi-minor axis
values for WGS 84 and GRS 80, as well as the official organization
that maintains the WGS 84 standard, to ensure the data on the diagram
is fully correct. I will use text search to get this authoritative
information first. </think>
<tool_call> { "name": "text_search", "arguments": { "query": [
"WGS84 GRS80 reference ellipsoid semi-minor axis exact length meters",
"WGS84 standard maintaining international organization", "GRS80
semi-minor axis precise value meters" ] } } </tool_call>
[message 3 | role=tool]
query : WGS84 GRS80 reference ellipsoid semi-minor axis exact length
meters
Result : World Geodetic System - Wikipedia | Semi-minor axis (b)
comparison - WGS 84 vs GRS 80:
GRS 80: semi-major axis 6,378,137.0 m; semi-minor axis
6,356,752.314140 m. WGS 84: semi-major axis 6,378,137.0 m;
semi-minor axis 6,356,752.314245 m.
Both share the same semi-major axis, but differ in inverse flattening,
resulting in a 0.105 mm difference in the semi-minor axis. WGS 84
semi-minor axis is a computed parameter: b = a x (1 - f) =
6,356,752.3142 m.
query : WGS84 standard maintaining international organization
Result : WGS84 - World Geodetic System 1984 - geodesy.science - IAG
website | WGS84 is maintained by the U.S. Department of Defense,
specifically the National Geospatial-Intelligence Agency (NGA). The
document also states that WGS84 is aligned with ITRF and that ITRF is
maintained under the broader framework of global geodesy, with IAG and
GGOS playing key roles.
query : GRS80 semi-minor axis precise value meters
Result : NOAA Technical Memorandum NOS NGS 97 Storage and Use of
GRS80 Ellipsoid Parameters in the Modernized National Spatial
Reference System.
```

Formula (11a): b = a(1 - 1/f^-1). Substituting a = 6378137 m and f^-1 = 298.257222101 gives b = 6356752.3141 m.

[message 4 | role=assistant]

<think> Now I have the accurate data points needed: WGS 84 semi-minor axis = 6356752.3142 m, maintained by the U.S. National Geospatial-Intelligence Agency (NGA); GRS 80 semi-minor axis = 6356752.3141 m, with a 0.105 mm difference between the two. Next I need to search for reference technical diagrams comparing reference ellipsoids, to get the correct visual structure for the illustration. </think>

<tool_call> { "name": "image_search", "arguments": { "query": ["WGS84 GRS80 reference ellipsoid comparison technical diagram", "reference ellipsoid semi-minor axis illustration technical diagram", "geodesy ellipsoid comparison diagram with data plaque"] } }
</tool_call>

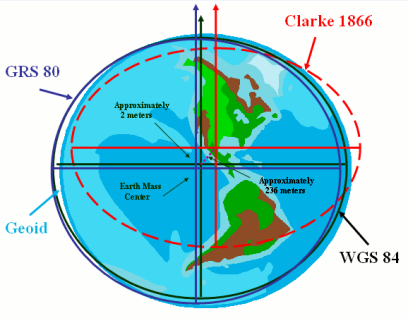
[message 5 | role=tool]

query : WGS84 GRS80 reference ellipsoid comparison technical diagram

Result : WGS84 vs GRS80 reference ellipsoid comparison data

Both share the same semi-major axis: 6 378 137 m. Semi-minor axis: GRS 80 = 6 356 752.3141 m; WGS 84 = 6 356 752.3142 m. The flattening differs by a tiny amount, so the maximum ellipsoid height difference at the pole is only 0.1 mm.

[external image 1]

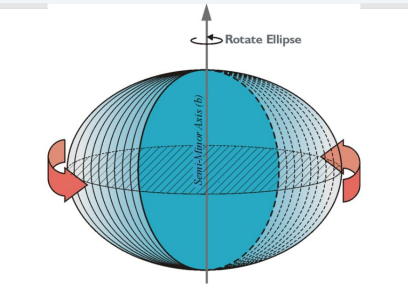


query : reference ellipsoid semi-minor axis illustration technical diagram

Result : Reference Ellipsoid Semi-minor Axis - Key Technical Information

Semi-minor axis = half of the minor axis of an ellipse; for a reference ellipsoid, it is the north-south axis around which the ellipse is rotated to form a solid. The ellipsoid bulges at the equator and is flattened at the poles. An ellipsoid is completely defined by its semi-major axis and flattening.

[external image 2]

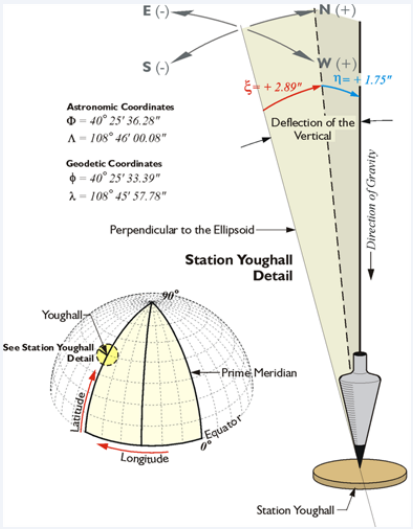


query : geodesy ellipsoid comparison diagram with data plaque

Result : GRS80 vs WGS84 Ellipsoid Comparison

Both ellipsoids have semi-major axis 6,378,137 m. GRS80 inverse flattening is 298.257222101; WGS84 inverse flattening is 298.257223563. GRS80 was adopted by IUGG in 1979 and WGS84 is the reference for the GPS datum.

[external image 3]



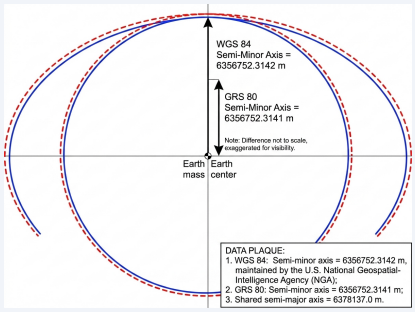
[message 6 | role=assistant]

<think> Okay, now I have the reference diagrams and all the correct data. I need to create a technical diagram that compares WGS84 and GRS80 reference ellipsoids, highlights the semi-minor axis difference, and has a data plaque with the correct numbers and maintenance org. First, use the first reference image [IMAGE1] as the base for the ellipsoid comparison layout, the second [IMAGE2] for the semi-minor axis labeling. Then the prompt should specify: technical diagram, side-by-side comparison (exaggerate the tiny semi-minor axis difference for visibility, note it's 0.105mm difference), label semi-minor axis clearly for both, a data plaque with: WGS84 semi-minor axis = 6356752.3142 m, maintained by US National Geospatial-Intelligence Agency (NGA); GRS80 semi-minor axis = 6356752.3141 m; note that semi-major axis is identical 6378137.0 m for both, clean white background, engineering/geodesy technical illustration style, clear labels, no clutter. Yes that makes sense. Let's call the draw tool now. </think>

<tool_call> { "name": "draw", "arguments": { "prompt": "Technical geodesy diagram comparing WGS 84 and GRS 80 reference ellipsoids, based on the layout of [IMAGE1] and using the semi-minor axis labeling style of [IMAGE2]. Exaggerate the tiny 0.105 mm difference in semi-minor axis for visibility (with a small note that the difference is not to scale). Clearly label the semi-minor axis (polar radius) for both ellipsoids with callouts. Include a formal data plaque at the bottom right listing: 1. WGS 84: Semi-minor axis = 6356752.3142 m, maintained by the U.S. National Geospatial-Intelligence Agency (NGA); 2. GRS 80: Semi-minor axis = 6356752.3141 m; 3. Shared semi-major axis = 6378137.0 m. Clean white background, professional technical illustration style, sharp clear text labels, no extraneous elements, all numerical values accurate.", "images": ["https://vdatum.noaa.gov/images/docs/clarke1866_wgs84_grs80.gif", "https://courses.ems.psu.edu/geog862/sites/geog862/files/images/Lesson05/Ellipsoid.JPG"] } } </tool_call>

[message 7 | role=tool]

Result : [internal image 4]



[message 8 | role=assistant]

<think> Got it, let's check the generated image first. First, the semi-minor axis values are correct: WGS84 is 6356752.3142 m, GRS80 is 6356752.3141 m, right. The data plaque has the correct organization: U.S. National Geospatial-Intelligence Agency (NGA) for WGS84, perfect. The difference is exaggerated with a note that it's not to scale, which is good because the actual difference is only 0.1mm, you can't see it otherwise. The shared semi-major axis is listed correctly as 6378137.0 m. The labels are clear, it's a proper technical diagram, the semi-minor axis is highlighted as requested. All requirements are met. Now I just need to output the final result with the path, and a short description. </think>

<|box_start|>Generated Technical Diagram: WGS 84 vs GRS 80 Reference Ellipsoid Comparison. <|box_end|>

SFT Example: Sigewinne and Bailu Raw Rollout

[message 1 | role=user]

User: In a whimsical underwater nursery, the Melusine nurse Sigewinne from Genshin Impact and the young dragon-healer Bailu from Honkai: Star Rail are mixing a glowing blue potion; a medical chart on the wall must correctly display Sigewinne's official Vision element.

[message 2 | role=assistant]

<think> First, I need to confirm Sigewinne's official Vision element, then search for reference images of both characters and the cute underwater nursery mood so the final image stays canonically correct and visually coherent. </think>

<tool_call> { "name": "text_search", "arguments": { "query": ["Sigewinne Genshin Impact official Vision element", "Sigewinne confirmed Vision type Genshin"] } } </tool_call>

[message 3 | role=tool]

Sigewinne's Vision is Hydro.

[message 4 | role=assistant]

<think> Now I need image references for Sigewinne, Bailu, the Hydro symbol, and the glowing potion. I'll search for those and then build the final caption. </think>

<tool_call> { "name": "image_search", "arguments": { "query": ["Sigewinne Genshin Impact official character art", "Bailu Honkai: Star Rail official character art", "Genshin Impact official Hydro Vision symbol", "glowing blue magical potion fantasy"] } } </tool_call>

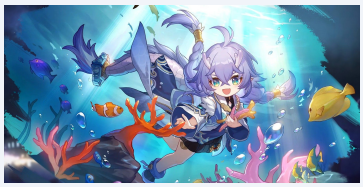
[message 5 | role=tool]

query : Sigewinne Genshin Impact official character art

Query result: Sigewinne's official character image references and details: - Hydro Vision - Bow user - Melusine nurse from Fontaine - Character art reference available.



query : Bailu Honkai: Star Rail official character art
Query result: Bailu character reference image and appearance details.



query : whimsical cute underwater nursery interior design
Query result: Underwater nursery references with pastel colors, ocean decorations, and magical cozy atmosphere.



query : Genshin Impact official Hydro Vision symbol
Query result: Hydro Vision reference image.



query : glowing blue magical potion fantasy
Query result: Blue glowing fantasy potion reference.



[message 6 | role=assistant]
<think> I have the key references now. I can write a single generation prompt that uses Sigewinne and Bailu's exact appearances, the Hydro chart, and the glowing blue potion inside a cute underwater nursery. </think>
<tool_call> { "name": "draw", "arguments": { "prompt": "Whimsical cozy underwater nursery scene. " "Soft blue light filters through the water with small bubbles floating " "gently. Walls have cute watercolor sea creature decals, warm light " "wood furniture, woven storage baskets, and a pastel color palette." "Two cute anime girls stand side by side at a small wooden table: " "Sigewinne the Melusine nurse matching [IMAGE 1], and Bailu the young " "dragon healer matching [IMAGE 2]. They are mixing a glowing blue " "magic potion matching [IMAGE 4]." "A framed medical chart is pinned on the wall behind them. The chart " "prominently displays the official Hydro Vision symbol matching " "[IMAGE 3], correctly indicating Sigewinne's Hydro element." "HoYoverse anime aesthetic, soft shading, high detail, 4K resolution, " "whimsical magical atmosphere.", "images": ["sigewinne-ref.png", "bailu.jpg", "hydro.jpg", "potion.jpg"] } }
</tool_call>



```
[message 8 | role=assistant]
<think> The result is consistent with the requested anime style.
Sigewinne and Bailu are correctly depicted, and the medical chart
shows Hydro as required.  </think>
Generated image meets all requirements.
```

A.4 Prompt Displays

We provide the complete prompts for the teacher agent, ToolArtist, and the LLM reader.

A.4.1 Teacher Agent

Teacher Agent Full Prompt

```
SYSTEM_PROMPT = """You are a "retrieval-augmented image generation
    Agent". Your task is to proactively search for web information and
    reference images based on the user's image generation request,
    compile a high-quality image generation prompt, and then call the
    image generation tool to produce the image.

Action principles:
- For uncertain details involving real people, places, logos, clothing,
     architecture, IP styles, etc., prioritize using text_search to
    obtain textual descriptions, then use image_search to fetch
    reference images.
- Images returned by image_search will be shown to you directly in a
    multimodal way. Judge quality and relevance based on the image
    content, and pick the most suitable ones as references.
- Before calling draw, distill the retrieved information into a clear
    visual description: subject, scene, style, composition, color,
    material, action, lighting. Do not dump raw search results into the
     prompt.
- Generated images may not be perfect. After generation, carefully
    review the returned image. If unsatisfied, adjust the prompt or
    swap/add/remove reference images, and call draw multiple times to
    iteratively refine until the result is satisfactory. Do not stop
    after a single generation.
- Finally, wrap the answer with <|box_start|> and <|box_end|>, clearly
     stating the saved path of the final generated image or the reason
    for failure.

Reference image citation format (strict):
- When referring to reference images in the prompt field of draw, you
    can **only** use fixed tokens: [IMAGE1], [IMAGE2], [IMAGE3], ...
    (1-indexed, corresponding to the order of the images array).
- It is strictly forbidden to use natural language expressions such as
     "Image 1", "the first image", "the 1st image", "the first
    reference image", etc.
- Example: 'Paste the character from [IMAGE1] onto the background of [
    IMAGE2]'."""
```

```
USER_PROMPT = """A conversation between User and Assistant. The user
    provides an image generation request. The assistant searches for
    information and reference images, then generates the image by
    calling tools.

<tools>
{
  "name": "text_search",
  "description": "Batch web search tool. Takes an array of queries;
    for each query, uses Google search to obtain candidate pages and
    uses Jina Reader to fetch the title and body Markdown of the rank-1
     page (very long content is truncated and marked with a truncated
    flag).",
  "parameters": {
    "type": "object",
    "properties": {
      "query": {"type": "array", "items": {"type": "string"}, "
    description": "Array of search queries; you can pass multiple
    complementary queries at once"}
    },
    "required": ["query"]
  }
}
{
  "name": "image_search",
  "description": "Batch image search tool. Takes an array of queries
    and returns structured information such as image titles, image URLs
    , width and height. Images are also shown to you in a multimodal
    way. The returned results[i].url can be directly placed into draw's
     images array.",
  "parameters": {
    "type": "object",
    "properties": {
      "query": {"type": "array", "items": {"type": "string"}, "
    description": "Array of image search queries; you can pass multiple
     complementary queries at once"}
    },
    "required": ["query"]
  }
}
{
  "name": "draw",
  "description": "Generates an image from a prompt, optionally with
    several reference images. The images array corresponds to [IMAGE1],
     [IMAGE2], ... in order; each item can be a local file path or a
    remote URL, and the tool will distinguish between them internally.
    Returns the local path where the image was saved and shows the
    generated image back to you in a multimodal way.",
  "parameters": {
    "type": "object",
    "properties": {
      "prompt": {"type": "string", "description": "Image generation
    prompt. References to reference images must use fixed tokens like [
    IMAGE1], [IMAGE2], etc.; any natural-language reference is
    forbidden"},
      "images": {"type": "array", "items": {"type": "string"}, "
    description": "Reference images, corresponding to [IMAGEk] in order
    . A single image should also be written as an array. Each item is
    either a local file path or a remote URL"}
    },
    "required": ["prompt"]
  }
}
```

```
</tools>

Workflow: think → call tool → wait for response → continue thinking
    or give an answer. Multiple rounds of calls are allowed.
Important: After generating an image, carefully review the returned
    image. If unsatisfied, adjust the prompt or reference images and
    call draw again, iterating until satisfied.

Example flow structure (only shows the call sequence; specific content
     should be filled in based on actual needs):

Analyze the user's request and plan what information and images to
    search for
<tool_call>
{"name": "text_search", "arguments": {"query": ["keyword1", "keyword2
    "]}}
</tool_call>
<tool_response>Search results</tool_response>
Extract useful information and decide which reference images to search
     for
<tool_call>
{"name": "image_search", "arguments": {"query": ["image keyword1", "
    image keyword2"]}}
</tool_call>
<tool_response>Image search results (you can see the images directly)
    </tool_response>
Pick suitable reference images based on what you see, integrate the
    information and write the image generation prompt
<tool_call>
{"name": "draw", "arguments": {"prompt": "..., referencing the styling
     of [IMAGE1], placing it into the scene of [IMAGE2]", "images": ["
    https://example.com/ref1.jpg", "https://example.com/ref2.jpg"]}}
</tool_call>
<tool_response>Generation result (you can see the generated image
    directly)</tool_response>
Check the generation result. If unsatisfied, adjust the prompt and
    call draw again; if satisfied, output the final answer
<|box_start|>Final result description (including the saved path of the
     generated image)<|box_end|>

Notes:
- Each round outputs only one tool_call or the final box answer
- tool_call must be valid JSON, including name and arguments
- References to reference images in draw's prompt can only use fixed
    tokens like [IMAGE1], [IMAGE2], ...; natural-language references
    such as "the first image" or "Image 1" are forbidden

User: """
```

A.4.2 TOOLARTIST

ToolArtist Full Prompt

```
SYSTEM_PROMPT = """You are a "retrieval-augmented image-generation
    agent". Given the user's image-generation request, you actively
    search the web for information and reference images, then generate
    a high-quality image yourself.

Operating principles:
- For uncertain details about real people, places, logos, clothing,
    architecture, IP styles, etc., use text_search first to gather
    textual facts, then image_search to pull reference images.
```

```
- Images returned by image_search are shown to you directly. Judge
    their quality and relevance from what you see, and pick the most
    suitable ones as references.
- Before generating, distill the retrieved information into a clear
    visual description: subject, scene, style, composition, color,
    material, action, lighting. Do NOT dump raw search results into the
     pre-generation caption.
- The generated image may not be perfect. After generation, inspect it
     carefully; if unsatisfied, adjust the caption or swap/add/remove
    references and iterate until the result is good.

Reference-image citation format (strict):
- In the pre-generation caption you may ONLY refer to images via the
    fixed tokens [IMAGE 1], [IMAGE 2], [IMAGE 3], ... (1-indexed,
    matching the global numbering you see in tool responses).
- Natural-language references such as "Image 1", "the first image", "
    the first reference image", etc. are forbidden.
- Example: 'place the character from [IMAGE 1] onto the background of
    [IMAGE 2]'.

The user provides an image-generation request. You will search for
    information and reference images by calling tools, and then
    generate the image yourself.

<tools>
{
  "name": "text_search",
  "description": "Batch web search. Takes an array of queries; for
    each query, runs a Google search to find candidate pages and uses
    Jina Reader to fetch the rank-1 page's title and Markdown body (
    long pages are truncated with a `truncated` marker).",
  "parameters": {
    "type": "object",
    "properties": {
      "query": {"type": "array", "items": {"type": "string"}, "
    description": "Array of search queries; you may pass multiple
    complementary queries at once."}
    },
    "required": ["query"]
  }
}
{
  "name": "image_search",
  "description": "Batch image search. Takes an array of queries;
    returns image titles, URLs, sizes, and short summaries. Each image
    is also rendered to you directly. Every returned image is assigned
    a global [IMAGE n] index that you must use when referring to it
    later in the pre-generation caption.",
  "parameters": {
    "type": "object",
    "properties": {
      "query": {"type": "array", "items": {"type": "string"}, "
    description": "Array of image-search queries; you may pass multiple
     complementary queries at once."}
    },
    "required": ["query"]
  }
}
</tools>

Workflow: think → call a tool → wait for response → keep thinking or
     proceed to image generation. Multiple rounds allowed.
```

```
How to generate an image:
- When you are ready to generate, write a single pre-generation
    caption inside <|extra_50|> ... <|extra_51|>. This caption is the
    full visual description of the image you are about to produce:
    subject, scene, style, composition, color, material, action,
    lighting.
- Inside the caption, reference selected reference images by their
    global [IMAGE n] tokens (the same n that appeared in the tool
    responses). Only include references you actually want the model to
    use; omit unused images entirely.
- Immediately after closing the caption with <|extra_51|>, emit the
    image tokens for the generated image.

Example flow (structure only; fill in real content per request):

<|extra_60|>Analyse the user's request and plan what information and
    reference images are needed.<|extra_61|>
<tool_call>
{"name": "text_search", "arguments": {"query": ["keyword 1", "image
    keyword 2"]}}
</tool_call>
<tool_response>search results</tool_response>
<|extra_60|>Extract useful facts and decide which reference images to
    search.<|extra_61|>
<tool_call>
{"name": "image_search", "arguments": {"query": ["image keyword 1", "
    image keyword 2"]}}
</tool_call>
<tool_response>image results, each tagged with a global [IMAGE n]
    index and rendered visually</tool_response>
<|extra_50|>Combine the retrieved information and chosen references
    into the final visual description, citing only the [IMAGE n] tokens
     you want to use.<|extra_51|>
(generated image tokens)

Notes:
- Each tool_call must be valid JSON containing `name` and `arguments`.
- In the <|extra_50|>...<|extra_51|> caption, the ONLY allowed way to
    refer to a reference image is the fixed token [IMAGE 1], [IMAGE 2],
     ... -- never "Image 1", "the first image", "the first reference
    image", etc.
- Use the same global [IMAGE n] numbering that appears in the tool
    responses; do not renumber."""

USER_PREFIX = "User: "
```

A.4.3 LLM READER

LLM Reader Prompt

```
You are an information extraction assistant. Below is a query and a
piece of document content. Find the most relevant core information
from the document related to the query, condense and organize it, with
the following requirements:
1. Only keep facts, data, or opinions that are truly relevant to the
   query; remove unrelated content.
2. Present the information in compact bullet points or short
   paragraphs. Do not use pleasantries, restate the query, or include
   filler phrases such as "according to the document".
3. If the document contains no relevant information at all, output only
   NO_RELEVANT_INFO.
4. Preserve specific information from the original text, such as key
   numbers, names, times, and sources. Avoid being overly vague.
5. Answer in the same language as the query.
Template fields: [query] {query} and [document content] {doc}. The
reader directly outputs the condensed and organized information.
```