Ovis-Image Technical Report

Ovis Team, Alibaba Group

https://github.com/AIDC-AI/Ovis-Image
https://huggingface.co/AIDC-AI/Ovis-Image-7B

Abstract

We introduce **Ovis-Image**, a 7B text-to-image model specifically optimized for high-quality text rendering, designed to operate efficiently under stringent computational constraints. Built upon our previous Ovis-U1 framework, Ovis-Image integrates a diffusion-based visual decoder with the stronger Ovis 2.5 multimodal backbone, leveraging a text-centric training pipeline that combines large-scale pre-training with carefully tailored post-training refinements. Despite its compact architecture, Ovis-Image achieves text rendering performance on par with significantly larger open models such as Qwen-Image and approaches closed-source systems like Seedream and GPT4o. Crucially, the model remains deployable on a single high-end GPU with moderate memory, narrowing the gap between frontier-level text rendering and practical deployment. Our results indicate that combining a strong multimodal backbone with a carefully designed, text-focused training recipe is sufficient to achieve reliable bilingual text rendering without resorting to oversized or proprietary models.

Figure 1: Comprehensive illustration of the functional capabilities of Ovis-Image.

1 Introduction

Recent breakthroughs in image generation have significantly enhanced both the visual fidelity and controllability of synthesized images, exemplified by powerful text-to-image models (Wu et al., 2025a; Team, 2025b) and unified multimodal frameworks (OpenAI, 2025; Wang et al., 2025; Google DeepMind, 2025). Nevertheless, despite this rapid progress, achieving reliable, high-quality text rendering within images at low computational cost remains a persistent challenge. It requires models to simultaneously master fine-grained visual synthesis and robust language comprehension. In practice, strong text-rendering capabilities are typically found only in very large models (Wu et al., 2025a; Team, 2025b), which are difficult to deploy, or in closed-source systems (Seedream et al., 2025; Google DeepMind, 2025; OpenAI, 2025), which hinder integration, customization, and reproducibility.

In our prior work, Wang et al. (2025) introduced *Ovis-U1*, a 3B unified model that integrates multimodal understanding, text-to-image generation, and image editing within a single framework. It achieves competitive performance on multimodal understanding and generation benchmarks, approaching the behavior of proprietary models like GPT4o (OpenAI, 2025) in many practical scenarios. Nonetheless, with limited parameters, Ovis-U1 struggles with artifacts and hallucinations, and its in-image text quality still lags behind the best closed-source generators (Seedream et al., 2025; Google DeepMind, 2025).

Concurrently, recent text-to-image systems (Wu et al., 2025a; Team, 2025b) with especially strong text rendering ability tend to follow two patterns. First, they are tightly integrated with powerful multimodal understanding backbones (Bai et al., 2025), leveraging improved visual perception and OCR-like capabilities to reason about embedded text. Second, they scale model size to tens of billions of parameters. This combination yields impressive text-centric performance but substantially increases the cost of training and deployment in real-world applications. Motivated by these observations, we aim to design a specialized generator that prioritizes text rendering while preserving solid visual fidelity on general concepts and keeping computational cost acceptable.

Within this design space, we present **Ovis-Image**, a 7B text-to-image model that narrows the gap between efficiency and capability. We retain the successful pattern of coupling a strong multimodal backbone with a diffusion-based visual decoder, upgrade the multimodal backbone to the more capable Ovis 2.5 (Lu et al., 2025), and scale the vision-side generator to 7B parameters. Despite its compact size, Ovis-Image delivers text rendering performance comparable to much larger 20B-class open models such as Qwen-Image (Wu et al., 2025a) and approaches state-of-the-art closed-source generators like GPT4o and Gemini (Google DeepMind, 2025) on a range of text-centric tasks. In practice, it produces sharp, legible, and semantically consistent text for posters, banners, UI mockups, and scenes, while maintaining competitive image quality and prompt adherence on general-purpose generation.

In summary, Ovis-Image offers the following key advantages:

- **Strong text rendering at a compact 7B scale.** Ovis-Image is a 7B text-to-image model that delivers text rendering quality comparable to much larger 20B-class systems such as Qwen-Image and competitive with leading closed-source models like GPT4o in text-centric scenarios, while remaining small enough to run on widely accessible hardware.
- **High fidelity on text-heavy, layout-sensitive prompts.** The model excels on prompts that demand tight alignment between linguistic content and rendered typography (e.g., posters, banners, logos, UI mockups, infographics), producing legible, correctly spelled, and semantically consistent text across diverse fonts, sizes, and aspect ratios without compromising overall visual quality.
- **Efficiency and deployability.** With its 7B parameter budget and streamlined architecture, Ovis-Image fits on a single high-end GPU with moderate memory, supports low-latency interactive use, and scales to batch production serving, bringing near–frontier text rendering to applications where tens-of-billions–parameter models are impractical.

Taken together, these advantages indicate that high-quality in-image text rendering does not inherently require extremely large models. With appropriate architectural choices and a text-centric training pipeline built on top of a strong multimodal backbone, it is possible to approach frontier-level text-to-image performance within a compact 7B footprint.

2 Architecture

The architecture of Ovis-Image is presented in Fig. 2. Ovis-Image builds upon the architecture of Ovis-U1 (Wang et al., 2025), simplifying certain structures while increasing the parameters of the MMDiT backbone. A detailed summary of each module is provided in Tab. 1.

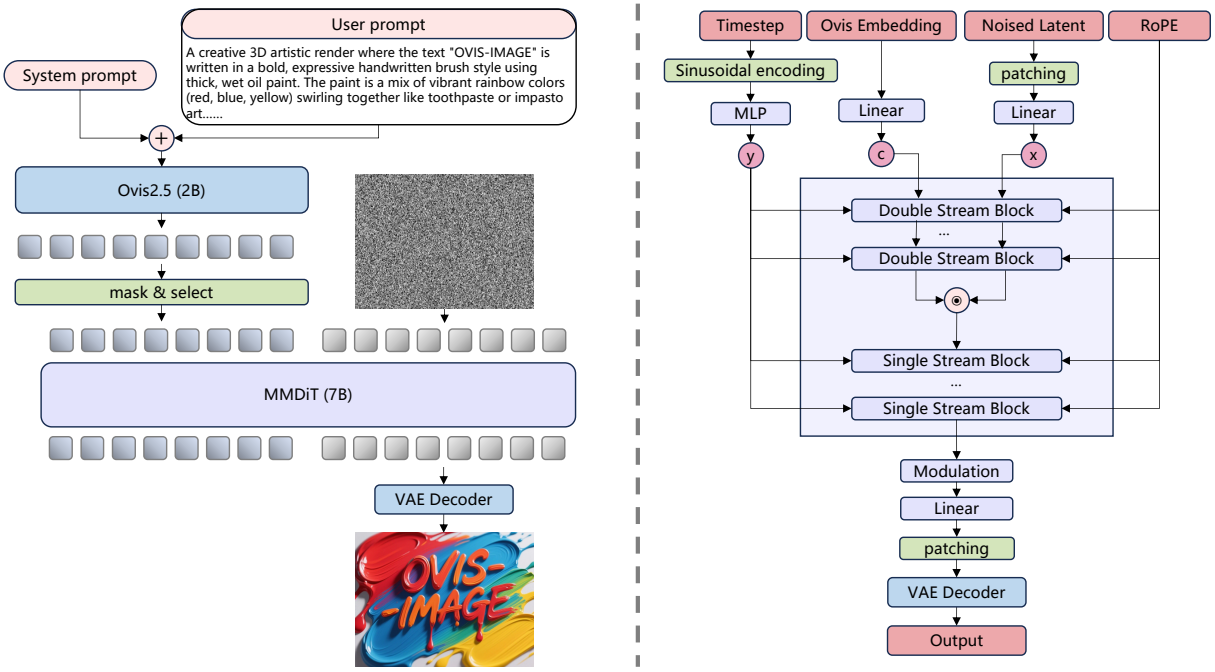

Figure 2: The overall architecture of Ovis-Image. The architecture of Ovis-Image builds upon Ovis-U1, enhancing its capabilities by increasing the parameters of MMDiT and streamlining the structural design to create a more efficient and refined overall framework.

Table 1: The model structure details of Ovis-Image.

Module	#Param. (B)	Pretrain
MMDiT	7.37	-
Text Encoder	2.57	AIDC-AI/Ovis2.5-2B
VAE	0.08	black-forest-labs/FLUX.1-schnell
Total	10.02	

MMDiT & VAE. Building upon Ovis-U1, we employ MMDiT (Esser et al., 2024) with RoPE (Su et al., 2024) as the visual decoder and use flow matching as the training objective. Inspired by Flux.1 Lite (Verdú & Martín, 2024), Ovis-Image incorporates a structure of 6 double-stream blocks and 27 single-stream blocks. To enhance model capacity, the number of attention heads has been increased to 24. SwiGLU (Dauphin et al., 2017) is utilized as the activation function. Additionally, we integrate the VAE model from FLUX.1-schnell (Labs, 2024) and keep it frozen throughout the training process.

Text Encoder. We use the Ovis (Lu et al., 2024) series as the text encoder in our framework. Unlike large language models like Qwen, Ovis is specifically trained on multimodal datasets, enabling superior alignment between visual and textual representations. To enhance computational efficiency, we adopt Ovis2.5-2B (Lu et al., 2025), which demonstrates better performance than Qwen2.5-VL-7B (Bai et al., 2025) on the OpenCompass benchmark suite. Unlike Ovis-U1, Ovis-Image simplifies the architecture by removing the refiner structure and directly utilizing the final hidden states of Ovis as the conditioning input for image generation.

3 Data Composition

3.1 Data for Pre-training

We pre-train Ovis-Image on a large, heterogeneous corpus of image–text pairs drawn from a mixture of web-scale, licensed, and synthetic sources. The corpus covers everyday photographs, illustrations, design assets, and UI-like mockups, with descriptions ranging from short captions to instruction-style prompts. To better align the text with the visual content, we perform large-scale recaptioning in both Chinese and English. In addition to generic visual content, we include data slices where text is a salient visual element, such as posters, banners, logos, and UI layouts.

To improve data quality, we apply a multi-stage filtering pipeline combining simple heuristics, lightweight models, and cross-modal consistency checks to remove corrupted images, severely mismatched or uninformative captions, and content that does not satisfy basic safety and policy requirements. We further perform coarse deduplication to reduce near-duplicate images and prompts. For text rendering in particular, we augment the corpus with synthetic samples generated by a rendering engine that composes clean typographic text into diverse backgrounds and layouts, providing the model with controlled examples of fonts, sizes, and placements.

3.2 Data for Supervised Fine-tuning

For supervised fine-tuning, we curate a higher-quality subset of image-text pairs, emphasizing clean visuals and well-formed prompts. Compared to the pre-training mixture, this corpus shifts toward higher-resolution images (typically at 1024 pixels) and covers a broad range of aspect ratios to better match real-world use cases. In addition to natural images, we include a moderate amount of synthetic data, which provides sharper details, more controlled layouts, and richer coverage of rare concepts. We perform simple balancing over content type, resolution, and aspect ratio to avoid overfitting to a few dominant patterns. Overall, this stage refines the model's ability to produce high-fidelity, instruction-following generations under consistent high-resolution constraints.

3.3 Data for DPO

For the DPO stage, we construct a preference dataset on top of the supervised distribution. About 90% of this pool consists of high-quality generations that cover common object categories and everyday scenes with strong aesthetic quality. These images are pre-filtered using an ensemble of automatic scorers (including HPSv3 Ma et al. (2025), CLIP Radford et al. (2021), PickScore Kirstain et al. (2023), and related metrics), so that only samples with both good visual appeal and reasonable prompt alignment are retained. The remaining $\sim$10% comes from an in-house collection focusing on design and creative content (e.g., posters, illustrations, and stylized compositions), which exposes the model to more structured layouts and non-photographic styles. For every selected prompt, we take the associated high-quality image from this pool as one candidate and generate a second candidate with the SFT model under the same text conditioning. Both images are then evaluated by multiple scoring models, and their scores are combined into an overall ranking. The higher-scoring image is treated as the winner and the other as the loser, yielding a preference pair for that prompt.

3.4 Data for GRPO

The GRPO stage operates on a prompt distribution that is deliberately different from the one used for DPO. Instead of broad, general-purpose generation prompts, we focus on a compact set of text-rendering prompts that stress the model's ability to place and stylize text in images. These prompts cover both Chinese and English, span a range of fonts and layouts (for example, posters, title cards, UI elements, and product labels), and vary in difficulty from short slogans to longer multi-line phrases. By concentrating the budget on this slice of the space, the GRPO data directly targets one of the known weaknesses of diffusion models, namely accurate and legible text generation.

4 Training

4.1 Training Infrastructure

Our training framework is built with PyTorch, utilizing Hybrid Sharding Data Parallel (HSDP) for efficient data parallelism and parameter sharding. To address memory limitations in larger models, we incorporate gradient checkpointing and activation offloading (Korthikanti et al., 2023). Training is conducted with bfloat16 (BF16) mixed precision while maintaining FP32 master weights for accuracy. To optimize training efficiency, we employ Flash Attention (Dao et al., 2022; Dao, 2024) and regional compilation techniques. Additionally, distributed checkpointing is implemented to minimize the overhead of saving model state.

4.2 Training Procedure

Ovis-Image is trained using a four-stage pipeline: a pretraining stage followed by three post-training stages. Each subsequent stage is initialized using the checkpoint from the previous stage.

Stage 0: Pretraining. The MMDiT is initialized randomly and optimized throughout all four training stages, while the text encoder and VAE use pretrained weights and remain frozen during training. The training objective follows the standard noise-prediction loss commonly applied in flow-matching-style

diffusion models (Esser et al., 2024). AdamW serves as the optimizer, paired with a constant learning rate schedule and a brief linear warmup period. Initially, the model is trained on 256×256 images, followed by training on images of varying resolutions and aspect ratios, ranging from 512 to 1024 pixels and 0.25 to 4.0, respectively. This process produces a robust initial model, which is further refined in the later stages using supervised data and preference optimization.

Stage 1: Supervised Fine-tuning. In the second stage, we move from generic caption data to instruction-style supervision tailored to common text-to-image usage. Starting from the pretraining checkpoint, we finetune the MMDiT on a mixture of open and proprietary datasets. This stage therefore teaches the model not only what to draw, but also how to interpret instruction-like descriptions, constraints, and text-rendering requirements.

The training objective remains the same noise-prediction loss as in pretraining, applied to latent representations of images up to 1024 resolution with different aspect ratios, so that the model learns to handle variable input sizes and aspect ratios at inference time. We use a small learning rate and a shorter schedule, which helps preserve the general visual competence learned during pretraining while adapting to the instruction-style and text-rendering distributions. In practice, we find that this supervised tuning substantially improves faithfulness to user prompts and aesthetic quality of the generated images.

Stage 2: DPO. In the second stage, we apply Direct Preference Optimization (DPO) (Wallace et al., 2024) directly to the diffusion model using a mixture of human and model generated preference data. Each training example consists of a prompt c and two images (x^w, x^ℓ), where x^w is labeled as preferred (winner) and x^ℓ as dispreferred (loser). We keep a frozen reference model p_{ref} at the end of the supervised stage and treat the current image decoder p_θ initialized from p_{ref} as the policy model, which is trained to assign higher probability to the denoising trajectory leading to the preferred sample.

For each pair, we compute a DPO-style log-likelihood ratio

$$\Delta \log p_\theta(x^w, x^\ell \mid c) = \big[\log p_\theta(x^w \mid c) - \log p_\theta(x^\ell \mid c)\big] - \big[\log p_{\text{ref}}(x^w \mid c) - \log p_{\text{ref}}(x^\ell \mid c)\big], \tag{1}$$

and minimize the standard Diffusion-DPO objective

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(c,x^w,x^\ell)}\Big[\log \sigma\big(\beta\, \Delta \log p_\theta(x^w, x^\ell \mid c)\big)\Big], \tag{2}$$

where $\sigma(\cdot)$ is the logistic function and β is a temperature hyperparameter. In practice, the log probabilities are instantiated via the diffusion reconstruction losses along the denoising trajectories, following Wallace et al. (2024).

Following Diffusion-SDPO (Fu et al., 2025), we further incorporate a winner-preserving safeguard that modifies how the winner and loser branches contribute to the update. Let $\mathcal{L}^w(\theta)$ and $\mathcal{L}^\ell(\theta)$ denote the per-pair diffusion reconstruction losses for the winner and loser, and let $g^w = \nabla_o \mathcal{L}^w$ and $g^\ell = \nabla_o \mathcal{L}^\ell$ be their output-space gradients. Diffusion-SDPO explicitly computes a gradient scale factor λ_{safe} to stabilize the optimization as follows:

$$\lambda_{\text{safe}} = \text{clip}\left(\mu\, \frac{\langle g^w, g^\ell \rangle}{\|g^\ell\|^2 + \varepsilon}, \lambda_{\min}, \lambda_{\max}\right), \tag{3}$$

where μ controls the overall strength of the loser branch, ε is a small constant for numerical stability, and the clipping interval $[\lambda_{\min}, \lambda_{\max}]$ is chosen so that the first-order change of the winner loss is non-positive. Intuitively, the loser gradient is down-weighted whenever it conflicts with the winner gradient, which implicitly clips overly aggressive loser updates and preserves the quality of the preferred branch. Omitting this safeguard leads to models that frequently produce noisy or artifact-prone images, whereas SDPO-style control yields more stable optimization and systematically better visual quality.

We retain the same learning rate as used in the SFT stage, but adopt a large global batch size and β in order to obtain stable preference gradients and avoid drifting far from the supervised baseline. The DPO stage enhances the model by improving its helpfulness, harmlessness, adherence to prompts (including layout and text rendering), and by reducing noticeable artifacts.

Stage 3: GRPO. After training with DPO, we refine the model using Group Relative Policy Optimization (GRPO) (Shao et al., 2024; Liu et al., 2025), conducting on-policy sampling during training and evaluating with a set of reward models. For each prompt, the model generates multiple candidate images as a group, which are then scored by a combination of reward models. Conditioned on a text prompt c, the flow model predicts a group of G individual images $\{x_0^i\}_{i=1}^G$ with their corresponding trajectorys $\{x_T^i, x_{T-1}^i, ..., x_0^i\}_{i=1}^G$. The advantage of i-th image within the group can be formulated as:

$$A_i = \frac{R(x_0^i, c) - mean(\{R(x_0^i, c)\}_{i=1}^G)}{std(\{R(x_0^i, c)\}_{i=1}^G)}, \tag{4}$$

where R denotes the reward model. Consequently, the training objective of GRPO is:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{c\sim\mathcal{D},\{x_T^i,\ldots,x_0^i\}_{i=1}^{G}\sim\pi_\theta} f(r, A, \theta, \epsilon, \beta) \tag{5}$$

where

$$f(r, A, \theta, \epsilon, \beta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{T}\sum_{t=0}^{T-1}\left(\min(r_t^i(\theta)A_i, \text{clip}(r_t^i(\theta), 1-\epsilon, 1+\epsilon)A_i) - \beta D_{KL}(\pi_\theta||\pi_{\text{ref}})\right)$$

$$r_t^i(\theta) = \frac{p_\theta(x_{t-1}^i|x_t^i, c)}{p_{\theta_{\text{old}}}(x_{t-1}^i|x_t^i, c)}.$$

To accelerate training with minimizing the impact on performance, we sample each candidate image using fewer denoising steps. Furthermore, we introduce coefficients-preserving sampling (Wang & Yu, 2025) during the GRPO stage to further enhance performance. The training window adaptively learns the needs of different denoise stages. We retain the same learning rate from the DPO stage and run GRPO for approximately 500 steps. Throughout this process, the policy is optimized to maximize the expected reward, while applying a KL penalty to constrain its divergence from the DPO model.

5 Evaluation

We evaluate Ovis-Image on the text-to-image task from two perspectives: text rendering capability and general text-to-image generation capability. To evaluate the text rendering capability, we conduct evaluations on LongText-Bench (Geng et al., 2025)and CVTG-2K (Du et al., 2025). To evaluate the general text-to-image generation capability, we present a comprehensive evaluation across multiple public benchmarks, including DPG-Bench (Hu et al., 2024), GenEval (Ghosh et al., 2023) and OneIG-Bench (Chang et al., 2025). Despite its compact parameter size, Ovis-Image consistently outperforms significantly larger open-source baselines (highlighted in gray), proving that it delivers competitive generation quality with superior parameter efficiency.

CVTG-2K Table 2 presents the results of the English rendering evaluation on CVTG-2K (Du et al., 2025). This benchmark comprises 2,000 prompts, each demanding the rendering of 2 to 5 English text regions on the generated image. It introduces Word Accuracy, NED and CLIPScore to evaluate the precision of the text rendering. As shown in the table, Ovis-Image achieves the highest overall word accuracy across all regions. Additionally, Ovis-Image obtains the highest NED and CLIPScore, further confirming its superior text rendering capability.

LongText-Bench Table 3 presents the evaluation results on LongText-Bench (Geng et al., 2025), a benchmark designed to examine the model's capability to accurately render long texts in both English and Chinese. As illustrated in the table, Ovis-Image demonstrates superior performance on the Chinese text. Despite its relatively small model parameter, Ovis-Image still excels at generating long English text, achieving performance comparable against closed-source models and models with larger parameter counts. This result highlights the particular strength of Ovis-Image in long text generation capabilities.

Table 2: Evaluation of text rendering ability on CVTG-2K.

Model	#Params.	Word Accuracy↑					NED↑	CLIPScore↑
		2 regions	3 regions	4 regions	5 regions	average		
Seedream 3.0 (Gao et al., 2025)	-	0.6282	0.5962	0.6043	0.5610	0.5924	0.8537	0.7821
GPT4o (OpenAI, 2025)	-	0.8779	0.8659	0.8731	0.8218	0.8569	0.9478	0.7982
SD3.5 Large (Esser et al., 2024)	11B+8B	0.7293	0.6825	0.6574	0.5940	0.6548	0.8470	0.7797
RAG-Diffusion (Chen et al., 2024)	11B+12B	0.4388	0.3316	0.2116	0.1910	0.2648	0.4498	0.7797
FLUX.1-dev (Labs, 2024)	11B+12B	0.6089	0.5531	0.4661	0.4316	0.4965	0.6879	0.7401
TextCrafter (Du et al., 2025)	11B+12B	0.7628	0.7628	0.7406	0.6977	0.7370	0.8679	0.7868
Qwen-Image (Wu et al., 2025a)	7B+20B	0.8370	0.8364	0.8313	0.8158	0.8288	0.9116	0.8017
Ovis-Image	2B+7B	**0.9248**	**0.9239**	**0.9180**	**0.9166**	**0.9200**	**0.9695**	**0.8368**

DPG-Bench Table 4 reports the results on DPG-Bench (Hu et al., 2024), a benchmark of 1,000 dense prompts intended to evaluate the alignment of text-to-image generation in various dimensions, allowing a detailed inspection of prompt adherence from multiple perspectives. Overall, Ovis-Image delivers robust performance compared to both close-source models and open-source models with larger parameter counts.

Table 3: Evaluation of text rendering ability on LongText-Bench.

Model	#Params.	LongText-Bench-EN	LongText-Bench-ZN
Kolors 2.0 (Team, 2025a)	-	0.258	0.329
GPT4o (OpenAI, 2025)	-	**0.956**	0.619
Seedream 3.0 (Gao et al., 2025)	-	0.896	0.878
OmniGen2 (Wu et al., 2025b)	3B+4B	0.561	0.059
Janus-Pro (Chen et al., 2025b)	7B	0.019	0.006
BLIP3-o (Chen et al., 2025a)	7B+1B	0.021	0.018
FLUX.1-dev (Labs, 2024)	11B+12B	0.607	0.005
BAGEL (Deng et al., 2025)	7B+7B	0.373	0.310
HiDream-I1-Full (Cai et al., 2025)	11B+17B	0.543	0.024
Qwen-Image (Wu et al., 2025a)	7B+20B	0.943	0.946
Ovis-Image	2B+7B	0.922	**0.964**

GenEval Table 5 summarizes the performance on GenEval (Ghosh et al., 2023) benchmark, which emphasizes object-centric text-to-image generation by employing compositional prompts with a wide range of object attributes. These results exhibit the competitive controllable generation capabilities of Ovis-Image.

OneIG-Bench Table 6 and Table 7 show the performance comparison on OneIG-Bench (Chang et al., 2025), a comprehensive benchmark developed for detailed evaluation of T2I models across multiple dimensions. As shown in the table, Ovis-Image demonstrates exceptional bilingual performance, particularly distinguished by its performance in the text dimensions.

Table 4: Evaluation of text-to-image generation ability on DPG-Bench.

Model	#Params.	Global	Entity	Attribute	Relation	Other	Overall
Seedream 3.0 (Gao et al., 2025)	-	**94.31**	**92.65**	91.36	92.78	88.24	88.27
GPT4o (OpenAI, 2025)	-	88.89	88.94	89.84	92.63	90.96	85.15
Ovis-U1 (Wang et al., 2025)	2B+1B	82.37	90.08	88.68	93.35	85.20	83.72
OmniGen2 (Wu et al., 2025b)	3B+4B	88.81	88.83	90.18	89.37	90.27	83.57
Janus-Pro (Chen et al., 2025b)	7B	86.90	88.90	89.40	89.32	89.48	84.19
BAGEL (Deng et al., 2025)	7B+7B	88.94	90.37	91.29	90.82	88.67	85.07
HiDream-I1-Full (Cai et al., 2025)	11B+17B	76.44	90.22	89.48	93.74	91.83	85.89
UniWorld-V1 (Lin et al., 2025)	7B+12B	83.64	88.39	88.44	89.27	87.22	81.38
Qwen-Image (Wu et al., 2025a)	7B+20B	91.32	91.56	**92.02**	**94.31**	**92.73**	**88.32**
Ovis-Image	2B+7B	82.37	92.38	90.42	93.98	91.20	86.59

Table 5: Evaluation of text-to-image generation ability on GenEval.

Model	#Params.	Single object	Two object	Counting	Colors	Position	Attribute binding	Overall
Seedream 3.0 (Gao et al., 2025)	-	0.99	0.96	**0.91**	**0.93**	0.47	**0.80**	0.84
GPT4o (OpenAI, 2025)	-	0.99	0.92	0.85	0.92	0.75	0.61	0.84
Ovis-U1 (Wang et al., 2025)	2B+1B	0.98	**0.98**	0.90	0.92	**0.79**	0.75	**0.89**
OmniGen2 (Wu et al., 2025b)	3B+4B	**1.00**	0.95	0.64	0.88	0.55	0.76	0.80
Janus-Pro (Chen et al., 2025b)	7B	0.99	0.89	0.59	0.90	**0.79**	0.66	0.80
BAGEL (Deng et al., 2025)	7B+7B	0.99	0.94	0.81	0.88	0.64	0.63	0.82
HiDream-I1-Full (Cai et al., 2025)	11B+17B	1.00	**0.98**	0.79	0.91	0.60	0.72	0.83
UniWorld-V1 (Lin et al., 2025)	7B+12B	0.99	0.93	0.79	0.89	0.49	0.70	0.80
Qwen-Image (Wu et al., 2025a)	7B+20B	0.99	0.92	0.89	0.88	0.76	0.77	0.87
Ovis-Image	2B+7B	**1.00**	0.97	0.76	0.86	0.67	**0.80**	0.84

Computational Overhead Table 8 presents a comparative analysis of computational overhead, focusing on inference time and GPU memory utilization. Ovis-Image exhibits a superior trade-off between resource efficiency and model performance. Most notably, Ovis-Image maintains a significantly lower memory footprint compared to larger baselines. Furthermore, in terms of temporal efficiency, Ovis-Image achieves a substantial speedup,thereby offering a more practical solution for resource-constrained environments.

Table 6: Evaluation of text-to-image generation ability on OneIG-EN.

Model	#Params.	Alignment	Text	Reasoning	Style	Diversity	Overall
Kolors 2.0 (Team, 2025a)	-	0.820	0.427	0.262	0.360	0.300	0.434
Imagen4 (Google, 2025)	-	0.857	0.805	0.338	0.377	0.199	0.515
Seedream 3.0 (Gao et al., 2025)	-	0.818	0.865	0.275	0.413	0.277	0.530
GPT4o (OpenAI, 2025)	-	0.851	0.857	**0.345**	**0.462**	0.151	0.533
Ovis-U1 (Wang et al., 2025)	2B+1B	0.816	0.034	0.226	0.443	0.191	0.342
CogView4 (Z.ai., 2025)	6B	0.786	0.641	0.246	0.353	0.205	0.446
Janus-Pro (Chen et al., 2025b)	7B	0.553	0.001	0.139	0.276	**0.365**	0.267
OmniGen2 (Wu et al., 2025b)	3B+4B	0.804	0.680	0.271	0.377	0.242	0.475
BLIP3-o (Chen et al., 2025a)	7B+1B	0.711	0.013	0.223	0.361	0.229	0.307
FLUX.1-dev (Labs, 2024)	11B+12B	0.786	0.523	0.253	0.368	0.238	0.434
BAGEL (Deng et al., 2025)	7B+7B	0.769	0.244	0.173	0.367	0.251	0.361
BAGEL+CoT (Deng et al., 2025)	7B+7B	0.793	0.020	0.206	0.390	0.209	0.324
HiDream-I1-Full (Cai et al., 2025)	11B+17B	0.829	0.707	0.317	0.347	0.186	0.477
HunyuanImage-2.1 (Team, 2025b)	7B+17B	0.835	0.816	0.299	0.355	0.127	0.486
Qwen-Image (Wu et al., 2025a)	7B+20B	**0.882**	0.891	0.306	0.418	0.197	**0.539**
Ovis-Image	2B+7B	0.858	**0.914**	0.308	0.386	0.186	0.530

Table 7: Evaluation of text-to-image generation ability on OneIG-ZN.

Model	#Params.	Alignment	Text	Reasoning	Style	Diversity	Overall
Kolors 2.0 (Team, 2025a)	-	0.738	0.502	0.226	0.331	0.333	0.426
Seedream 3.0 (Gao et al., 2025)	-	0.793	0.928	0.281	0.397	0.243	0.528
GPT4o (OpenAI, 2025)	-	0.812	0.650	**0.300**	**0.449**	0.159	0.474
CogView4 (Z.ai., 2025)	6B	0.700	0.193	0.236	0.348	0.214	0.338
Janus-Pro (Chen et al., 2025b)	7B	0.324	0.148	0.104	0.264	**0.358**	0.240
BLIP3-o (Chen et al., 2025a)	7B+1B	0.608	0.092	0.213	0.369	0.233	0.303
BAGEL (Deng et al., 2025)	7B+7B	0.672	0.365	0.186	0.357	0.268	0.370
BAGEL+CoT (Deng et al., 2025)	7B+7B	0.719	0.127	0.219	0.385	0.197	0.329
HiDream-I1-Full (Cai et al., 2025)	11B+17B	0.620	0.205	0.256	0.304	0.300	0.337
HunyuanImage-2.1 (Team, 2025b)	7B+17B	0.775	0.896	0.271	0.348	0.114	0.481
Qwen-Image (Wu et al., 2025a)	7B+20B	**0.825**	**0.963**	0.267	0.405	0.279	**0.548**
Ovis-Image	2B+7B	0.805	0.961	0.273	0.368	0.198	0.521

Table 8: Comparison of model inference time and GPU memory usage (1024×1024 images, 50-step sampling, BF16 inference)

Model	#Params.	Accelerate	A100		H100	
			Memory (MB)	Time (s)	Memory (MB)	Time (s)
Flux.1-dev	11B+12B	guidance dis.	34637	23.51	34661	11.03
Qwen-Image	7B+20B	-	59329	45.16	59354	20.27
Ovis-U1	2B+1B	-	10528	8.41	11937	4.29
Ovis-Image	2B+7B	-	24959	30.56	24276	13.74

6 Conclusion

We presented Ovis-Image, a 7B text-to-image model designed to reconcile strong in-image text rendering with practical deployment cost. By pairing a diffusion-based visual decoder with the Ovis 2.5 multimodal backbone and training it through a text-centric pipeline, Ovis-Image attains text rendering quality comparable to much larger open models and approaches leading closed-source systems, while preserving solid general-purpose generation and fitting on a single high-end GPU. Beyond the empirical gains, Ovis-Image illustrates a more general design principle: frontier-like text-aware generation can emerge from moderate-scale models when architectural choices, data curation, and alignment objectives are explicitly organized around the demands of in-image text, rather than treated as a byproduct of generic image synthesis.

7 Contributors

Guo-Hua Wang[1], Liangfu Cao, Tianyu Cui, Minghao Fu[2], Xiaohao Chen, Pengxin Zhan, Jianshan Zhao, Lan Li[2], Bowen Fu[2], Jiaqi Liu, Qing-Guo Chen

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Qi Cai, Jingwen Chen, Yang Chen, Yehao Li, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Yiheng Zhang, Fengbin Gao, Peihan Xu, et al. HiDream-I1: A high-efficient image generative foundation model with sparse diffusion transformer. *arXiv preprint arXiv:2505.22705*, 2025.

Jingjing Chang, Yixiao Fang, Peng Xing, Shuhan Wu, Wei Cheng, Rui Wang, Xianfang Zeng, Gang Yu, and Hai-Bao Chen. OneIG-Bench: Omni-dimensional nuanced evaluation for image generation. *arXiv preprint arXiv:2506.07977*, 2025.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. BLIP3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025a.

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. *arXiv preprint arXiv:2501.17811*, 2025b.

Zhennan Chen, Yajie Li, Haofan Wang, Zhibo Chen, Zhengkai Jiang, Jun Li, Qian Wang, Jian Yang, and Ying Tai. Region-aware text-to-image generation via hard binding and soft refinement. *arXiv preprint arXiv:2411.06558*, 2024.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pp. 933–941. PMLR, 2017.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Nikai Du, Zhennan Chen, Shan Gao, Zhizhou Chen, Xi Chen, Zhengkai Jiang, Jian Yang, and Ying Tai. Textcrafter: Accurately rendering multiple texts in complex visual scenes. *arXiv preprint arXiv:2503.23461*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Minghao Fu, Guo-Hua Wang, Tianyu Cui, Qing-Guo Chen, Zhao Xu, Weihua Luo, and Kaifu Zhang. Diffusion-sdpo: Safeguarded direct preference optimization for diffusion models. *arXiv preprint arXiv:2511.03317*, 2025.

Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, et al. Seedream 3.0 technical report. *arXiv preprint arXiv:2504.11346*, 2025.

Zigang Geng, Yibing Wang, Yeyao Ma, Chen Li, Yongming Rao, Shuyang Gu, Zhao Zhong, Qinglin Lu, Han Hu, Xiaosong Zhang, et al. X-omni: Reinforcement learning makes discrete autoregressive image generative models great again. *arXiv preprint arXiv:2507.22058*, 2025.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GENEVAL: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems*, 2023.

[1]Correspondence to Guo-Hua Wang <wangguohua@alibaba-inc.com>
[2]Work done during the internship at Alibaba Group

Google. Imagen. https://deepmind.google/models/imagen/, 2025.

Google DeepMind. Gemini 3 Pro Image: Model card. https://deepmind.google/models/model-cards/gemini-3-pro-image/, 2025. Model card, accessed Nov. 25, 2025.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. EllA: Equip diffusion models with LLM for enhanced semantic alignment. *arXiv:2403.05135*, 2024.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, pp. 36652 – 36663, 2023.

Vijay Anand Korthikanti, Jared Casper, Sangkug Lym, Lawrence McAfee, Michael Andersch, Mohammad Shoeybi, and Bryan Catanzaro. Reducing activation recomputation in large transformer models. *Proceedings of Machine Learning and Systems*, 5:341–353, 2023.

Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024.

Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. UniWorld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025.

Shiyin Lu, Yang Li, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Han-Jia Ye. Ovis: Structural embedding alignment for multimodal large language model. *arXiv:2405.20797*, 2024.

Shiyin Lu, Yang Li, Yu Xia, Yuwei Hu, Shanshan Zhao, Yanqing Ma, Zhichao Wei, Yinglun Li, Lunhao Duan, Jianshan Zhao, et al. Ovis2.5 technical report. *arXiv preprint arXiv:2508.11737*, 2025.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15086–15095, 2025.

OpenAI. Introducing 4o image generation. https://openai.com/index/introducing-4o-image-generation/, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, et al. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Kuaishou Kolors Team. Kolors 2.0. https://app.klingai.com/cn/, 2025a.

Tencent Hunyuan Team. HunyuanImage 2.1: An efficient diffusion model for high-resolution (2k) text-to-image generation. https://github.com/Tencent-Hunyuan/HunyuanImage-2.1, 2025b.

Daniel Verdú and Javier Martín. Flux.1 Lite: Distilling flux.1-dev for efficient text-to-image generation, 2024. Freepik Research. Contact: dverdu@freepik.com, javier.martin@freepik.com.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Feng Wang and Zihao Yu. Coefficients-preserving sampling for reinforcement learning with flow matching. *arXiv preprint arXiv:2509.05952*, 2025.

Guo-Hua Wang, Shanshan Zhao, Xinjie Zhang, Liangfu Cao, Pengxin Zhan, Lunhao Duan, Shiyin Lu, Minghao Fu, Xiaohao Chen, Jianshan Zhao, et al. Ovis-U1 technical report. *arXiv preprint arXiv:2506.23044*, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-Image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. OmniGen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

THUKEG Z.ai. Cogview4. https://github.com/THUDM/CogView4, 2025.