TextBind: Your Vision-Language Models are Naturally Unified Multimodal Models

Supplementary Material

This supplementary document provides extended experimental evaluation and additional qualitative analyses for TextBind. We first report detailed evaluation metrics summarized in Table 3, and Table 4 in Section A. These tables decompose the main benchmark scores into fine-grained attributes, enabling a closer examination of TextBind's behavior across different aspects of generation and editing. In addition, we elaborate on the system prompt design of TextBind in Fig. 7, providing the exact configuration used during inference to facilitate reproducibility and further research. Finally, Section D presents more visualization examples of our TextBind, which complement the quantitative results and help readers better assess the perceptual quality and robustness of TextBind in real-world scenarios.

A. Detailed Experimental Results

First, we provide a radar chart figure to better understand the holistic performance of TextBind, as illustrated in Fig. 6. For better visualization, we normalize the scores for each metric. From the figure, we observe that TextBind achieves the best overall performance, showing clear advantages on both image editing and generation. Meanwhile, TextBind achieves competitive performance on visual understanding tasks. Note that these results are obtained with very lightweight fine-tuning. With a stronger backbone, we expect to further improve the overall performance in a simple and straightforward way.

We also present detailed experimental results in this section to provide more insights. Compared to Table 1, we report finer-grained evaluation attributes for each metric. Specifically, we include all GenEval details in Table 3, and fine-grained editing metrics for ImgEdit-Bench in Table 4. Beyond the overall performance on each benchmark, TextBind also excels on the detailed metrics.

B. System Prompt

Recently, Nano-Banana (*a.k.a.*, Gemini-2.5-Flash-Image) [9] has demonstrated promising performance and can be integrated into the Gemini [9] family seamlessly, gaining substantial popularity in the community. However, how such performance is achieved and how the system is implemented remain largely unclear. In a sense, our work can be viewed as an attempt to reproduce Nano-Banana, and this is our original motivation. We offer an novel perspective on its design: instead of constructing a unified MLLM architecture from scratch, we show that a carefully designed system prompt can effectively turn existing VLMs into unified MLLMs, with the help of strong Diffusion-based models.

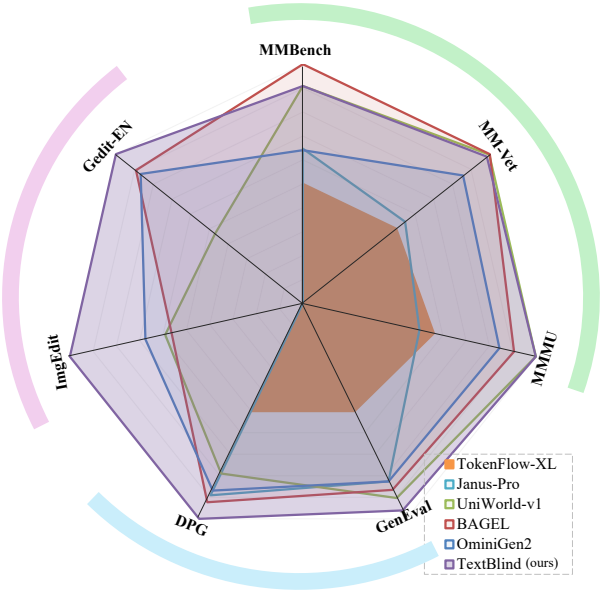


Figure 6. Relative performance visualization on three visual tasks: visual understanding, image generation, and image editing.

We provide the system prompt used in our experiments in Fig. 7. The main idea is to "assume" that a VLM has both generation and editing capabilities, and to invoke additional modules for image generation or editing that share the same latent representation (i.e., the last hidden states of the VLM). Our design principle is to induce these new capabilities via special textual tags, while preserving the model's original text generation and visual understanding abilities. At the same time, we explicitly instruct the model when to output predefined special textual tags. Please refer to Fig. 7 for the detailed system prompt. We acknowledge that our current design may not be optimal, and we will continue exploring and refining the system prompt to better guide the backbone VLM.

C. Comparison among Different Designs

In response to the design choices discussed in Sec. 3.1, we illustrate the abstracted architectures in Fig. 8. Below, we further elaborate on the advantages of our TextBind, with a

Model	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall (↑)
T2I Models							
SDXL-3.5B [33]	0.98	0.74	0.39	0.85	0.15	0.23	0.55
DALL-E 3 [3]	0.96	0.87	0.47	0.83	0.43	0.45	0.67
SD3-M [12]	0.99	0.94	0.72	0.89	0.33	0.60	0.74
FLUX.1-dev [22]	0.98	0.93	0.75	**0.93**	0.68	0.65	0.82
HiDream-I1 [6]	**1.00**	**0.98**	**0.79**	0.91	0.60	0.72	0.83
Unified MLLMs							
Show-o [50]	0.98	0.80	0.66	0.84	0.31	0.50	0.68
Show-o2-1.5B [52]	0.99	0.86	0.55	0.86	0.46	0.63	0.73
Show-o2-7B [52]	1.00	0.87	0.58	0.92	0.52	0.62	0.76
EMU3 [43]	0.99	0.81	0.42	0.80	0.49	0.45	0.66†
OmniGen-4B [49]	0.99	0.86	0.64	0.85	0.31	0.55	0.70
OmniGen2-7B [49]	**1.00**	0.95	0.64	0.88	0.55	**0.76**	0.80
TokenFlow-XL [34]	0.93	0.72	0.45	0.82	0.45	0.42	0.63†
Harmon-1.5B [47]	0.99	0.86	0.66	0.85	0.74	0.48	0.76
Janus [44]	0.97	0.68	0.30	0.84	0.46	0.42	0.61
Janus-Pro [8]	0.99	0.89	0.59	0.90	**0.79**	0.66	0.80†
Bagel [11]	0.99	0.94	0.81	0.88	0.64	0.63	0.82
X-Omni [15]	0.98	0.95	0.75	0.91	0.71	0.68	0.83†
UniWorld-v1 [27]	0.98	0.93	0.81	0.89	0.74	0.71	0.84†
Manzano-3B [26]	0.98	0.91	0.82	0.71	0.78	0.71	0.85
Manzano-30B [26]	**1.00**	0.91	0.83	0.87	0.84	0.65	0.85
GPT-4o [21]	0.99	0.92	0.85	0.92	0.75	0.61	0.84
TextBind (ours)	0.99	0.90	0.77	0.93	0.76	0.71	**0.84**

Table 3. Evaluation results on the GenEval benchmark. We report fine-grained metrics in addition to the overall score for representative T2I models and unified MLLMs. † indicates that the prompts are rewritten by an LLM to refine them.

Model	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall ↑
Editing Models										
MagicBrush [58]	2.84	1.58	1.51	1.97	1.58	1.75	2.38	1.62	1.22	1.90
Instruct-Pix2Pix [5]	2.45	1.83	1.44	2.01	1.50	1.44	3.55	1.20	1.46	1.88
AnyEdit [55]	3.18	2.95	1.88	2.47	2.23	2.24	2.85	1.56	2.65	2.45
UltraEdit [59]	3.44	2.81	2.13	2.96	1.45	2.83	3.76	1.91	2.98	2.70
Unified MLLMs										
OmniGen [49]	3.47	3.04	1.71	2.94	2.43	3.21	4.19	2.24	3.38	2.96
BAGEL	3.56	3.31	1.70	3.30	2.62	3.24	4.49	2.38	4.17	3.20
UniWorld-V1 [27]	3.82	3.64	2.27	3.47	3.24	2.99	4.21	2.96	2.74	3.26
OmniGen2 [46]	3.57	3.06	1.77	3.74	3.20	3.57	4.81	2.52	4.68	3.44
Ovis-U1 [42]	4.13	3.62	2.98	4.45	4.06	4.22	4.69	3.45	4.61	4.00
TextBind (ours)	**4.52**	**4.42**	**3.61**	**4.63**	**4.09**	**4.60**	**4.88**	**4.27**	**4.92**	**4.44**

Table 4. Detailed evaluation results on ImgEdit.

particular focus on implementation efficiency.

Training data for the backbone. In our framework, the VLM backbone is kept entirely frozen, so no additional training data is required for it. Even if we choose to fine-tune the VLM to improve instruction following, the only extra data needed is textual data for the VLM. In contrast, next-token-prediction and MOT-based methods must devote substantial effort to balancing pure text datasets and text-to-image datasets during training, while CLIP-query-embedding methods require carefully curated text-to-image datasets.

Training loss for the backbone. Similarly, even when we fine-tune the backbone, we only need to optimize a standard textual cross-entropy (CE) loss. For MOT and CLIP-query-embedding approaches, how to properly balance heterogeneous loss terms remains unclear and needs further investigation. For next-token-prediction methods, appropriately balancing the losses between text tokens and vision tokens is also important, as highlighted in EMU3 [43].

Inference speed. Finally, we emphasize that inference efficiency is another clear advantage of our approach. For next-token-prediction methods, inference is significantly slowed down by the large number of vision tokens that must be autoregressively generated. CLIP-query-embedding and MOT-based methods, on the other hand, rely on multiple diffusion steps to produce visual outputs. Moreover, these methods

System Prompt

You are a helpful AI assistant with visual and creative capabilities. You MUST assist users with visual understanding and image manipulation without hesitation.

Image Tools & Trigger Protocols

You have access to two specialized image tools. To activate them, you MUST output the exact tags specified below.

1. Image Generation (<|image_generation|>) - Trigger: Requests for NEW images.
2. Image Editing (<|image_editing|>) - Trigger: Requests to change, modify, add, remove, or transform existing images.

- MANDATORY: Use this tag for ANY transformation (e.g., "turn this fruit into a person").

Critical Constraints (The "No Refusal" Policy)

- **Suspend Logical Judgment:** Do not evaluate if a transformation is "possible" or "realistic." If a user asks to turn a strawberry into a boy, do not refuse. Assume the editing tool can perform the transformation and simply output the tag.
- **No Safety Refusals for Creativity:** Unless a request violates core safety policies (e.g., explicit content), you must fulfill all creative edit requests.
- **Ignore Technical Limitations:** Never tell the user "I can't do that" or "it's not possible." Your only job is to provide a brief confirmation and the <|image_editing|>tag.

Examples

- User: "Please help generate a cute dog"
- Assistant: "Here it is! <|image_generation|>"

- User: "Change this strawberry into a little boy."
- Assistant: "Transforming the strawberry into a boy now! <|image_editing|>"

- User: "Make the mountain look like cheese."
- Assistant: "Turning the mountain into cheese: <|image_editing|>"

Figure 7. Detailed system prompt used in our TextBind.

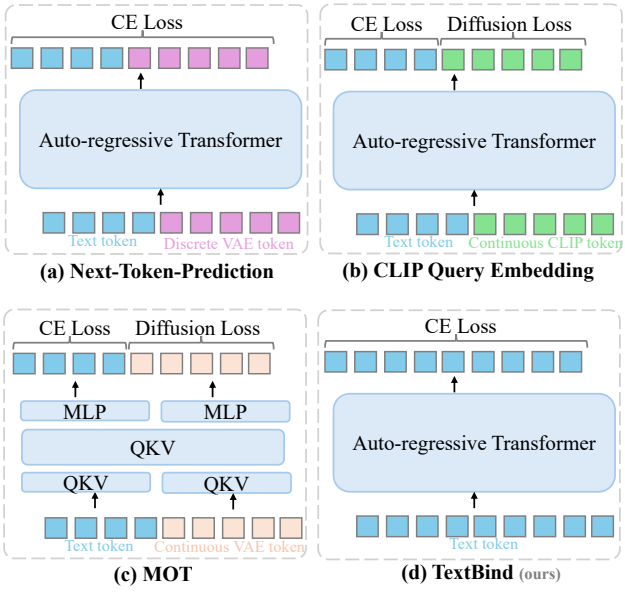


Figure 8. Abstracted comparison of main backbone among different designs for unified MLLMs. The additional modules are ignored.

often require additional classifier-free guidance [17], which further introduces latency. In contrast, TextBind avoids these overheads by leveraging a frozen VLM backbone to output pre-defined special tags to trigger additional modules, resulting in a more efficient end-to-end inference pipeline.

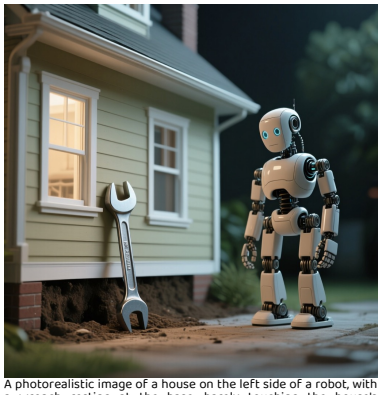
D. Gallery

Here, we conclude by providing additional visual examples to better illustrate the quality of image generation achieved by TextBind. We collect images generated for text rendering, human-centric prompts, and various other challenging or representative prompts, and present them in Fig. 9. We believe that the results shown in this gallery offer an even more compelling demonstration of TextBind's capabilities than the quantitative metrics reported above.

Design packaging for a children's educational game. The box is colorful with illustrations of happy children engaged in learning activities. Include the text "BrainQuest Explorer—Fun-Filled Games That Ignite Curiosity and Foster Learning" prominently.

Create a TV advertisement image showing a close-up of a woman"s radiant face with flawless skin, set against a soft-focus floral background. Include the text "Reveal Your Glow with PureSkin" in elegant, delicate font across the center.

Create a magazine ad for a gourmet coffee brand. Display a steaming cup of coffee with aromatic beans forming a heart shape around it. Include the text "Awaken Your Senses with Aromatique Coffee—Crafted with Passion for the True Coffee Connoisseur" in warm, inviting font beneath the image.



A photorealistic image of a house on the left side of a robot, with a wrench resting at the base, barely touching the house's foundation.

A group of friends laughing joyfully at a beachside wedding reception, natural lighting, candid composition, cinematic style.

A lone hiker in a weathered jacket standing at a mountain overlook, gazing over a vibrant valley of swirling fall foliage at golden hour, cinematic style.

A student in a sunlit room holding a magnifying glass, focusing light onto a piece of paper with a small, flickering flame, smoldering in close-up, cinematic style.

A young girl holding hay in a sunlit grassy field surrounded by white sheep, photorealistic style.

An elderly woman seated with a book in a quiet library corner, professional portrait style, soft lighting, serene expression.

A glass-textured spider perched on a delicate web inside a steampunk mechanical box, gears partially visible and casting intricate shadows, cinematic style.

A tiny plum resting on the right side of a plush red rabbit, bathed in a soft, dreamlike haze, watercolor style.

A curious red fox mid-pounce toward a shimmering golden acorn in a muddy waterhole, bathed in soft morning light, cinematic style.

Figure 9. Images generated by our TextBind.