

GOAL: Grounded text-to-image Synthesis with Joint Layout Alignment Tuning

Anonymous Authors

1 DATASET CONSTRUCTION

To assess the effectiveness of our alignment-based objectives, we construct a concise yet robust dataset named **GOAL2K** for fine-tuning our model. In this section, we provide a detailed description of the dataset construction process, which includes three processes: prompts collection, image generation, and layout annotation.

1.1 Prompts Collection

GOAL2K comprises a total of 2K high-quality and semantically accurate text-image pairs with layout annotations. Specifically, GOAL2K includes 1,000 template-based prompts and 1,000 natural prompts. To design template-based prompts, we initially selected 80 classes from the COCO [7] dataset and devised various templates covering different categories such as color assignment, spatial relationship, and object counting.

- For the color assignment task, we begin by utilizing the concept *A scene with a [color] [object]*, where the color is randomly selected from a predefined list (red, orange, yellow, green, blue, purple, pink, brown, black, white, and gray). Then, we generate complete prompts by using concept conjunctions to connect 1 to 5 objects with the word 'and'.
- For the spatial relationship task, we use the template prompt *A scene with [object name 1] on the [location1] and [object name2] on the [location2]*, where the location is chosen from left, right, top, and bottom.
- For the object counting task, we employ the template prompt *A scene with [number] [object name]*, where the number ranges from 1 to 5.

To ensure the fluency and grammatical correctness of the generated prompts, we employ GPT-4 [1] to generate prompts based on the provided templates and objects. This results in a total set of 1,000 template-based prompts, consisting of 400 prompts for color assignment, 300 for spatial relationships, and 300 for object counting. The instructions provided to GPT-4 [1] to generate template-based prompts for different categories are shown in Table 1.

For natural prompts, we directly select 1000 captions from the COCO [7] dataset containing 1 to 6 objects. Specifically, we include 200 prompts mentioning colors, 261 prompts containing spatial relations (e.g., top, right, left, side, next), and 239 prompts involving specific numbers. Additionally, we randomly add 300 extra prompts from COCO [7] dataset to enhance diversity. Table 2 summarizes our dataset statistics with examples for different categories.

1.2 Image Generation

To ensure the high quality and semantic accuracy of the images in GOAL2K, we utilize the state-of-the-art model DALL-E-3 [11] to generate images of size 1024×1024 for the training set, which are then resized to 512×512 as input for the model. The images generated by DALL-E-3 exhibit remarkable proficiency in following instructions and demonstrating creativity.

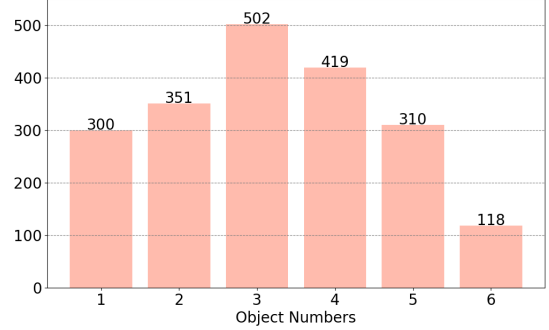


Figure 1: Distribution of object number of bounding boxes in an image for GOAL2K.

1.3 Layout Annotation

In our work, we utilize GroundingDINO [8] to generate layout annotations. Then, we manually verify the generated image-text pairs along with the layout annotations to ensure their fidelity to accurate semantic information. We show the distribution of the object number of bounding boxes in an image in Figure 1. The number of bounding boxes in an image ranges from 1 to 6, and the average number of bounding boxes per image in GOAL2K is 3.22. Figure 2 visualizes the data in GOAL2K. A single data of GOAL2K contains the prompt, the input image, and the layout conditions.

2 PROMPTS FOR LAYOUT PLANNING

In our work, given that manually annotating layouts are costly, we utilize GPT-4 [1] for layout planning during inference. The layout generated by GPT-4 consists of a bounding box for each foreground object, represented by coordinates in the (x, y, width, height) format, and a brief descriptive phrase corresponding to each bounding box. As shown in Table 3, the complete prompt utilized for layout planning consists of two components: basic instructions and in-context learning examples. The basic instructions indicate the role of the large language model (LLM) and the particular task we aim for the model to execute. Subsequently, following [2], we provide the LLM with several in-context learning examples to guide the model towards improved performance and proper formatting. In our work, we provide five examples (5-shot) for layout planning.

3 EXPERIMENTAL RESULTS

3.1 Effect of Distance Function

In our work, we employ discriminative semantic alignment (DSAlign) to ensure low-level semantic alignment by minimizing the spherical distance between the embeddings of the target region and the corresponding phrase. To investigate the effects of distance functions

Table 1: The instructions provided to GPT-4 to generate template-based prompts for different tasks.

Task	Instruction of GPT-4
Color assignment	You are an intelligent prompts generator. Your task is to generate prompts for the color assignment task based on the provided examples. Color in the generated prompts can be selected from a predefined list (red, orange, yellow, green, blue, purple, pink, brown, black, white, gray, silver, violet gold). Note that objects in the generated prompts should be classes of the COCO dataset. Please refer to examples below for the desired format.
	Templates: A scene with a [color1] [object 1].
	Examples:
	1. A scene with a pink stop sign.
	2. A scene with an orange tennis racket and a gray bear.
	3. A scene with a red vase and a yellow backpack and black elephant.
Spatial relationship	4. A scene with a red tie and an blue bed and a blue bowl and a red dog.
	5. A scene with a blue giraffe and a purple sheep and a white baseball bat and a green motorcycle and a white cup.
	You are an intelligent prompts generator. Your task is to generate prompts for the spatial relationship task based on the provided examples. Location in the generated prompts can be selected from a predefined list (left, right, top, and bottom). Note that objects in the generated prompts should be classes of the COCO dataset. Please refer to the example below for the desired format.
	Templates: A scene with [object name 1] on the [location1] and [object name2] on the [location2].
Object counting	Example: A scene with a cat on the left and a dog on the right.
	You are an intelligent prompts generator. Your task is to generate prompts for the counting task based on the provided examples. The number in the generated prompts ranges from 1 to 5. Note that objects in the generated prompts should be classes of the COCO dataset. Please refer to the example below for the desired format.
	Templates: A scene with [number] [object name 1].
	Example: A scene with two cats.

Table 2: Dataset statistics and examples of the GOAL2K.

Task	Type	Number	Example Prompt
Color assignment	Template-based	400	A scene with a purple cup and a red book.
	Natural	200	A red painted wall is against a television.
Spatial relationship	Template-based	300	A dog on the top and a chair on the bottom.
	Natural	261	A cat sitting on a counter top next to a stove below an oven mitt.
Object counting	Template-based	300	A scene with three chairs.
	Natural	239	A thick piece of pizza with two mugs of beer.
Others	Natural	300	Bench and window with shutters with bricked wall.
Total	-	2,000	-

Table 3: Full prompt to the GPT-4 for layout planning.

You are an intelligent bounding box generator. I will provide you with a caption for a photo, image, or painting. Your task is to generate the bounding boxes for the objects mentioned in the caption, along with a background prompt describing the scene. The images are of size 512×512 . The top-left corner has coordinates [0, 0]. The bottom-right corner has coordinates [512, 512]. The bounding boxes should not overlap or go beyond the image boundaries. Each bounding box should be in the format of (object name, [top-left x coordinate, top-left y coordinate, box width, box height]) and include exactly one object (i.e., start the object name with "a" or "an" if possible). If needed, you can make reasonable guesses. Please refer to the example below for the desired format.
Caption: A green bench and a blue bowl
Objects: [(‘a green bench’, [50, 284, 412, 82]), (‘a blue bowl’, [217, 244, 78, 40])]
Caption: A dog is curled up on a bed under a blanket.
Objects: [(‘a bed’, [59, 231, 394, 148]), (‘a dog’, [210, 281, 92, 98]), (‘a blanket’, [120, 281, 272, 98])]
Caption: there is a white disney bus that passed under the train tracks
Objects: [(‘a white Disney bus’, [100, 204, 312, 179]), (‘train tracks’, [0, 20, 512, 60])]
Caption: A cat is sitting on top of a computer chair which is covered in hair.
Objects: [(‘a cat’, [201, 100, 110, 160]), (‘a computer chair’, [30, 250, 452, 262])]
Caption: A black and white dog sitting on top of a bench.
Objects: [(‘a black and white dog’, [156, 204, 200, 150]), (‘a bench’, [51, 354, 410, 58])]
Caption:

Table 4: Effect of distance function for discriminative semantic alignment (DSAlign).

Distance Function	Color	Shape	Texture	Spatial
L2 Distance	50.03	51.08	57.06	35.36
Cosine Distance	50.01	50.33	55.09	35.99
Spherical Distance	53.55	51.19	58.37	37.28

used in DSAlign, we conduct experiments on the T2I-Compbench [5] using common distance metrics, including L2 distance and cosine distance. The results presented in Table 4 indicate that the spherical distance metric is more effective compared to other distance functions and is therefore adopted as the distance function for DSAlign.

4 MORE QUALITATIVE RESULTS

In this section, we present more qualitative results from the T2I-Compbench [5] across attributes such as color, spatial, shape, and texture. Figure 3 compares our proposed method with other layout-to-image methods, while Figure 4 compares it with text-to-image methods. The proposed method demonstrates outstanding performance in text-image alignment, generating images faithfully capturing the details of text prompts.

REFERENCES

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altmenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).
- [2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18392–18402.
- [3] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. 2023. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–10.
- [4] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. 2022. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032* (2022).
- [5] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. 2023. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems* 36 (2023), 78723–78747.
- [6] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. 2023. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22511–22521.
- [7] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*. Springer, 740–755.
- [8] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499* (2023).
- [9] Quynh Phung, Songwei Ge, and Jia-Bin Huang. 2023. Grounded text-to-image synthesis with attention refocusing. *arXiv preprint arXiv:2306.05427* (2023).
- [10] Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. 2023. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. In *Proceedings of the 31st ACM International Conference on Multimedia*. 643–654.
- [11] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International conference on machine learning*. Pmlr, 8821–8831.
- [12] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In

Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 10684–10695.

- [13] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. 2023. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7452–7461.

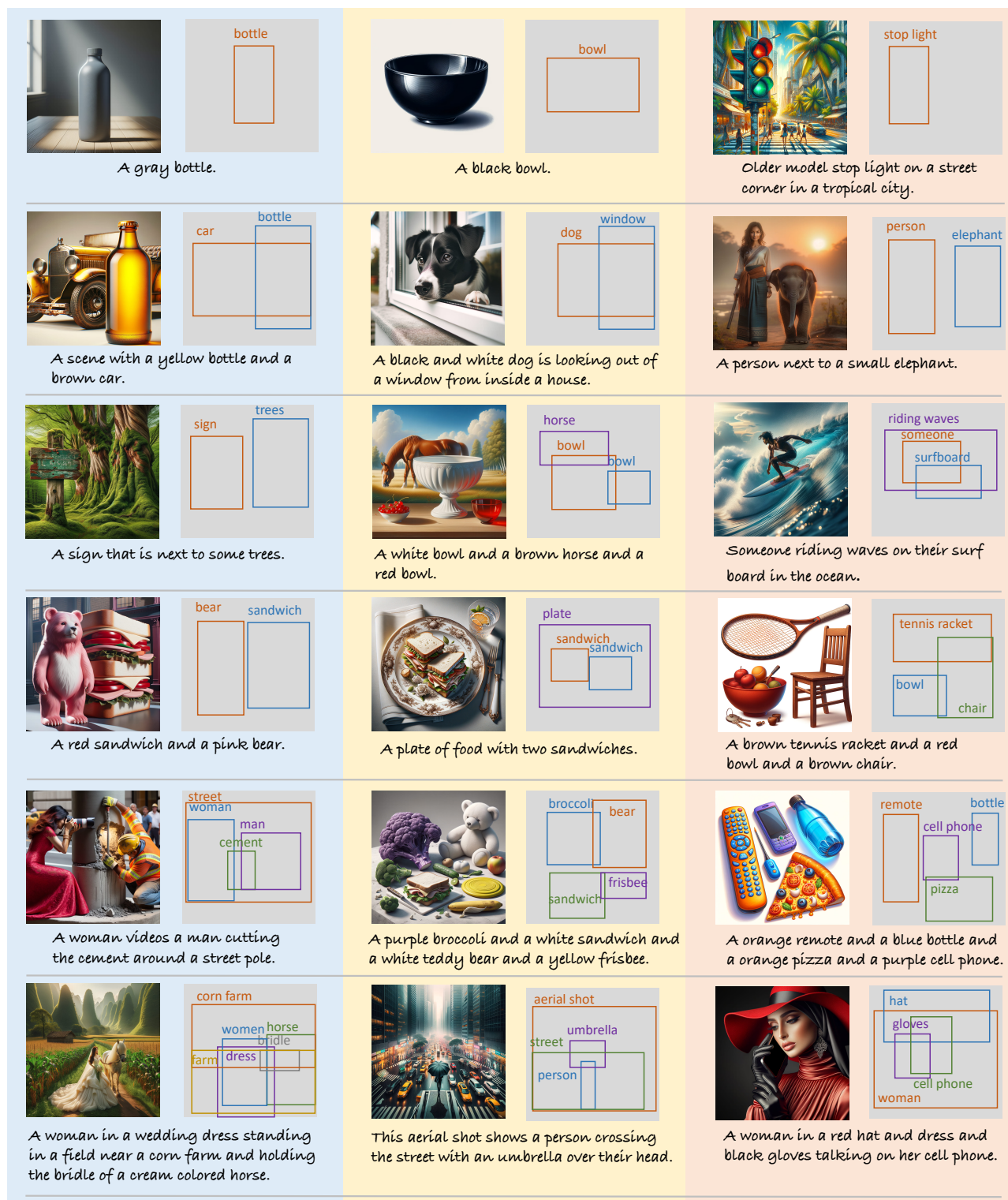


Figure 2: Visualizations of the data in GOAL2K. A single data of GOAL2K contains the prompt, the input image, and the layout conditions.

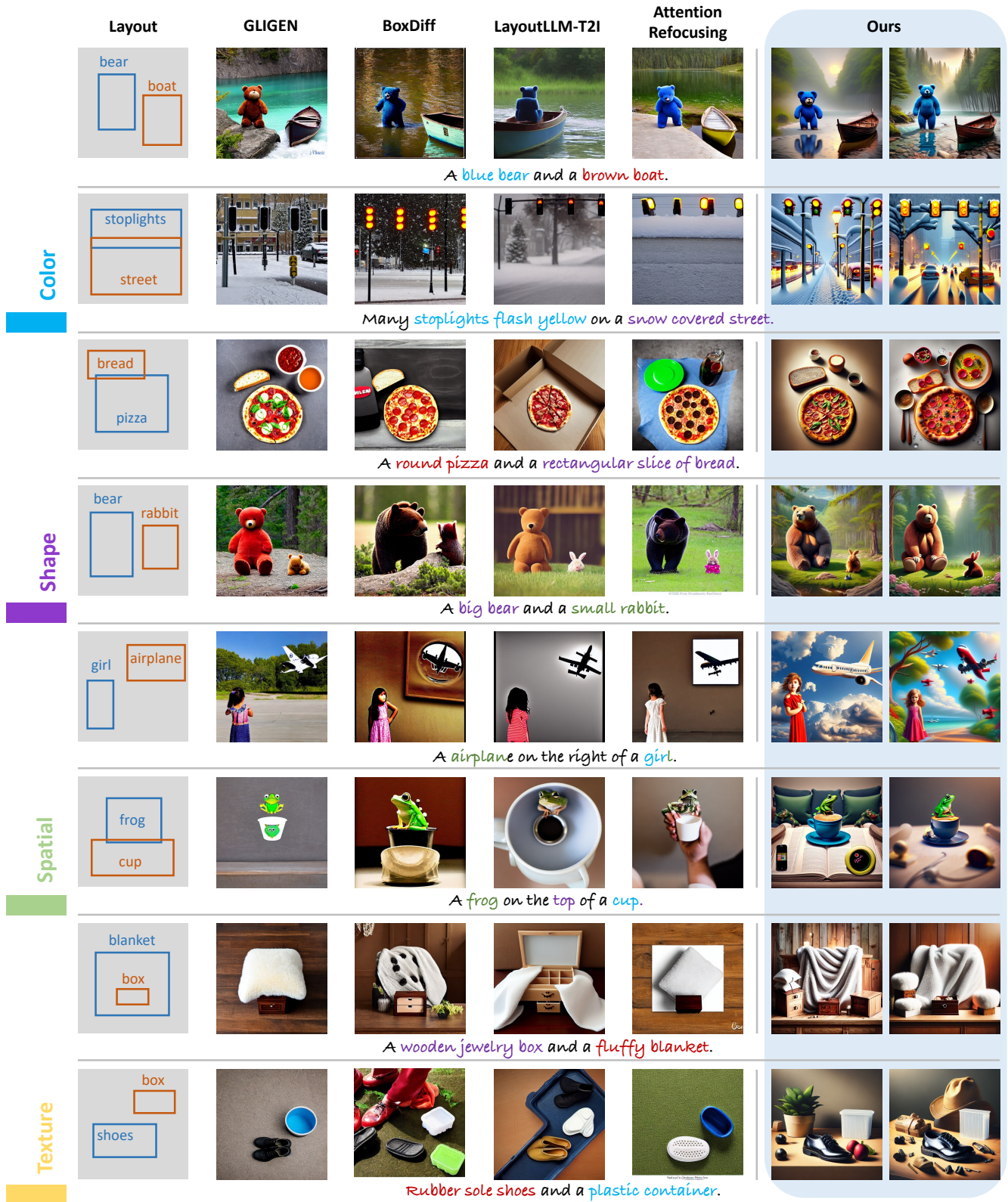


Figure 3: Qualitative results from T2I-Compbench for various attributes such as color, spatial, shape and texture. We demonstrate the effectiveness of the proposed method in text-image alignment compared with other layout-to-image methods, including GLIGEN [6], BoxDiff [13], LayoutLLM-T2I [10], Attention Refocusing [9].

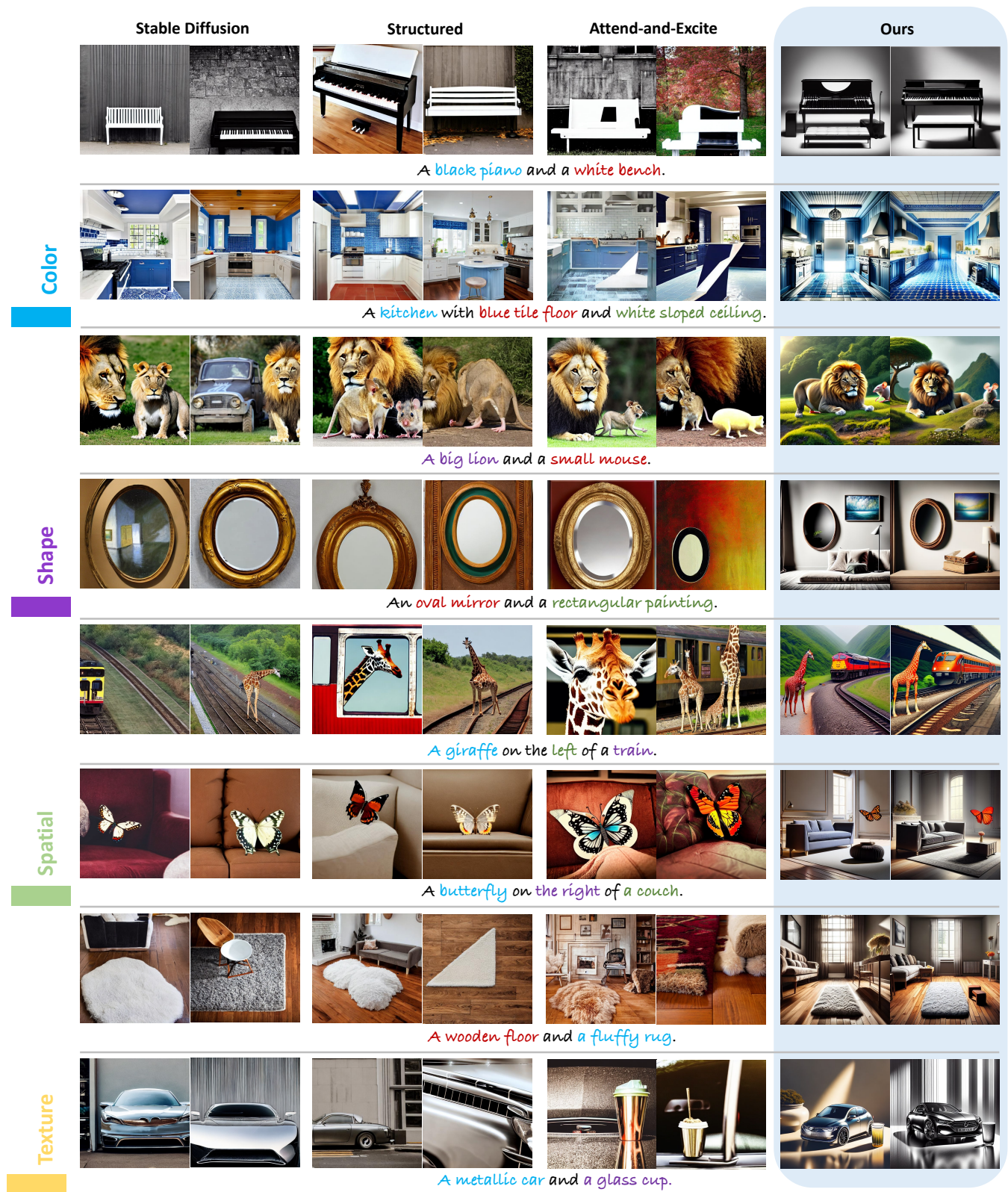


Figure 4: Qualitative results from T2I-Compbench for various attributes such as color, spatial, shape and texture. We demonstrate the effectiveness of the proposed method in text-image alignment compared with text-to-image methods, including Stable Diffusion [12], Structure [4], Attend-and-Excite [3].