Supplementary Material: Benchmarking Layout-Guided Diffusion Models through Unified Semantic-Spatial Evaluation in Closed and Open Settings

Luca Parolari* Nicla Faccioli Lamberto Ballan
Department of Mathematics, University of Padova, Padova, Italy

1. Sensitivity Analysis for the Unified Score

To examine how the unified metric responds to changes in the inputs, we perform a sensitivity analysis in which the instruction set of the closed-set benchmark (C-Bench) is progressively perturbed. The goal of this experiment is to verify whether the unified score decreases as the correspondence between the instructions and the visual content of the generated images is gradually degraded.

Starting from the original C-Bench, we construct several modified variants by replacing a portion of its prompt-layout pairs with randomly generated instructions obtained through our automatic pipeline. These substituted instructions are structurally valid but intentionally mismatched with the images they are evaluated against. We quantify the amount of perturbation through a perturbation ratio defined as the fraction of instructions that are replaced: a ratio of 0.0 corresponds to the unaltered C-Bench, while ratios of 0.25, 0.50, 0.75, and 1.0 represent increasingly severe levels of perturbation. To ensure that perturbation does not introduce unintended biases, replacements are performed independently within each scenario and for each object-count category, preserving the original distribution across both axes. Due to computational constraints and the cost of synthesizing LLM-based prompts, perturbations are applied only to the template-driven scenarios of C-Bench, while the complex-composition scenario is excluded.

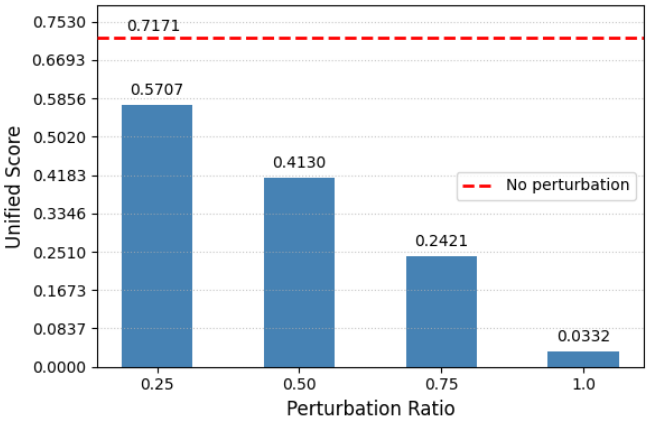


Figure 1. Sensitivity of the unified score under increasing perturbation ratios. Each bar reports the unified score obtained when a given fraction of the original C-Bench instructions is replaced with randomly generated ones and evaluated against previously generated images by MIGC [9]. As the perturbation ratio increases, the unified score decreases monotonically, indicating that the metric responds consistently and proportionally to mismatches between the evaluation inputs and the image content.

We evaluate each perturbed instruction set by measuring its alignment with the set of images generated on C-Bench using our evaluation framework. Fig. 1 reports the unified score for each perturbation ratio on images previously obtained with MIGC [9]. As the ratio increases—meaning that a larger portion of the benchmark is replaced with mismatched instruction-image pairs—the unified score decreases smoothly and monotonically, confirming that the metric reacts proportionally to the degree of perturbation introduced. To provide a more granular view, Fig. 2 illustrates how perturbations affect both text-alignment and layout-alignment scores across individual scenarios. The charts show a clear degradation in semantic and spatial alignment as the perturbation ratio increases, demonstrating that both metrics are sensitive to mismatches in the evaluation inputs—an effect that is consistently captured and reflected in the unified score. Together, these results demonstrate that the unified metric—and its text and layout components—are sensitive to inconsistencies between prompts, layouts, and image content, and respond predictably as the evaluation inputs are perturbed. This aligns our metric to human expectations.

*Corresponding author. Email: luca.parolari@studenti.unipd.it

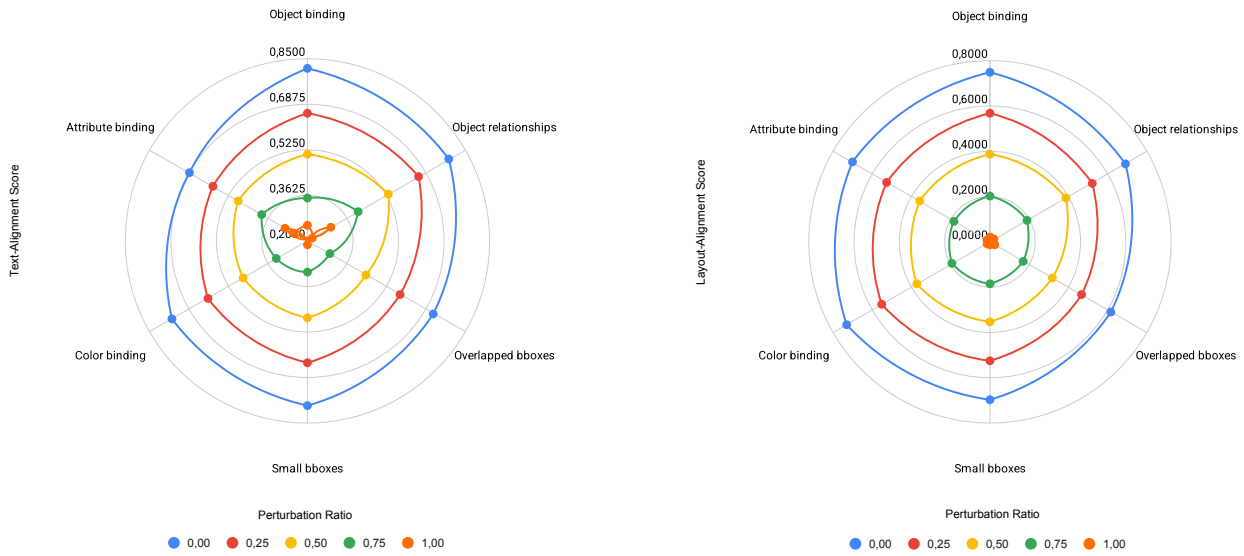


Figure 2. Sensitivity analysis of text (left) and layout (right) alignment across scenarios. Each chart reports per-scenario scores for increasing perturbation ratios. As more instructions are replaced with mismatched ones, both textual and layout alignment scores decline consistently across scenarios.

2. Ranking Stability

We compare the ranking induced by the unified score with those obtained using simple and weighted means of the text and layout alignment scores. Specifically, we consider aggregations of the form $\lambda \cdot s_{\text{text}} + (1 - \lambda) \cdot s_{\text{layout}}$, with $\lambda \in \{0.0, 0.1, \ldots, 1.0\}$, where the simple mean is the special case $\lambda = 0.5$. For each value of λ, we compute the resulting model ranking and compare it with the unified-score ranking using Kendall's τ and Spearman's ρ. The simple mean yields perfect agreement with the unified ranking, and weighted combinations remain highly consistent overall, with average Kendall's $\tau = 0.9152$ and Spearman's $\rho = 0.9584$ across all values of λ. These results show that the model ordering induced by the unified score is robust to reasonable changes in the aggregation rule, and therefore does not depend on a specific weighting of semantic and spatial alignment. At the same time, the unified score retains an important advantage over linear averages: by construction, it penalizes imbalanced performance, preventing strong alignment on one dimension from compensating for weak alignment on the other. This makes it particularly suitable for layout-guided generation, where both semantic correctness and spatial fidelity are necessary to satisfy the input guidance.

3. Human Validation of Pipeline Components

To assess the reliability and sensitivity of the key components in our evaluation framework, we perform two user studies designed to compare automated measures with human judgment. These studies evaluate whether our semantic and spatial alignment metrics behave consistently according to human perception.

Semantic Alignment We measure the agreement on semantic alignment between TIFA [2] and human judgments through a controlled binary evaluation task. We select a subset of 640 generated images from different models and manually annotate their semantic correctness with respect to the input prompt. For each image, human evaluators assign a positive label only when two conditions were simultaneously satisfied: (i) all objects mentioned in the prompt are present in the image (element presence), and (ii) all associated attributes are correctly bound to their respective objects (attribute correctness). If either condition failed, the image receives a negative label. TIFA is evaluated under an analogous binary scheme. For each image, TIFA generates a set of VQA-based queries corresponding to the elements and attributes specified in the prompt. An image is labeled as positive if all TIFA queries returned the correct answer-i.e., if the VQA finds every object along with the attributes described in the prompt. Otherwise, the image is considered negative. By comparing the outcomes of the human annotations with those produced by TIFA (Fig. 3), we observe strong overall consistency between the two, with small discrepancies in cases involving color attributes, where subjective human interpretation may vary.

Spatial Alignment Second, we assessed the reliability of the object detector [5] employed for layout-alignment evaluation. A subset of 448 generated images was manually annotated with bounding boxes to evaluate layout adherence, and detector predictions were compared with human annotations by measuring their spatial overlap. The evaluation, reported in Fig. 4, showed closely aligned behaviors, with humans exhibiting slightly higher localization precision.

Figure 3. Semantic alignment between given prompts and generated images obtained through TIFA [2] and human judgements on a subset of images from object and attribute binding scenarios. SD = Stable Diffusion [7], SD_CAG = Cross-Attention Guidance [1], G = GLIGEN [4], G_BD = BoxDiff [8], G_AR = Attention Refocusing [6].

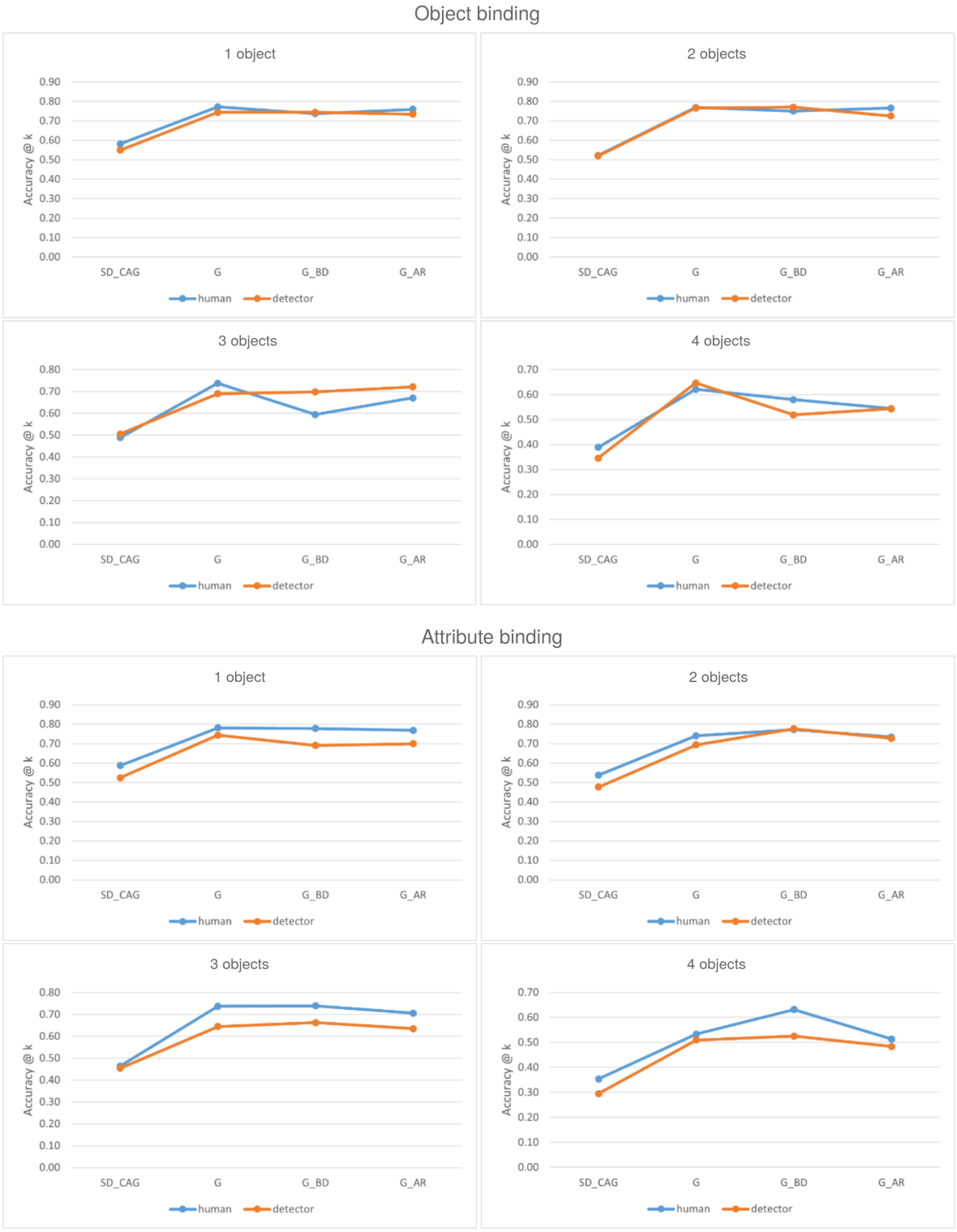
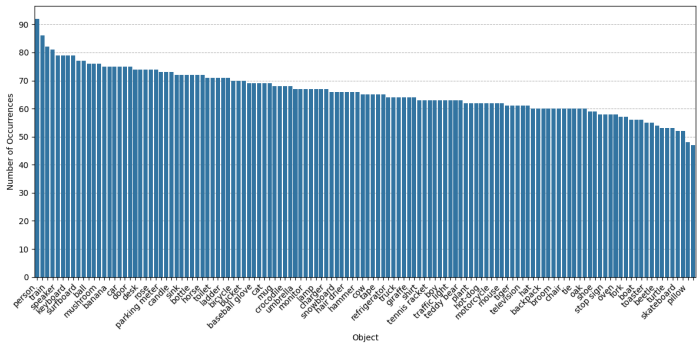


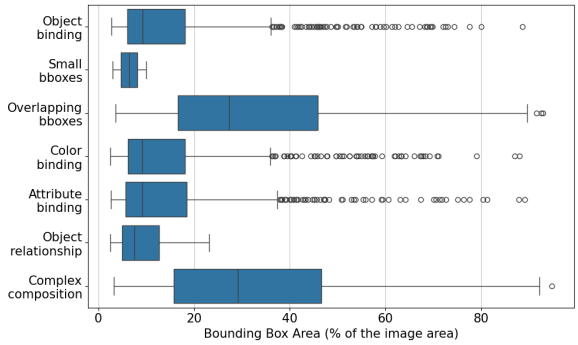
Figure 4. Spatial alignment between given layouts and generated images obtained through the object detector [5] and human judgements on a subset of images from object and attribute binding scenarios. SD = Stable Diffusion [7], SD_CAG = Cross-Attention Guidance [1], G = GLIGEN [4], G_BD = BoxDiff [8], G_AR = Attention Refocusing [6].

4. Objects and Bounding Boxes Distribution in C-Bench

To better understand the characteristics of C-Bench, we provide an analysis of its object and layout distributions. Fig. 5a reports the frequency of objects across all prompts, highlighting the coverage and balance of categories represented in the benchmark. Fig. 5b shows the distribution of bounding box areas for each scenario, illustrating how object sizes vary across different settings. We recall that bounding boxes in C-Bench are generated automatically. To satisfy the constraints imposed by each prompt or scenario, we sample bounding boxes at different sizes. The size depends on the number of objects described in the prompt: single-object prompts allow larger boxes (150-500 px per side), while prompts with two, three, or four objects use progressively smaller ranges (120-250, 100-180, and 80-150 px, respectively) to minimize overlap and reduce spatial conflicts. Together, these analyses confirm that C-Bench offers both semantic diversity and spatial variability, while maintaining controlled conditions for systematic evaluation.



(a) Frequency of objects across all prompts on C-Bench.



(b) Distribution of bounding boxes area per scenario. Values are displayed as percentage with respect to the image size.

Figure 5. Analysis of the C-Bench. On the left we report the frequency of objects across all prompts, while on the right we show the distribution of bounding box areas per scenario.

5. Object Distribution in O-Bench

As described in the main paper, O-Bench is derived from the Flickr30k Entities test split, which contains 4,969 caption–bounding-box pairs. Our sampling procedure preserves the original distribution of object counts while reducing the dataset size for computational feasibility. Tab. 6 reports the object-count distribution in the Flickr30k Entities test set, which serves as the basis for sampling. Captions naturally vary in compositional complexity, typically containing between 1 and 8 objects. Only a negligible fraction of captions describe 9 or more objects (less than 0.2%), forming the long tail of the distribution. We filtered out these extremely rare cases to avoid statistically unreliable categories while retaining virtually the entire dataset (99.83% of captions fall within 1–8 objects). Fig. 7 then shows the resulting distribution in O-Bench after sampling, which follows the same pattern while reducing the dataset size. The final benchmark contains 3,319 prompts, each paired with its corresponding set of object bounding boxes. This preserves the natural variability and compositional diversity of real-world captions while keeping the evaluation computationally manageable.

6. Object, Attribute, and Relation Sets

For template-based prompt generation in C-Bench, we define three sets: objects $\mathcal{O}$, attributes $\mathcal{A}$, and relations $\mathcal{R}$. We detail each of them in Tab. 1, Tab. 2 and Tab. 3 respectively.

Objects	Count	Cumulative	Cumulative (%)
1	490	490	9.86
2	1585	2075	41.76
3	1579	3654	73.54
4	838	4492	90.40
5	281	4773	96.06
6	117	4890	98.41
7	41	4931	99.24
8	25	4956	99.74
9	5	4961	99.84
10	4	4965	99.92
11	1	4966	99.94
12	1	4967	99.96
13	1	4968	99.98
14	1	4969	100.00

Figure 6. Object-count distribution in the Flickr30k Entities test set.

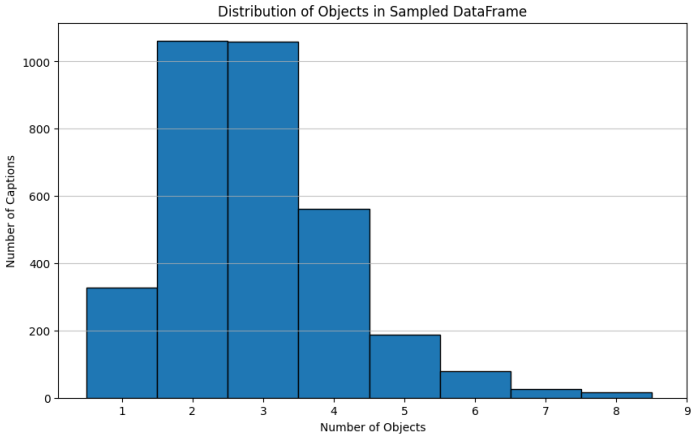


Figure 7. Object-count distribution in O-Bench after sampling from Flickr30k Entities.

rose	oak	beetle	skyscraper	tree
baby	bed	lamp	dog	laptop
bicycle	person	car	bus	cat
book	chair	boy	couch	table
plant	toilet	cellphone	microwave	sheep
boat	banana	stop sign	donut	cow
clock	bottle	umbrella	bird	guitar
toothbrush	parking meter	bench	platypus	keyboard
baseball bat	vase	surfboard	tiger	train
flower	sandwich	spoon	pizza	carrot
teddy bear	hot-dog	skateboard	kite	broom
apple	handbag	horse	snowboard	giraffe
tie	shower	traffic light	bear	toaster
knife	baseball glove	crocodile	suitcase	fork
cake	cup	bowl	hair drier	elephant
mouse	mushroom	motorcycle	turtle	tennis racket
truck	zebra	fire hydrant	oven	sink
frisbee	hat	ruler	shoe	ball
candle	ladder	charger	mug	tape
shirt	pillow	pan	plate	shampoo
hammer	blender	basket	screwdriver	wallet
bin	leaf	bucket	monitor	watch
flashlight	sock	door	scarf	speaker
desk	backpack	printer	remote	glass
curtain	toolbox	drill	notebook	television
soap	ring	refrigerator		

Table 1. List of objects © used by the Prompt Generation Engine to instantiate a template.

aggressive	dark	large	short	tall
black	fast	pink	silver	warm
blue	fluffy	red	small	white
bright	fuzzy	rotten	smooth	wooden
clean	green	rough	snowy	yellow
crowded	happy	shiny	soft	

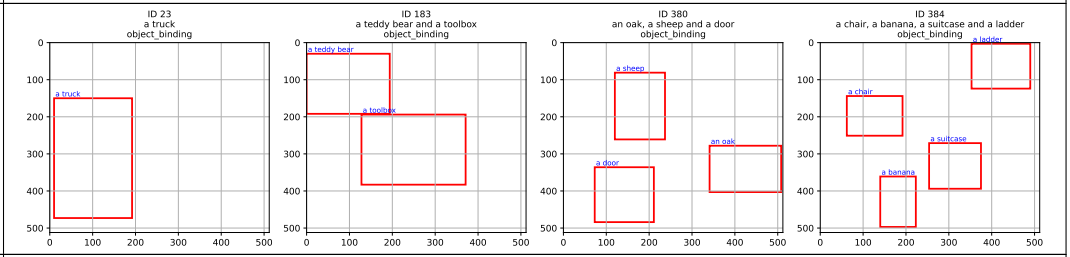
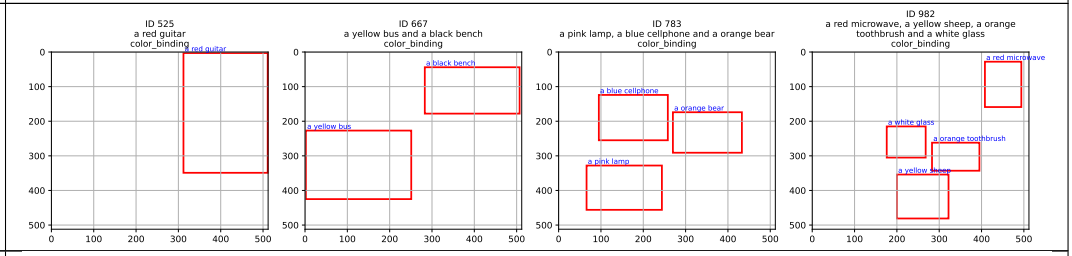
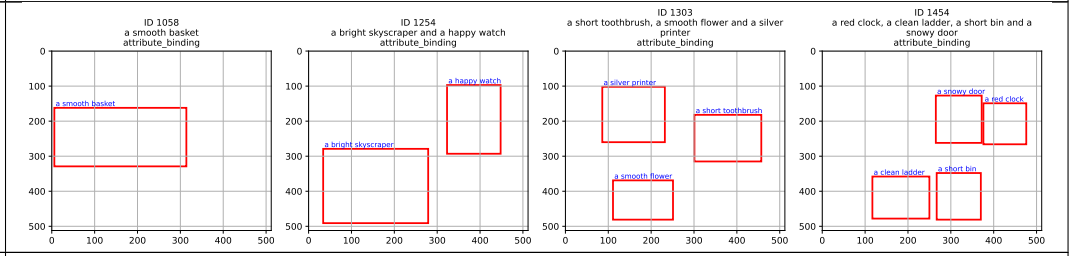
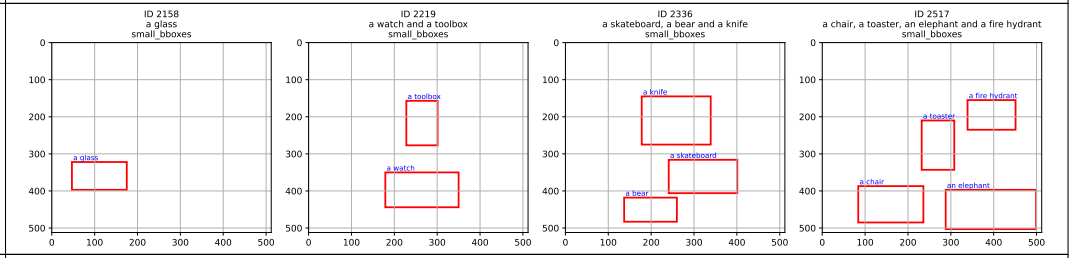
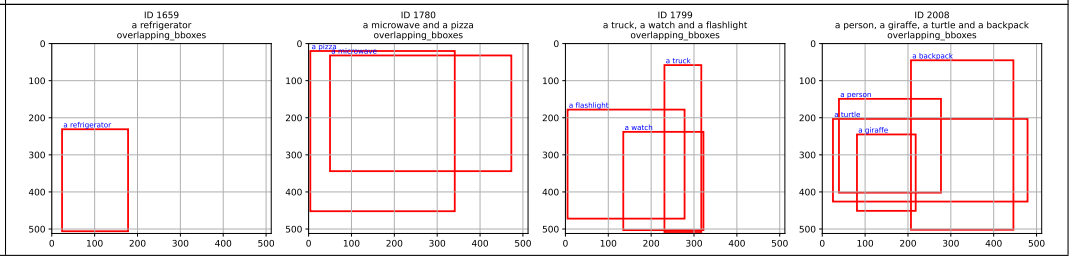
Table 2. List of attributes $\mathcal{A}$ used by the Prompt Generation Engine to instantiate a template.

above	far from	next to	over	to the right of
below	near	on	under	to the left of
beside				

Table 3. List of relations $\mathcal{R}$ used by the Prompt Generation Engine to instantiate a template.

7. Qualitative Examples of Instructions

We present qualitative examples of the instruction set in Fig 4, consisting of paired prompts and layouts. These examples illustrate the diversity and complexity of scenarios included in C-Bench, ranging from simple single-object cases to more challenging multi-object compositions involving attributes and spatial relations, as well as natural prompts in O-Bench. Each instruction defines both the textual description and the corresponding spatial arrangement, ensuring precise control over what is generated and where it should appear. The first figure showcases representative samples across different scenarios and object counts, highlighting how the benchmark systematically isolates key generative challenges while maintaining scalability and interpretability.

	Scenario	Instruction
1	Object binding	
2	Color binding	
3	Attribute binding	
4	Small bboxes	
5	Overlapped bboxes	

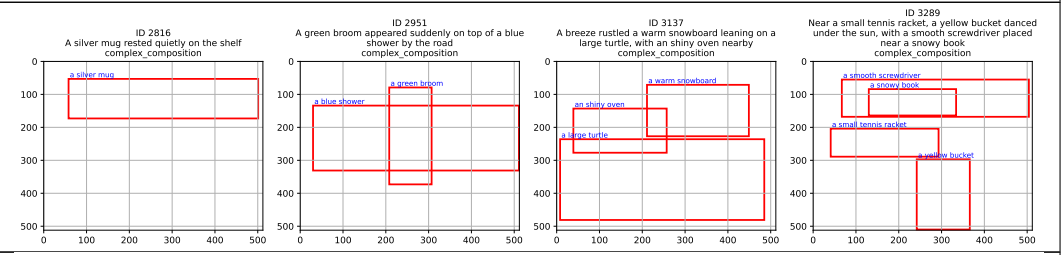
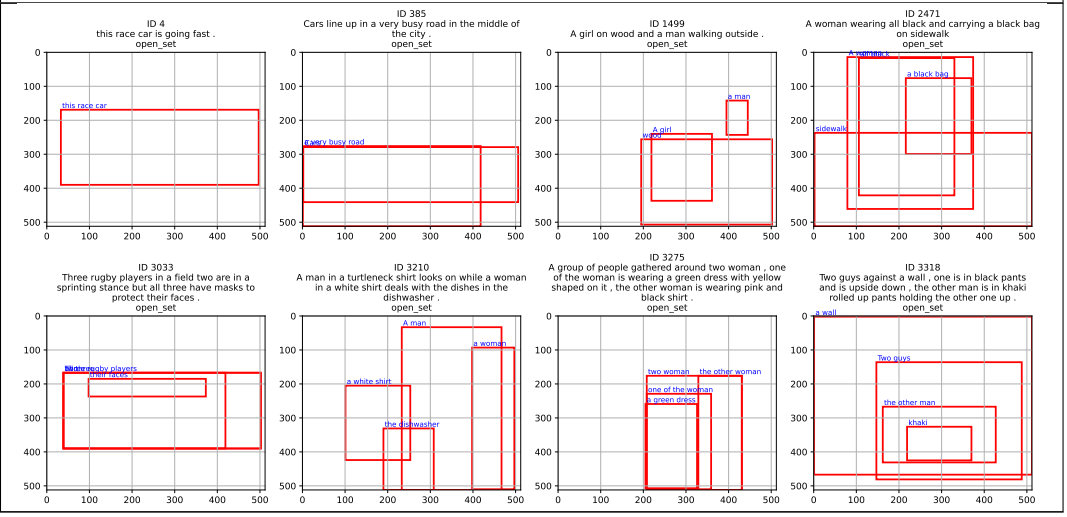
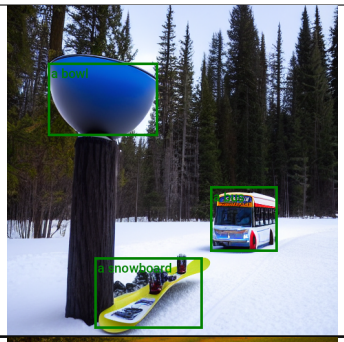
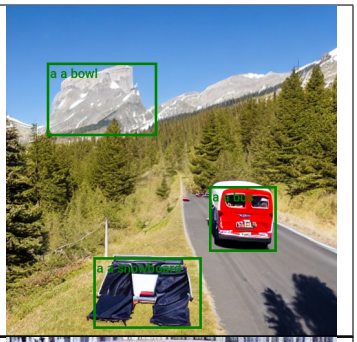
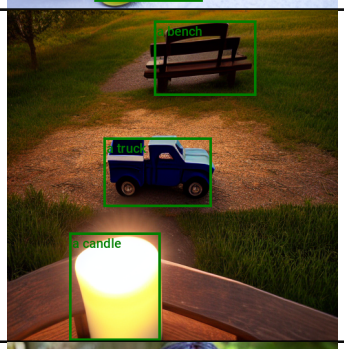
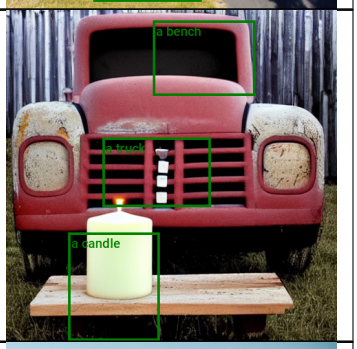
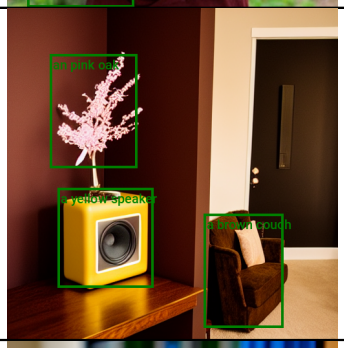
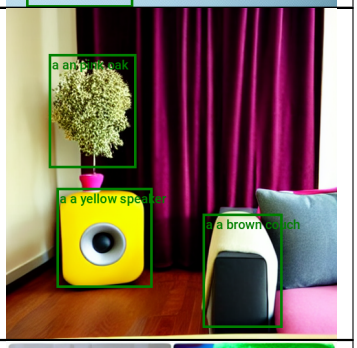
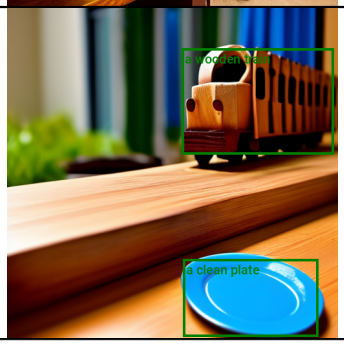
6	Object relationship	
7	Complex composition	
8	Open set	

Table 4. Examples of instructions (prompt-layout pairs) from both C-Bench (rows 1-7) and O-Bench (row 8).

8. Qualitative Examples of Generated Images

To provide a qualitative perspective on model behavior, we include additional examples of generated images in Tab. 5. The figure shows, for each scenario in C-Bench successful generations, where objects are correctly rendered with the expected attributes and spatial configuration as well as failure cases, which highlight common challenges such as missing objects, incorrect attributes, or misplaced elements.

	Scenario	Prompt	Success	Fail

1	Object binding	A bus, a snowboard and a bowl.		
2	Small bboxes	A bench, a truck and a candle.		
3	Overlapping bboxes	A book, a scarf and a soap.		
4	Color binding	A pink oak, a brown couch and a yellow speaker.		
5	Attribute binding	A wooden train and a clean plate.		

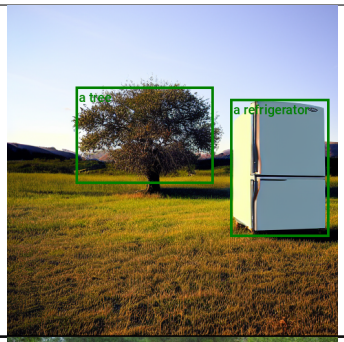
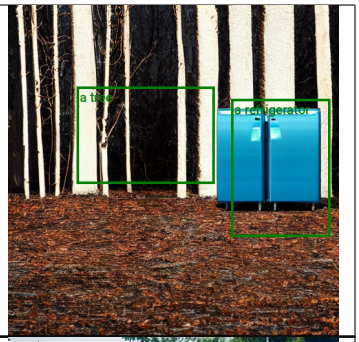
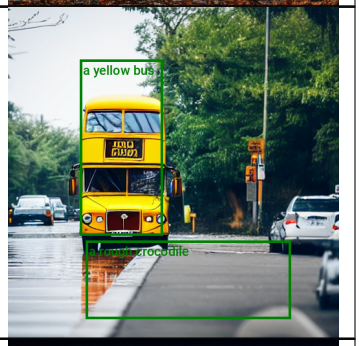
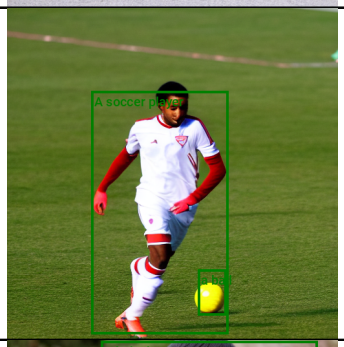
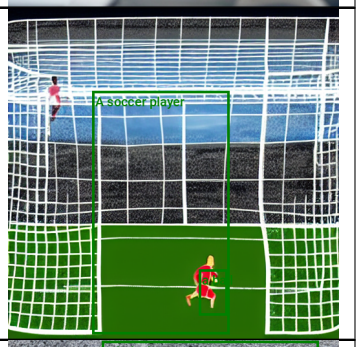
6	Object relationship	A refrigerator beside a tree.		
7	Complex composition	A yellow bus stood above a rough crocodile in the quiet morning.		
8	Open set	A soccer player is running while kicking a ball.		
9	Open set	A baby boy in overalls is crying.		

Table 5. Examples of generated images from both C-Bench (rows 1-7) and O-Bench (rows 8-9). The Success column illustrates correct generations, where all objects appear with the intended attributes, while the Fail column shows cases where generation errors occur.

9. Custom LLM Prompt for Generation of Complex Compositions

As described in Sec. 2 in the main paper, we used large language models, specifically ChatGPT [3], to generate prompts for the complex composition scenario, where all previously isolated challenges are combined into a single setting. The listing below shows the custom prompt used for $N = 4$ objects per sentence, which can be easily adapted to cases with 1–3 objects.

```
Generate 128 natural compositional phrases with various structures and creativity.
```

Each prompt must describe a scene involving exactly 4 unique objects. Objects can be reused across multiple prompts, but each object must appear only once within any given prompt. Each object must be enriched with at least one descriptive attribute, which may describe :

Color, shape, material, appearance, or dimension

Or a spatial relation between objects in the same prompt

Each prompt should be a short, vivid sentence that situates the objects in a dynamic or descriptive context, similar to the following examples:

A bright pink flower swayed gently under the tall oak tree.

A black laptop rested beside a green coffee mug on the messy desk.

Avoid passive or overly generic constructions-aim for imaginative, specific scenarios.

Adjust the articles of the objects if needed.

Object List (You can freely reuse any of these objects across different prompts, but not within the same prompt.)

```
obj_with_articles = [
  'a rose', 'an oak', 'a beetle', 'a skyscraper', 'a tree', 'a baby', 'a bed', 'a lamp', 'a
      dog', 'a laptop', 'a bicycle', 'a person', 'a car', 'a bus', 'a cat', 'a book', 'a
      chair', 'a boy', 'a couch', 'a table', 'a plant', 'a toilet', 'a cellphone', 'a
      microwave', 'a sheep', 'a boat', 'a banana', 'a stop sign', 'a donut', 'a cow', 'a
      clock', 'a bottle', 'an umbrella', 'a bird', 'a guitar', 'a toothbrush', 'a parking
      meter', 'a bench', 'a platypus', 'a keyboard', 'a baseball bat', 'a vase', 'a
      surfboard', 'a tiger', 'a train', 'a flower', 'a sandwich', 'a spoon', 'a pizza', 'a
      carrot', 'a teddy bear', 'an hot-dog', 'a skateboard', 'a kite', 'a broom', 'an apple
      ', 'a handbag', 'a horse', 'a snowboard', 'a giraffe', 'a tie', 'a shower', 'a traffic
       light', 'a bear', 'a toaster', 'a knife', 'a baseball glove', 'a crocodile', 'a
      suitcase', 'a fork', 'a cake', 'a cup', 'a bowl', 'a hair drier', 'an elephant', 'a
      mouse', 'a mushroom', 'a motorcycle', 'a turtle', 'a tennis racket', 'a truck', 'a
      zebra', 'a fire hydrant', 'an oven', 'a sink', 'a frisbee', 'a hat', 'a ruler', 'a
      shoe', 'a ball', 'a candle', 'a ladder', 'a charger', 'a mug', 'a tape', 'a shirt', 'a
       pillow', 'a pan', 'a plate', 'a shampoo', 'a hammer', 'a blender', 'a basket', 'a
      screwdriver', 'a wallet', 'a bin', 'a leaf', 'a bucket', 'a monitor', 'a watch', 'a
      flashlight', 'a sock', 'a door', 'a scarf', 'a speaker', 'a desk', 'a backpack', 'a
      printer', 'a remote', 'a glass', 'a curtain', 'a toolbox', 'a drill', 'a notebook', 'a
       television', 'a soap', 'a ring', 'a refrigerator'
]
```

Allowed Attributes (for appearance or material):

```
attributes = [ 'aggressive', 'black', 'blue', 'bright', 'clean', 'crowded', 'dark', 'fast',
    'fluffy', 'fuzzy', 'green', 'happy',
'large', 'pink', 'red', 'rotten', 'rough', 'shiny', 'short', 'silver', 'small', 'smooth', '
    snowy', 'soft', 'tall', 'warm', 'white', 'wooden', 'yellow'
]
```

Allowed Colors (subset of attributes):

```
colors = ['black', 'blue', 'brown', 'gray', 'green', 'pink', 'purple', 'red', 'white', '
    yellow', 'orange']
```

```
Spatial Relations (to be used as part of attributes or composition logic):

spatial\_relations = [
'on top of', 'beside', 'under', 'above', 'next to', 'beneath', 'behind', 'in front of', '
    between', 'leaning on',
'inside', 'resting on', 'attached to', 'surrounded by', 'placed near'
]

Output Format:

Return the result as a CSV with two columns:

prompt, object1, object2, ..

Each row should contain:

The generated sentence

The noun chunks used

Example output format (for N=2):

prompt,object1,object2

A fluffy cat jumped onto the soft couch near the window" ,a fluffy cat, a soft couch

A red bicycle leaned against a wooden bench in the park ,a red bicycle, a wooden bench


The prompts should be inside quotes if needed to keep the correct number of columns in the
    CSV.

In the sentence, keep noun chunks unbroken, adjectives modifying a noun should not be split
    by commas. Treat the entire noun chunk as a single unit (e.g., "a small red ball", not "
    a small, red ball").

The articles should remain consistent between the prompt and the noun chunks.
```

References

[1] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pages 5331–5341. IEEE, 2024. 4, 5

[2] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. TIFA: accurate and interpretable text-to-image faithfulness evaluation with question answering. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 20349–20360. IEEE, 2023. 3, 4

[3] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 12

[4] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: open-set grounded text-to-image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22511–22521. IEEE, 2023. 4, 5

[5] Matthias Minderer, Alexey A. Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple open-vocabulary object detection with vision transformers. *CoRR*, abs/2205.06230, 2022. 3, 5

[6] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 7932–7942. IEEE, 2024. 4, 5

[7] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent

diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022. 4, 5

[8] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 7418–7427. IEEE, 2023. 4, 5

[9] Dewei Zhou, You Li, Fan Ma, Xiaoting Zhang, and Yi Yang. MIGC: multi-instance generation controller for text-to-image synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 6818–6828. IEEE, 2024. 1