Evaluating and Improving Compositional Text-to-Visual Generation

Supplementary Material

Outline

This document supplements the main paper with benchmark and method details. Below is the outline:

- **Section A** presents additional qualitative studies.
- **Section B** details GenAI-Bench's evaluated aspects.
- **Section C** describes how we collect GenAI-Bench.
- **Section D** describes how we compute VQAScore.
- **Section E** describes how CLIP-FlanT5 is trained.
- **Section F** discusses other baseline methods.
- **Section G** discusses other alignment benchmarks.

A. Additional Examples

Ranking SD-XL images by VQAScore. Figure 9 shows that ranking by VQAScore can also improve the prompt alignment of SD-XL using only its image generation API. We encourage future work to explore other white-box techniques for finetuning [21, 33, 36, 38, 57, 66, 70].

B. Evaluated Aspects of GenAI-Bench

We now detail the evaluated aspects of GenAI-Bench.

Skill definitions. Most literature on text-to-visual generation [6, 25, 26, 58, 80] primarily focuses on generating *basic* objects, attributes, relations, and scenes. While these "basic" visual compositions still pose challenges, real-world user prompts often introduce greater complexity. Such prompts require higher-order reasoning beyond basic compositions, including comparison, differentiation, counting, and logic. For example, while existing benchmarks focus only on counting objects [25, 80], real-world prompts often require counting attribute-object pairs or even object-relation-object triplets, like "`one person wearing a white shirt and the other five wearing blue shirts`". To this end, after thoroughly reviewing relevant literature [26, 48, 69, 80], we define a set of compositional reasoning skills common in real-world prompts, categorizing them into "basic" and "advanced", where the latter can build upon the former. For logical reasoning, we consider "negation" and "universality", which are the two most common types of logic we see in real-world prompts. We provide detailed definitions for "basic" skills in Table 4 and "advanced" skills in Table 5.

Comparing skills across benchmarks. We find the skill categorization in benchmarks like PartiPrompt [80] to be ambiguous or even confusing. For example, PartiPrompt introduces two categories "*complex*" and "*fine-grained detail*". The former refers to "*...fine-grained, interacting details or relationships between multiple participants*", while the latter refers to "*...attributes or actions of entities or objects in a scene*". Upon closer examination, the categorization of spatial, action, and part relations into these categories appears arbitrary. To address this, we attempt to compare the skill coverage across all benchmarks by our unified set of skills. For benchmarks (PartiPrompt/T2I-CompBench) with pre-defined skill categories, we map their skills to our definitions. For benchmarks (TIFAv1/Pick-a-pic/DrawBench/EditBench/HPDv2/EvalCrafter) without a comprehensive skill set, we manually annotate a random subset of samples. Finally, we calculate the skill proportions in each benchmark, identifying skills that constitute more than 2% as genuinely present.

C. GenAI-Bench

This section describes how we collect GenAI-Bench.

Details of GenAI-Bench. GenAI-Bench consists of 1,600 diverse prompts that cover advanced skills not addressed in previous benchmarks [26, 58, 80]. To source prompts relevant to real-world applications, we employ two graphic designers experienced in text-to-visual tools like Midjourney [48]. First, we introduce them to our skill definitions and examples. Then, we ask them to craft prompts for each skill, collaborating with ChatGPT to brainstorm prompt variants across diverse visual domains. Importantly, these designers ensure that the prompts are *objective*. This contrasts with T2I-CompBench [26], whose prompts are almost entirely auto-generated. For example, in T2I-CompBench's "*texture*" category, an overwhelming 40% of the 1000 programmatically-generated prompts use "metallic" as the attribute, which limits their diversity. Other T2I-CompBench's prompts generated by ChatGPT often contain subjective (non-visual) phrases. For instance, in the prompt "`the delicate, fluttering wings of the butterfly signaled the arrival of spring, a natural symbol of rebirth and renewal`", the "rebirth and renewal" can convey different meanings to different people. Similarly, in "`the soft, velvety texture of the rose petals felt luxurious against the fingertips, a romantic symbol of love and affection`", the "love and affection" is also open to diverse interpretations. Thus, we carefully guide the designers to avoid such prompts. Lastly, each prompt in GenAI-Bench is tagged with all its evaluated aspects. We streamline this process by using GPT4 for automatic tagging, providing it the skill definitions and in-context exemplars. Later, we manually verify and correct all tags

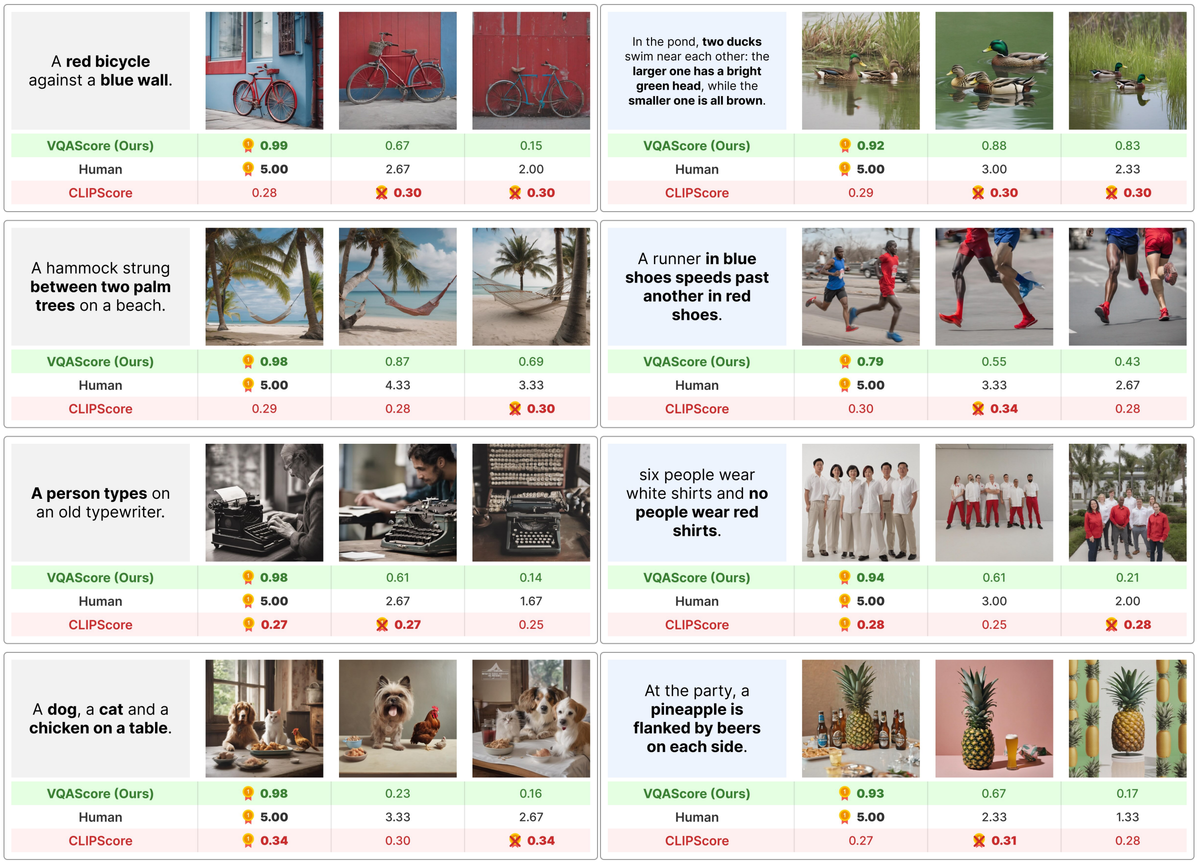
A red bicycle against a blue wall.
VQAScore (Ours) 0.99 0.67 0.15
Human 5.00 2.67 2.00
CLIPScore 0.28 0.30 0.30
In the pond, two ducks swim near each other: the larger one has a bright green head, while the smaller one is all brown.
VQAScore (Ours) 0.92 0.88 0.83
Human 5.00 3.00 2.33
CLIPScore 0.29 0.30 0.30
A hammock strung between two palm trees on a beach.
VQAScore (Ours) 0.98 0.87 0.69
Human 5.00 4.33 3.33
CLIPScore 0.29 0.28 0.30
A runner in blue shoes speeds past another in red shoes.
VQAScore (Ours) 0.79 0.55 0.43
Human 5.00 3.33 2.67
CLIPScore 0.30 0.34 0.28
A person types on an old typewriter.
VQAScore (Ours) 0.98 0.61 0.14
Human 5.00 2.67 1.67
CLIPScore 0.27 0.27 0.25
six people wear white shirts and no people wear red shirts.
VQAScore (Ours) 0.94 0.61 0.21
Human 5.00 3.00 2.00
CLIPScore 0.28 0.25 0.28
A dog, a cat and a chicken on a table.
VQAScore (Ours) 0.98 0.23 0.16
Human 5.00 3.33 2.67
CLIPScore 0.34 0.30 0.34
At the party, a pineapple is flanked by beers on each side.
VQAScore (Ours) 0.93 0.67 0.17
Human 5.00 2.33 1.33
CLIPScore 0.27 0.31 0.28

Figure 9. **Ranking SD-XL generated images with VQAScore and CLIPScore.** VQAScore outperforms CLIPScore in ranking candidate images generated by SD-XL, particularly for advanced prompts that involve complex visio-linguistic reasoning.

for accuracy, resulting in over 5,000 human-verified tags.

Collecting human ratings. We evaluate six text-to-image models: Stable Diffusion [56] (SD v2.1, SD-XL, SD-XL Turbo), DeepFloyd-IF [12], Midjourney v6 [48], DALL-E 3 [2]; along with four text-to-video models: ModelScope [71], Floor33 [14], Pika v1 [52], Gen2 [17]. Due to the lack of APIs for Floor33 [14], Pika v1 [52], and Gen2 [17], we manually download videos from their websites. For image generative models, we generate images using all 1,600 GenAI-Bench prompts. We use a coreset of 800 prompts to collect videos for the four video models. The same 800 prompts are used to collect the ranking benchmark in the main paper. In total, we collect over 80,000 human ratings, greatly exceeding the scale of human annotations in previous work [5, 25], e.g., TIFA160 collected 2,400 ratings.

GenAI-Bench performance. We detail the performance of the ten image and video generative models across all skills in Table 6. Both humans and VQAScores rate DALL-E 3 [2] higher than the other models in nearly all skills, except for negation. In addition, prompts requiring "advanced" compositions are rated significantly lower by both humans and VQAScores, with negation being the most challenging skill. Lastly, current video models do not perform as well as image models, suggesting room for improvement.

D. Implementing VQAScore

In this section, we describe how we compute VQAScore.

Computing VQAScore as an auto-regressive product. Recall that VQAScore calculates the alignment score of an image $\mathbf{i}$ and text $\mathbf{t}$ directly from a VQA model. We first use a simple QA template to convert the text $\mathbf{t}$ to a question and an answer (denoted as $\mathbf{q}(\mathbf{t})$ and $\mathbf{a}(\mathbf{t})$), for example:

$$\mathbf{q}(\mathbf{t}) = \text{Does this figure show ''\{t\}''? Please answer yes or no.} \quad (2)$$

$$\mathbf{a}(\mathbf{t}) = \text{Yes} \quad (3)$$

We find that such a straightforward question-answer pair is sufficient for good performance across all benchmarks. In language modeling [1], a piece of text is pre-processed (or tokenized) into a token sequence, e.g., $\mathbf{a}(\mathbf{t}) = \{a_1, \cdots, a_m\}$. Although "`Yes`" usually counts as a single

Table 4. **Skill definitions and examples for basic compositions.**

Skill Type	**Definition**	**Examples**
	Basic Compositions	
`Object`	Basic entities within an image, such as person, animal, food, items, vehicles, or text symbols (e.g., "A", "1+1").	*a* **dog**, *a* **cat** *and a* **chicken** *on a* **table**; *a young* **man** *with a green* **bat** *and a blue* **ball**; *a '***No Parking***' sign on a busy street.*
`Attribute`	Visual properties of entities, such as color, material, emotion, size, shape, age, gender, state, and so on.	*a* **silver** *spoon lies to the left of a* **golden** *fork on a* **wooden** *table; a* **green** *pumpkin is smiling* **happily**, *a* **red** *pumpkin is sitting* **sadly**.
`Scene`	Backgrounds or settings of an image, such as weather, location, and style.	*A child making a sandcastle on a* **beach in a cloudy day**; *a grand fountain surrounded by historic buildings in a* **town square**.
`Spatial Relation`	Physical arrangements of multiple entities relative to each other, e.g., on the right, on top, facing, towards, inside, outside, near, far, and so on.	*a bustling city street, a neon 'Open 24 Hours' sign glowing* **above** *a small diner; a teacher standing* **in front of** *a world map in a classroom; tea steams* **in** *a cup,* **next to** *a closed diary with a pen resting* **on** *its cover.*
`Action Relation`	Action interactions between entities, e.g., pushing, kissing, hugging, hitting, helping, and so on.	*a dog* **chasing** *a cat; a group of children* **playing** *on the beach; a boat* **glides** *across the ocean, dolphins* **leaping** *beside it and seagulls* **soaring** *overhead.*
`Part Relation`	Part-whole relationships between entities – one entity is a component of another, such as body part, clothing, and accessories.	*a pilot* **with aviator sunglasses**; *a baker* **with a cherry pin on a polka dot apron**.; *a young lady* **wearing a T-shirt** *puts* **her hand** *on a* **puppy's head**.

Table 5. **Skill definitions and examples for advanced compositions.**

Skill Type	**Definition**	**Examples**
	Advanced Compositions	
`Counting`	Determining the quantity, size, or volume of entities, e.g., objects, attribute-object pairs, and object-relation-object triplets.	**two** *cats playing with a* **single** *ball;* **five** *enthusiastic athletes and* **one** *tired coach;* **one** *pirate ship sailing through space, crewed by* **five** *robots;* **three** *pink peonies and* **four** *white daisies in a garden.*
`Differentiation`	Differentiating objects within a category by their attributes or relations, such as distinguishing between "old" and "young" people by age, or "the cat on top of the table" versus "the cat under the table" by their spatial relations.	**one cat** *is sleeping on the table and* **the other** *is playing under the table; there are two men in the living room,* **the taller one** *to the left of* **the shorter one**; *a notebook lies open in the grass, with sketches on* **the left page** *and blank space* **on the right**; *there are two shoes on the grass,* **the one without laces** *looks newer than* **the one with laces**.
`Comparison`	Comparing characteristics like number, attributes, area, or volume between entities.	*there are* **more** *people standing than sitting; between the two cups on the desk, the* **taller** *one holds* **more** *coffee than the* **shorter** *one, which is half-empty; a small child on a skateboard has* **messier** *hair than the person next to him; three little boys are sitting on the grass, and the boy in the middle looks the* **strongest**.
`Negation`	Specifying the absence or contradiction of elements, as indicated by "no", "not", or "without", e.g., entities not present or actions not taken.	*four elephants,* **no** *giraffes; six people wear white shirts and* **no** *people wear red shirts; a bookshelf with* **no** *books, only picture frames.; a person with short hair is crying while a person with long hair* **is not**; *a smiling girl with short hair and* **no** *glasses.; a cute dog* **without** *a collar.*
`Universality`	Specifying when every member of a group shares a specific attribute or is involved in a common relation, indicated by words like "every", "all", "each", "both".	*in a room,* **all** *the chairs are occupied except one; a bustling kitchen where* **every** *chef is preparing a dish; in a square, several children are playing,* **each** *wearing a red T-shirt; a table laden with apples and bananas, where* **all** *the fruits are green; the little girl in the garden has roses in* **both** *hands.*

Table 6. **Performance breakdown on GenAI-Bench.** We present the averaged human ratings and VQAScores (based on CLIP-FlanT5) for "basic" and "advanced" prompts. Human ratings use a 1-5 Likert scale, and VQAScore ranges from 0 to 1, with higher scores indicating better performance for both. Generally, both human ratings and VQAScores favor DALL-E 3 over other models, with DALL-E 3 preferred across almost all skills except for negation. We find that "advanced" prompts that require higher-order reasoning present significant challenges. For instance, the state-of-the-art DALL-E 3 receives a remarkable average human rating of 4.3 for "basic" prompts, indicating the images and prompts range from "*having a few minor discrepancies*" to "*matching exactly*". However, it scores only 3.4 for "advanced" prompts, suggesting "*several minor discrepancies*". In addition, video models receive significantly lower scores than image models. Overall, VQAScores closely match human ratings.

Method	**Attribute**	**Scene**	**Relation**			**Avg**
			Spatial	Action	Part	
Image models						
SD v2.1	3.3	3.3	3.0	3.2	3.1	3.2
SD-XL Turbo	3.7	3.7	3.4	3.5	3.5	3.6
SD-XL	3.8	3.7	3.4	3.7	3.6	3.6
DeepFloyd-IF	3.7	3.7	3.7	3.7	3.6	3.7
Midjourney v6	4.0	3.9	3.7	4.0	4.0	3.9
DALL-E 3	4.3	4.5	4.2	4.2	4.2	4.3
Video models						
ModelScope	3.1	3.1	2.8	3.0	3.1	3.0
Floor33	3.2	3.2	2.9	3.2	3.1	3.1
Pika v1	3.4	3.4	3.1	3.3	3.2	3.3
Gen2	3.6	3.7	3.4	3.6	3.6	3.6

(a) Human ratings on "basic" prompts

Method	**Attribute**	**Scene**	**Relation**			**Avg**
			Spatial	Action	Part	
Image models						
SD v2.1	0.80	0.81	0.76	0.77	0.79	0.79
SD-XL Turbo	0.83	0.83	0.80	0.81	0.84	0.83
SD-XL	0.86	0.86	0.82	0.83	0.89	0.84
Midjourney v6	0.89	0.89	0.87	0.87	0.91	0.87
DALL-E 3	0.91	0.91	0.91	0.89	0.91	0.90
Video models						
ModelScope	0.69	0.69	0.65	0.65	0.70	0.66
Floor33	0.70	0.71	0.64	0.66	0.67	0.67
Pika v1	0.78	0.80	0.74	0.72	0.76	0.75
Gen2	0.79	0.81	0.74	0.76	0.83	0.77

(b) VQAScores on "basic" prompts

Method	**Count**	**Differ**	**Compare**	**Logical**		**Avg**
				Negate	Universal	
Image models						
SD v2.1	2.7	2.4	2.5	2.7	2.9	2.8
SD-XL	2.8	2.6	2.5	2.7	3.2	2.8
SD-XL Turbo	2.8	2.5	2.6	2.8	3.2	2.9
DeepFloyd-IF	3.1	2.8	2.9	2.8	3.3	3.0
Midjourney v6	3.3	3.1	3.1	2.9	3.5	3.2
DALL-E 3	3.4	3.3	3.4	2.8	3.7	3.4
Video models						
ModelScope	2.4	2.4	2.2	2.6	2.8	2.5
Floor33	2.7	2.7	2.5	2.8	3.2	2.8
Pika v1	2.7	2.7	2.6	2.9	3.3	2.9
Gen2	2.8	2.7	2.6	2.9	3.3	2.9

(c) Human ratings on "advanced" prompts

Method	**Count**	**Differ**	**Compare**	**Logical**		**Avg**
				Negate	Universal	
Image models						
SD v2.1	0.67	0.67	0.66	0.55	0.59	0.62
SD-XL	0.71	0.71	0.72	0.53	0.62	0.64
SD-XL Turbo	0.70	0.69	0.71	0.55	0.61	0.65
DeepFloyd-IF	0.70	0.69	0.71	0.52	0.64	0.65
Midjourney v6	0.76	0.78	0.77	0.53	0.70	0.70
DALL-E 3	0.80	0.81	0.77	0.53	0.72	0.71
Video models						
ModelScope	0.58	0.61	0.57	0.52	0.52	0.55
Floor33	0.60	0.64	0.59	0.53	0.55	0.57
Pika v1	0.65	0.64	0.63	0.55	0.63	0.61
Gen2	0.69	0.69	0.64	0.54	0.58	0.62

(d) VQAScores on "advanced" prompts

token, we include the EOS (end-of-sentence) token at the end of the text sequence for a simpler implementation. We find that the EOS token only marginally affects the VQAScore results. Next, the generative likelihood of the answer (conditioned on both the question and image) can be naturally factorized as an auto-regressive product [1]:

$$\text{VQAScore}(\mathbf{i}, \mathbf{t}) := P(\mathbf{a}(\mathbf{t})|\mathbf{i}, \mathbf{q}(\mathbf{t})) = \prod_{k=1}^{m} P(a_k|a_{<k}, \mathbf{i}, \mathbf{q}(\mathbf{t})) \tag{4}$$

The answer decoders of VQA models [9, 42] return back m softmax distributions corresponding to the m terms in the above expression. Computing VQAScore is more efficient than generating answer token-by-token. Since the entire sequence of tokens $\{a_k\}$ is already available as input for VQAScore, the above m terms can be efficiently computed in *parallel*. In contrast, answer generation as done by [5, 25] requires *sequential* token-by-token prediction, as token a_k must be generated before it can serve as input to generate the softmax distribution for the subsequent token a_{k+1}.

Pseudocode of VQAScore. To better explain how VQAScore works, we attach the pseudocode in algorithm 1. We will release a pip-installable API to compute VQAScore using one-line of Python code.

E. CLIP-FlanT5

This section describes the training of CLIP-FlanT5.

Algorithm 1: PyTorch-style pseudocode for VQAScore.

```
# tokenize(): text tokenizer that converts texts
 to a list of token indices
# vqa_model(): VQA model returns logits for
 predicted answer

def vqa_score(image, text):
    # Format the text into the below QA pair
    question = f"Does this figure show '{text}'?
     Please answer yes or no."
    answer = "Yes"

    # Tokenize the QA pair into tokens
    question_tokens = tokenize(question)
    answer_tokens = tokenize(answer)

    # Extract logits for predicted answer of shape
     [len(answer_tokens), vocab_size]
    # answer_tokens is a required input for
     auto-regressive decoding
    logits = vqa_model(image, question_tokens,
     answer_tokens)

    # labels must skip the first BOS
     (Begin-Of-Sentence) token
    labels = answer_tokens[1:]
    # logits must skip the last EOS
     (End-Of-Sentence) token
    logits = logits[:-1]

    # Compute the log likelihood of the answer
    log_likelihood =
     -torch.nn.CrossEntropyLoss()(logits, labels)
    # (Optional) Cancel the log to obtain P("Yes"
     | image, question)
    score = log_likelihood.exp()
    return score
```

CLIP-FlanT5. We adhere to the training recipe of the state-of-the-art LLaVA-1.5 [41]. We adopt the same (frozen) CLIP visual encoder (ViT-L-336) [53] and the 2-layer MLP projector for image tokenization. We also follow LLaVA-1.5's two-stage finetuning procedure and datasets. In stage-1 training, we finetune the MLP projector on 558K captioning data (LAION-CC-SBU with BLIP captions [35]). To accommodate FlanT5's encoder-decoder architecture, we adopt the split-text training method proposed in BLIPv2 [35]. This involves splitting a caption into two parts at a random position, with the first part sent to the encoder and the second part to the decoder. In stage-2 training, we finetune both the MLP projector and the language model (FlanT5) on 665K mixture of public VQA datasets (e.g., VQAv2 [20] and GQA [27]). To efficiently train the encoder-decoder architecture, we convert all multi-turn VQA samples into single-turn, resulting in 3.4M image-question-answer pairs. We also retrain LLaVA-1.5 on the same single-turn VQA samples and observe the same VQAScore results. We borrow hyperparameters of LLaVA-1.5 (see Table 7), such as the learning rate schedule, optimizer, number of epochs, and weight decay. We use 8 A100 (80Gbs) GPUs to train all our models. Our largest CLIP-FlanT5-XXL (11B) takes 5 hours for the stage-1 and 80 hours for the stage-2. For stage-2 training, we adhere to the system (prefix) prompt of LLaVA-1.5 during training [1]:

> A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions.
> **USER:** image \n question **ASSISTANT:** answer

Table 7. **Training hyperparameters for CLIP-FlanT5.**

Hyperparameter	**Stage-1**	**Stage-2**
dataset size	558K	665K
batch size	256	96
lr	1e-2	2e-5
lr schedule	cosine decay	
lr warmup ratio	0.03	
weight decay	0	
epoch	1	
optimizer	AdamW	
DeepSpeed stage	2	3

F. Details of Baseline Methods

In this section, we detail the implementation of the baseline methods. Note that Table 8 reports VQAScore performance on seven more benchmarks that measures correlation with human judgments.

CLIPScore and BLIPv2Score. To calculate CLIPScore, we use the same CLIP-L-336 model [22] of CLIP-FlanT5. To calculate BLIPv2Score, we use the ITM head of BLIPv2-vit-G [35]. For an in-depth analysis of how these discriminatively pre-trained VLMs behave as bag-of-words models, we refer readers to previous studies [28, 39, 69, 81].

Metrics finetuned on human feedback (PickScore/ImageReward/HPSv2). We use the official code and model checkpoints to calculate these metrics. Specifically, PickScore [30] and HPSv2 [77] finetune the CLIP-H model, and ImageReward [78] finetunes the BLIPv2 [35], using costly human feedback from either random web users or expert annotators. Our experiments on the Winoground and EqBen benchmarks (Table 8) show that these metrics perform no better than random chance, likely because the discriminative pre-trained VLMs bottleneck their performance due to bag-of-words encodings. In addition, their finetuning datasets may lack compositional texts. Finally, we observe that human annotations can be noisy or subjective, especially when these annotators are not well trained (e.g., random web users of the Pick-a-pic dataset [30]). We discuss these issues in Appendix G.

QG/A methods (VQ2/Davidsonian). We first note that these divide-and-conquer methods are the most popular in

[1] By default, we also use the system prompt during inference. Interestingly, removing the system prompt ("A chat between a curious user ... answers to the user's questions") during inference does not affect CLIP-FlanT5 but will hurt LLaVA-1.5's performance.

Table 8. **VQAScore on image-text alignment benchmarks.** We show Group Score for Winoground and EqBen; AUROC for DrawBench, EditBench, and COCO-T2I; pairwise accuracy [13] for TIFA160 and GenAI-Bench; and binary accuracy for Pick-a-Pick, with higher scores indicating better performance for all metrics. VQAScore (based on CLIP-FlanT5) outperforms all prior art across all benchmarks.

Method	Models	Winoground	EqBen	DrawBench	EditBench	COCO-T2I	TIFA160	Pick-a-Pic	GenAI-Bench
Based on vision-language models									
CLIPScore [22]	CLIP-L-14	7.8	25.0	49.1	60.6	63.7	54.1	76.0	51.9
Finetuned on human feedback									
PickScore [30]	CLIP-H-14 (finetuned)	6.8	23.6	72.3	64.3	61.5	59.4	70.0	57.7
ImageReward [78]	BLIPv2 (finetuned)	12.8	26.4	70.4	70.3	77.0	67.3	75.0	57.4
HPSv2 [77]	CLIP-H-14 (finetuned)	4.0	17.0	63.1	64.1	60.3	55.2	69.0	50.1
QG/A methods									
VQ2 [79]	FlanT5, LLaVA-1.5	10.0	20.0	52.8	52.8	47.7	48.7	73.0	53.3
Davidsonian [5]	ChatGPT, LLaVA-1.5	15.5	20.0	78.8	69.0	76.2	54.3	70.0	45.8
VQAScore (ours) using open-source VQA models									
VQAScore	InstructBLIP	28.5	38.6	82.6	75.7	83.0	70.1	83.0	61.9
VQAScore	LLaVA-1.5	29.8	35.0	82.2	70.6	79.4	66.4	76.0	61.6
VQAScore (ours) using our VQA model									
VQAScore	**CLIP-FlanT5**	**46.0**	**47.9**	**85.3**	**77.0**	**85.0**	**71.2**	**84.0**	**63.1**

recent text-to-visual evaluation [2, 26, 65, 75]. VQ2 [79] uses a finetuned FlanT5 to generate free-form QA pairs and computes the average score of P(answer | image, question). Davidsonian uses a more sophisticated pipeline by prompting ChatGPT to generate yes-or-no QA pairs while avoiding inconsistent questions. For example, given the text "the moon is over the cow", if a VQA model already answers "No" to "Is there a cow?", it then skips the follow-up question "Is the moon over the cow?". However, these methods often generate nonsensical QA pairs, as shown in Table 9 on real-world user prompts from GenAI-Bench.

G. Details of Alignment Benchmarks

This section discusses other benchmarks.

TIFA160 [25]. TIFA160 collects 160 text prompts from four sources: MSCOCO captions [37], DrawBench [58], PartiPrompts [80], and PaintSkill [6]. Each text prompt is paired with five text-to-image models, generating a total of 800 image-text pairs. Furthermore, Davidsonian [5] labels these image-text pairs using 1-5 Likert scale for human evaluation.

Pic-a-pick [30]. We find that the text-to-image evaluation benchmark, Pic-a-pick, contains an excessive amount of NSFW (sexual/violent) content and incorrect labels, likely due to an inadequate automatic filtering procedure. Specifically, after manually reviewing the test set of 500 samples, we find that 10% contain inappropriate content (e.g., "*zentai*" and "*Emma Frost as an alluring college professor wearing a low neckline top*") and approximately 50% had incorrect labels. This may also account for the inferior performance of PickScore. As a result, we manually filter the test set to obtain a clean subset of 100 prompts paired with 200 images for evaluating binary accuracy. We also remove all tied labels due to their subjective nature. We will release this subset of Pick-a-pic for reproducibility.

SeeTrue [79] (DrawBench/EditBench/COCO-T2I). We utilize the binary match-or-not labels collected by SeeTrue [79] for the three benchmarks. These benchmarks consist of individual image-text pairs, where some pairs are correctly paired and others are not. We follow their original evaluation protocols to report the AUROC (Area Under the Receiver Operating Characteristic curve), taking into account all possible classification thresholds.

Winoground [69] and EqBen [73]. In our study, we use the entire Winoground dataset consisting of 400 pairs of image-text pairs. For EqBen, because the official test set includes low-quality images (e.g., very dark or blurry pictures), we analyze the higher-quality EqBen-Mini subset of 280 pairs of image-text pairs, as recommended by their official codebase. These two benchmarks evaluate image-text alignment via matching tasks: each sample becomes 2 image-to-text matching tasks with one image and two candidate captions, and 2 text-to-image matching tasks with one caption and two candidate images. The text (and image) score is awarded 1 point only if *both* matching tasks are correct. The final group score is awarded 1 point only if *all* 4 matching tasks are correct. Importantly, we discover that these benchmarks (especially Winoground) test advanced compositional reasoning skills crucial for understanding real-world prompts, such as counting, comparison, differentiation, and logical reasoning. These advanced compositions operate on basic visual entities, which themselves

Table 9. **Failure cases of divide-and-conquer methods (VQ2/Davidsonian).** We show generated question-and-answer pairs of VQ2 and Davidsonian on three GenAI-Bench prompts. These methods often generate irrelevant or erroneous QA pairs (highlighted in red), especially with more compositional texts.

Method	**Generated questions**	**Candidate answers (correct answer choice in bold)**
Text: "`a snowy landscape with a cabin, but no smoke from the chimney`"		
VQ2	What is the name of the landscape on which it's a cabin?	**a snowy landscape**
	In this landscape what does the fire not go off?	**a cabin**
Davidsonian	Is there a landscape?	**yes**, no
	Is there no smoke from the chimney?	**yes**, no
	Is the cabin in the landscape?	**yes**, no
Text: "`six people wear white shirts and no people wear red shirts`"		
VQ2	What does the average American wear?	**white shirts**
	What kind of clothes do not all people wear?	**red shirts**
Davidsonian	Are there people?	**yes**, no
	Are the shirts red?	**yes**, no
	Are the shirts white?	**yes**, no
Text: "`in the classroom there are two boys standing together, the boy in the red jumper is taller than the boy in the white t-shirt`"		
VQ2	Where do two tall kids stand?	**the classroom**
	Which color of jumper is the tallest?	**the red jumper**
Davidsonian	Is the boy in the red jumper wearing a red jumper?	**yes**, no
	Is the boy in the white t-shirt wearing a white t-shirt?	**yes**, no
	Are the boys standing together?	**yes**, no

can be compositions of objects, attributes, and relations.