

REVISITING TEXT-TO-IMAGE EVALUATION WITH GECKO: ON METRICS, PROMPTS AND HUMAN RATING

Olivia Wiles^{*,†} **Chuhan Zhang^{*,†}** **Isabela Albuquerque^{*,†}** **Ivana Kajic[†]** **Su Wang[†]**
Emanuele Bugliarello[†] **Yasumasa Onoe[†]** **Pinelopi Papalampidi[†]** **Ira Ktena[†]**
Chris Knutson[†] **Cyrus Rashtchian[‡]** **Anant Nawalgaria[§]** **Jordi Pont-Tuset[†]**
Aida Nematzadeh[†]

ABSTRACT

While text-to-image (T2I) generative models have become ubiquitous, they do not necessarily generate images that align with a given prompt. While many metrics and benchmarks have been proposed to evaluate T2I models for alignment, the impact of the evaluation components (prompt sets, human annotations, evaluation task) has not been systematically measured. We find that looking at only *one slice of data*, i.e. one set of skills or human annotations, is not enough to obtain stable conclusions that generalise to new conditions or slices when evaluating T2I models or alignment metrics. We address this by introducing an evaluation suite of >100K annotations across four human annotation templates that comprehensively evaluates models’ capabilities across a range of methods for gathering human annotations and comparing models. In particular, we propose (1) a carefully curated set of prompts – *Gecko2K*; (2) a statistically grounded method of comparing T2I models; and (3) a framework to systematically evaluate metrics under three *evaluation tasks* – *model ordering*, *pair-wise instance scoring*, *point-wise instance scoring*. Using this evaluation suite, we compare a wide range of metrics and find that a given metric may do better in one setting but worse in another. As a result, we introduce a new, interpretable auto-eval metric that is consistently better correlated with human ratings than existing ones on our evaluation suite—across different human templates and evaluation settings—and on TIFA160.

1 INTRODUCTION

Text-to-image (T2I) models (Saharia et al., 2022; Yu et al., 2022b; Betker et al., 2023; Rombach et al., 2022) generate images of impressive quality, but the images are not necessarily aligned with the prompt. The key to comparing T2I models is in the dataset of prompts and human annotations we collect. Human annotation is slow and expensive, motivating the creation (Hu et al., 2023; Cho et al., 2023a) of automatic-evaluation (auto-eval) metrics as a replacement. To evaluate both metrics and models, human annotation is the gold standard. However, Clark et al. (2021) show that the template design and annotator knowledge can significantly impact results in the text domain. In this work, we create a comprehensive benchmark to answer the question: *how do the choices around prompts and human annotation templates impact our metric and modelling decisions?*

There has been limited work analysing the impact on model and metric ranking due to these choices. Previous work builds a benchmark by collecting annotations across *one* template and prompts that cover a limited distribution of skills (see Table 1 for a comparison). A skill refers to a generation

^{*}Equal contribution. Correspondence to: oawiles@google.com; nematzadeh@google.com.

[†]Google DeepMind, [‡]Google Research, [§]Google Cloud.

Github link: https://github.com/google-deepmind/gecko_benchmark_t2i

	Likert	Word Level	DSG(H)	Side-by-Side
Model 1 (M1) 	1-2-3-4-5	A cartoon cat in a professor outfit, writing a book with the title "what if a cat wrote a book."	Q1: Is there a cat? ✓ Q2: Is the cat a cartoon? ✗ Q3: Is the cat in a professor outfit? ✓ Q4: Is the cat writing a book? ✓ Q5: Is the book title "what if a cat wrote a book?" ✗	=
Model 2 (M2) 	1-2-3-4-5	A cartoon cat in a professor outfit, writing a book with the title "what if a cat wrote a book."	Q1: Is there a cat? ✓ Q2: Is the cat a cartoon? ✗ Q3: Is the cat in a professor outfit? ✓ Q4: Is the cat writing a book? ✓ Q5: Is the book title "what if a cat wrote a book?" ✗	=
Model ordering	M1 < M2	M1 > M2	M1 = M2	M1 = M2

Figure 1: **Model ordering outcomes for one annotation template do not necessarily generalise to other templates.** We generate images for two models using the prompt: *A cartoon cat in a professor outfit, writing a book with the title “what if a cat wrote a book.”*. By collecting extensive human evaluation, we expose disparities across templates: outcomes between T2I models or auto-eval metric obtained for one template may not generalise to others.

challenge, such as text rendering or generating different colors and shapes and a sub-skill refers to sub-challenges (e.g. generating longer text or Gibberish). Other work does not systematically gather prompts to ensure a wide coverage of skills and properties with the exception of Zhu et al. (2023). Cho et al. (2023a) does consider different lengths of prompts and Zhu et al. (2023) some skills but this is done for a specific template and does not consider varied challenges for a given skill.

By looking at *one slice of data* (e.g., too few or too specific prompts or one human annotation template), we are at risk of drawing conclusions that are *specific to that slice and do not generalise*. As a result, we collect a comprehensive dataset (Table 1) systematically over different prompt sources and different skills/subskills. Given this prompt set, we generate images from four T2I models and rate them across four human annotation templates. By considering model rankings across annotation templates for a given prompt, we can further determine how reliably a prompt measures alignment.

Similarly, the choice of task may impact our results. Auto-eval metrics are typically evaluated using correlation with human judgement. However, in practice, we would want to use metrics for three tasks: (1) model ordering, ranking T2I models based on *significant* relationships; (2) pair-wise instance scoring, choosing whether a given output is better than another and (3) point-wise instance scoring, an estimation of a samples’ overall alignment. Evaluating metrics on one task is not enough: we may *think* we are choosing the best metric for all three but we show that conclusions for one task *do not necessarily generalise*. An overview of our contributions follows:

- *Gecko*: An evaluation suite for T2I alignment which includes a comprehensive set of 2K prompts, 4 human templates to evaluate 4 T2I models to give $\sim 100K$ human annotations (Table 1). We get predictions from a wide range of auto-eval metrics and evaluate under 3 realistic settings (model ordering, pair-wise instance scoring, point-wise instance scoring).
- Using our suite, we demonstrate limitations of looking at a single slice of data as currently done in the literature: different metrics and models show different results depending on the prompt slice or template.
- Based on our analyses, we introduce an interpretable state-of-the-art QA/VQA metric. It gets the most number of model comparisons right, and performs on average 40.5%/22% better than interpretable baselines on our dataset in terms of pair-wise instance scoring and point-wise instance scoring respectively, and 10.5% better on TIFA160 (Hu et al., 2023).

2 RELATED WORK

Benchmarking alignment in T2I models. Many benchmarks have been proposed to holistically evaluate model capabilities within T2I alignment. Early benchmarks are small scale and created

	Likert	Word Level	DSG(H)	SxS	# prompts annotated	# anns # img	# anns	# Skills (All Categories)	# Sub-Skills
DSG1K (Cho et al., 2023a)	✗	✗	✓	✗	1.06K	3	9.6K	11(13)	✗
DrawBench (Saharia et al., 2022)	✗	✗	✗	✓	200	25	25K	7(11)	✗
PartiP (Yu et al., 2022b)	✗	✗	✗	✓	<u>1.6K</u>	5	16K	9(23)	✗
TIFA160 (Hu et al., 2023)	✓	✗	✗	✗	160	2	1.6K	8(12)	✗
PaintSkills (Cho et al., 2023b)	✓	✗	✗	✗	150	5	2.25K	3(3)	✗
W-T2I (Zhu et al., 2023)	✓	✗	✗	✗	200	3	2.4K	15(20)	9
HEIM (Lee et al., 2024)	✓	✗	✗	✗	708	~5.4	~150K	6(6)	✗
Gecko2K	✓	✓	✓	✓	2K	~13.5	~108K	12(12)	36
Gecko(R)	✓	✓	✓	✓	1K	~13.5	~54K	11(11)	✗
Gecko(S)	✓	✓	✓	✓	1K	~13.5	~54K	12(12)	36

Table 1: **Comparison of annotated alignment datasets.** We report the amount of human annotation and skill division for each dataset. We can see that many datasets include only a handful of annotated prompts or a small number of annotations (anns) per image or overall. No dataset besides Gecko2K collects ratings across multiple different human annotation templates. We also include the number of skills and sub-skills in each dataset. Again, Gecko includes the most number of sub-skills, allowing for a fine-grained evaluation of metrics and models. When datasets do not include skills, we map their categories into skills/sub-skills as appropriate.

alongside model development to perform side-by-side model comparisons ([Saharia et al., 2022](#); [Yu et al., 2022b](#); [Betker et al., 2023](#)). Later work (e.g., TIFA ([Hu et al., 2023](#)), DSG1K ([Cho et al., 2023a](#)) and HEIM ([Lee et al., 2024](#))) focuses on creating holistic benchmarks by drawing from existing datasets (e.g., MSCOCO ([Lin et al., 2014](#)), Localized Narratives ([Pont-Tuset et al., 2020](#)) and CountBench ([Paiss et al., 2023](#))) to evaluate a range of capabilities including counting, spatial relationships, and robustness. Other datasets focus on a specific challenge such as compositionality ([Huang et al., 2024a](#)), contrastive reasoning ([Zhu et al., 2023](#)), text rendering ([Tuo et al., 2023](#)), reasoning ([Cho et al., 2023b](#)), spatial reasoning ([Gokhale et al., 2022](#)), or specifically image ordering given an increasing number of errors ([Saxon et al., 2024](#)). The Gecko2K benchmark is similar in spirit to TIFA and DSG1K in that it evaluates a set of skills. However, in addition to drawing from previous datasets—which may be biased or poorly representative of the challenges of a particular skill—we collate prompts across sub-skills for each skill to obtain a discriminative prompt set. Moreover, we gather human annotations across multiple templates and many prompts (see Table 1).

Automatic metrics measuring T2I alignment. Inspired by work in image captioning, a widely used auto-eval metric is CLIPScore ([Hessel et al., 2021](#)). However, such metrics poorly capture finer-grained aspects of images ([Bugliarello et al., 2023](#); [Yuksekgonul et al., 2022](#)). Motivated by work in NLP on evaluation using entailment or QA metrics ([Maynez et al., 2020](#); [Kryściński et al., 2019](#); [Honovich et al., 2021](#)), similar metrics ([Yarom et al., 2024](#)) have been devised for T2I alignment. However, such a metric may not generalise to new settings and is not interpretable—one cannot diagnose why an alignment score is given. Visual question answering (VQA) methods such as TIFA ([Hu et al., 2023](#)), VQ² [Yarom et al. \(2024\)](#) and DSG ([Cho et al., 2023a](#)) do not require task-specific finetuning and give an interpretable explanation for their score. These metrics create QA pairs which are then scored with a VLM given an image and aggregated into a single score. However, the performance of such methods is conditional on the behaviour of the underlying LLMs used for question generation, and VLMs used for answering questions.

3 Gecko2K: THE GECKO BENCHMARK

We curate a fine-grained skill-based benchmark, Gecko2K, with good coverage by curating two sets of prompts: one created systematically based on a set of skills and subskills (Gecko(S)) and one generated by combining existing datasets but tagging them and resampling to ensure good coverage over those tags (Gecko(R)). We generate Gecko(R) by extending the DSG1K ([Cho et al., 2023a](#)) benchmark creation approach to use automatic tagging and improve the distribution of skills and linguistic properties (see App. B for details on the automatic tagging). However, due to the automatic tagging and nature of the underlying datasets, Gecko(R) is limited in the skills/sub-skills it covers. To generate our systematic set, we propose a hierarchical method combined with LLM generation in order to ensure a systematic distribution across skills (e.g., *counting*) and subskills (e.g., *simple modifier*: ‘1 cat’ vs *additive*: ‘1 cat and 3 dogs’). This notion of sub-skills ensures we are capturing a wide distribution of prompts and not just one easy slice (e.g. generating counts of 1-4 objects).

3.1 GECKO(R): RESAMPLING DAVIDSONIAN SCENE GRAPH BENCHMARK

The recent DSG1K benchmark (Cho et al., 2023a) curates a list of prompts from existing image-text datasets* but does not control for the coverage or complexity of a given skill. The authors randomly sample 100 prompts and limit the prompt length to 200 characters. The resulting dataset is imbalanced in terms of the distribution of skills. Also, as T2I models take in longer and longer prompts, the dataset will not test models on that capability. We take a principled approach in creating Gecko(R) by resampling from the base datasets in DSG1K for better coverage and lifting the length limit. After this process, there are 175 prompts longer than 200 characters and a maximum length of 570 characters. Also, this new dataset has better coverage over a variety of skills than the original DSG1K dataset (see Fig. 6).

While resampling improves the distribution of skills, Gecko(R) has the following shortcomings. Due to the limitations of automatic tagging, it does not include all skills we wish to explore (e.g., language). It also does not include sub-skills: e.g., text rendering prompts do not focus on numerical text, or longer text (see Fig. 7). Finally, automatic tagging can be error prone.

3.2 GECKO(S): A CONTROLLED AND DIAGNOSTIC PROMPT SET

The aim of Gecko(S) is to generate prompts in a controllable manner for skills that are not well represented in previous work. We divide skills into sub-skills to diversify the difficulty and content of prompts. We take inspiration from psychology literature where possible (e.g., colour perception) and known limitations of current T2I models.

Curating a controlled set of prompts with an LLM. To generate a set of prompts semi-automatically, we use an LLM. We first decide on the sub-skills we wish to test for. For example, for text rendering, we may want to test for (1) English vs Gibberish to evaluate the model’s ability to generate uncommon words, and (2) the length of the text to be generated. We then create a template which conditions the generation on these properties. Note that as we can generate as much data as desired, we can define a distribution over the properties and control the number of examples generated for each sub-skill. Finally, we run the LLM and manually validate that the prompts are reasonable, fluent, and match the conditioning variables (e.g., the prompt has the right length and is Gibberish / English). A sample template is given in App. B.3.

Gecko(S) make up. Using this approach and also some manual curation, we focus on twelve skills falling into five categories (Fig. 5): (1) NAMED ENTITIES; (2) TEXT RENDERING; (3) LANGUAGE/LINGUISTIC COMPLEXITY; (4) RELATIONAL: ACTION, SPATIAL, SCALE; (5) ATTRIBUTES: COLOR, COUNT, SURFACES (TEXTURE/MATERIAL), SHAPE, STYLE. For sub-skills, we give a full breakdown for all skills in App. B.4. In Table 2 we give examples of the sub-skills and corresponding prompts for TEXT RENDERING. Using this approach, we get better coverage over the given sub-skills than other datasets (including Gecko(R)) as shown in Fig. 7.

4 COMPARING ANNOTATION TEMPLATES FOR MODELLING

We examine how the choice of human annotation template impacts results when comparing four models: SD1.5 (Rombach et al., 2022), SDXL (Podell et al., 2023), Muse[†] (Chang et al., 2023), and Imagen Vermeer (Vasconcelos et al., 2024). We consider *absolute comparison* templates (i.e., Likert, Word Level from Liang et al. (2023), and DSG(H) from Cho et al. (2023a)) which evaluate models individually, and a template for *relative comparison* of two models (side-by-side or SxS). A

*TIFA(Hu et al., 2023), Stanford Par.(Krause et al., 2017), Localized Narr.(Pont-Tuset et al., 2020), Count-Bench (Paiss et al., 2023), VRD (Lu et al., 2016), DiffusionDB (Wang et al., 2022), MJ (Turc & Nemade, 2023), PoseScript (Delmas et al., 2022), Whoops (Bitton-Guetta et al., 2023), DrawText-Creative (Liu et al., 2022).

[†]Muse is based on the original model, but trained on different data sources.

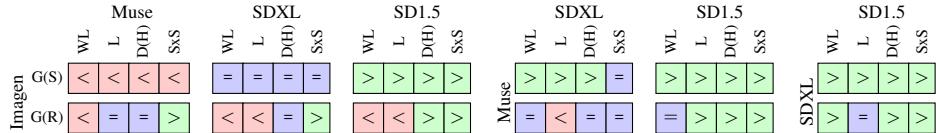


Figure 2: **Comparing models using human annotations.** We compare model rankings on Gecko(S)/(R). Each grid represents a comparison between two models. Entries in the grid depict results for WL, Likert (L), DSG(H) (D(H)), and side-by-side (SxS) scores. The $>$ sign indicates the left-side model is better, worse ($<$), or not significantly different ($=$) than the model on the top.

high-level visualisation of each template is in Fig. 1 and details in App. D.1. We further introduce a principled method to determine significant model orderings based on human judgements.

4.1 DATA QUALITY

We validate the reliability of each template and examine if the choice of the template impacts the quality of the collected data. Given the collected human ratings across the three templates, we compute inter-annotator agreement (IAA) for each generative model by measuring Krippendorff’s α , $\mathcal{K}_\alpha \in [-1, 1]$ (Hayes & Krippendorff, 2007), where a value of 1 indicates perfect agreement and 0 chance (Zapf et al., 2016). Results reported in Table 3 show that agreements are all high, with $\alpha > 0.5$, except for the Likert—SD1.5 pair for Gecko(R); we conjecture this is due to the lower quality images of SD1.5. Overall we find that fine-grained templates (WL and DSG(H)) are more reliable (e.g., have higher IAA) and WL achieves the highest IAA for the diverse Gecko(R). We also measure IAA for the SxS template in Table 11. We see lower IAA for the SxS template (though still far above chance) compared to the fine-grained ones. The IAA is < 0.5 for 6 out of 12 model comparisons. It seems, given the same number of annotators, the SxS template is less reliable.

Reliable prompts. Upon manual investigation, we find that differences in human ratings across templates can arise when prompts are difficult to judge with respect to alignment (and not due to the choice of the template): for example, when a prompt contains domain specific knowledge such as “*A bottle of Irn-Bru is sitting on a shelf*” or subjective notions such as “*a futuristic sculpture*”. To understand how this impacts our results, we consider a subset of the prompts that achieve high IAA across templates and models. For each model and absolute template, we select the prompts for which inter-rater *disagreement*[‡] is $< 50\%$ of the maximum *disagreement* observed across all prompts for that model-template pair. The intersection of these prompts across models and templates gives our *reliable prompts*. We additionally remove instances where all Likert ratings are *Unsure* to get 531 and 725 *reliable prompts* for Gecko(R) and Gecko(S), respectively. We first validate that using reliable prompts increases IAA on the SxS template (which was not used in the selection process) and find that it increases the average \mathcal{K}_α from 0.45 to 0.47 on Gecko(R), and 0.49 to 0.54 on Gecko(S) (see App. D.2 for details). In the next sections, we demonstrate how this subset of prompts increases agreement among templates, but at the expense of removing some potentially meaningful prompts.

4.2 ABSOLUTE ANNOTATION TEMPLATES: COMPARING T2I MODELS

Average ratings. For the absolute annotation templates, previous work compares T2I models by comparing the average ratings across examples. We report these values in Table 3 and find that the chosen prompt set impacts which model is best (e.g. SDXL for Gecko(R) and Muse for Gecko(S)). Moreover, the model with the lowest rating depends on both the prompt set and the template: given Gecko(R), Imagen is worse if using Likert, but SD1.5 is worse if using DSG(H). This highlights the importance of examining models in various conditions. When using T2I models in practice, we need to make conclusions about model ordering with high confidence. We argue that this evaluation is not enough: it does not measure if the difference between models is significant, which is particularly important as models start to saturate. As a result, we introduce the model ordering task.

Model ordering. We verify the significance of outcomes by performing the Wilcoxon signed-rank test with $p < 0.001$. Where results indicate the null-hypothesis is rejected (i.e., the distribution

[‡]Defined as the variance across image, word, and question ratings for Likert, WL and DSG(H) respectively.

Gen. model	Inter annotator agreement						Scores					
	Gecko(R)			Gecko(S)			Gecko(R)			Gecko(S)		
	WL	Likert	DSG(H)	WL	Likert	DSG(H)	WL	Likert	DSG(H)	WL	Likert	DSG(H)
Imagen	0.81	0.64	0.68	0.72	0.57	0.75	0.74 ± 0.30	0.60 ± 0.22	0.84 ± 0.18	0.80 ± 0.24	0.59 ± 0.20	0.78 ± 0.23
Muse	0.82	0.78	0.72	0.69	0.58	0.72	0.84 ± 0.24	0.61 ± 0.25	0.83 ± 0.22	0.88 \pm 0.18	0.63 \pm 0.21	0.84 \pm 0.21
SDXL	0.75	0.76	0.57	0.67	0.56	0.70	0.87 ± 0.19	0.68 \pm 0.22	0.86 \pm 0.16	0.80 ± 0.23	0.60 ± 0.21	0.79 ± 0.22
SD1.5	0.66	0.36	0.66	0.69	0.59	0.74	0.86 ± 0.16	0.67 ± 0.22	0.76 ± 0.23	0.61 ± 0.33	0.49 ± 0.21	0.68 ± 0.27

Table 3: **Inter-annotator agreement and ratings for all models and templates.** We measure inter-annotator agreement for each human evaluation template with Krippendorff’s α . Higher values indicate better agreement. We also show the mean and std. deviation for the annotated judgements of all templates after mapping the ratings to the $[0, 1]$ interval, with 1 indicating perfect alignment.

of ratings is significantly different), we can say that one model is better than another. To determine which model is best, we compare the mean values of their ratings. In Fig. 2 we visualise the outcomes for all model pairs across all templates. We see that Muse is not worse than any of the contenders across all templates and prompt sets, except for 2 out of the 12 comparisons involving Muse for Gecko(R); we determine it is the best overall model. In contrast with the results presented in Table 3, where SDXL is identified as the best model for Gecko(R) across all the templates, we observe that the significance results reveal that Muse and SDXL actually have similar performance, showcasing the importance of determining significance before drawing conclusions.

Reliable prompts. Constraining Gecko(R) using the reliable subset decreases the number of conflicts between the different templates, but at the potential expense of comparing models on fewer, potentially easier, prompts. Considering the two prompt sets, we observe that when using the synthetic prompts, Gecko(S)-rel, all templates agree in Fig. 13 in Appendix D.2. We hypothesise this is because the skills (e.g., color or shape), while hard to generate, are easy to evaluate within generation. For Gecko(R)-rel, we see disagreements between templates, where surprisingly, DSG(H) often result in a different relation than the two other templates. We also consider the full prompt-set, Gecko2K-rel, as it better captures the overall use cases of T2I models: we find that there is always a majority agreement, and the two fine-grained templates (WL and DSG(H)) always agree.

Results by skill. We explore how human judgements vary by skill and template; average ratings for each absolute template are shown in Fig. 23–Fig. 26 in the appendix. A lower average ratings per skill across templates indicates how ‘challenging’ a given skill is: we can see that ‘lang compositional’, ‘lang complexity’, ‘count’ and ‘text’ are consistently difficult across templates.

4.3 RELATIVE ANNOTATION TEMPLATE: COMPARING T2I MODELS

Model ordering. For the SxS template, a model is considered better if it is chosen as preferred more often than the competitor and the *Unsure* rating. To assess statistical significance, we perform a similar procedure as for the absolute annotation templates using binary scores for the ratings: 0 when there was a tie, +1 when a model was preferred by the majority of raters, and -1 otherwise.

We also compare the SxS template with the considered absolute annotation templates by computing the accuracy obtained by each absolute template when predicting the preferred model given by SxS on Gecko2K-rel. Results presented in Table 12 in App.D.2 show that all absolute annotation templates predict SxS judgements with similar average accuracy of around 70%, with DSG being the overall best. This shows that, although the results of pairwise model comparisons are the same in many cases for Gecko2K-rel as shown in Fig. 13, absolute and side-by-side annotations do not necessarily correspond to the same model ordering at the datapoint level.

Takeaway 1: Fine-grained templates (i.e. ones that require multiple annotations per example), WL and DSG(H), yield the highest inter-annotator agreement. **Takeaway 2:** All three absolute annotation templates achieve similar, but not perfect, accuracy when predicting relative comparison annotations for each datapoint. **Takeaway 3:** To compare models reliably, we need to measure the *significant model ordering*. Model ordering depends on the human template and prompt set, but some prompt sets lead to consistent agreement across templates (e.g., are *discriminative*) such as our skill-based Gecko(S) or the larger, reliable set Gecko2k-Rel.

Metrics	Zero-shot	Gecko(R)				Gecko(S)			
		WL	Likert	DSG(H)	SxS	WL	Likert	DSG(H)	SxS
		SpearmanR		Acc	SpearmanR		Acc		
<i>Interpretable (QA/VQA)</i>									
TIFA _{PALM-2/PALI}	✓	0.26	0.34	0.28	41.7	0.39	0.32	0.39	53.2
DSG _{PALM-2/PALI}	✓	0.35	0.47	0.42	49.6	0.45	0.45	0.45	58.1
Gecko _{PALM-2/PALI}	✓	0.41	0.55	0.46	<u>62.1</u>	0.47	0.52	0.45	74.6
Gecko _{Gemini Flash}	✓	0.43	0.58	0.48	72.2	0.54	0.59	0.56	78.8
<i>Uninterpretable (single score)</i>									
CLIP	✓	0.14	0.16	0.13	54.4	0.25	0.18	0.26	67.2
PyramidCLIP	✓	0.26	0.27	0.26	64.3	0.22	0.25	0.23	70.7
VQAScore _{Gemini Flash}	✓	0.42	0.54	0.45	<u>73.1</u>	0.51	0.57	0.49	76.5
VNLI	✗	0.37	0.49	0.42	54.4	0.45	0.55	0.45	72.7

Table 4: **Correlation between auto-eval metrics and human ratings across annotation templates on Gecko2K.** With the same backend, Gecko outperforms all other QA/VQA metrics across all evaluations and Gecko with GeminiFlash performs even better; it performs better or similar to the strongest single-score approach (VQAScore). **Bold:** Top results. Underlined: Top results by category.

5 THE GECKO METRIC

An auto-eval metric is more useful if it is (1) interpretable—it reports where a model fails in addition to its overall goodness, (2) reference-free—does not require a reference distribution, and (3) modular—can easily leverage better pretrained models for improved performance. As a result, we focus on improving recent work using a two-stage QA/VQA metric (Hu et al., 2023; Cho et al., 2023a; Yarom et al., 2024) that matches this criteria (as opposed to metrics such as VNLI (Yarom et al., 2024) and CLIP (Radford et al., 2021)). However, the QA/VQA pipelines are impacted by the shortcomings of the pretrained models used. In particular, we identify two main limitations of these pipelines and address them: the QA generation is not always *grounded* in the prompt as the generated questions might not necessarily cover *all* key parts of the prompt and also there might be *hallucinated* questions that are not related to the prompt. Moreover, at the VQA stage, the highest scoring answer might still be low probably but is treated as the “right” answer—we model this *uncertainty* in the VQA responses. Finally, we simplify the previously proposed methods by removing complexities (such as scene graph generation in DSG) and show that our simplified and improved setup is significantly better across the board.

A standard QA setup (e.g., Hu et al. (2023)) consists of three steps: (1) QA generation: prompting an LLM to generate a set of binary question-answer pairs $\{Q_i, A_i\}_{i=1}^N$ on a given T2I text description T . (2) VQA assessment: employing a VQA model to predict answer $\{A'_i\}_{i=1}^N$ for the generated questions given the generated image I . (3) Scoring: computing the alignment score by assessing the VQA accuracy using Eq. (1):

$$\text{Alignment}(T, I) = \frac{1}{N} \sum_{i=1}^N \mathbb{1}[A'_i = A_i]. \quad (1)$$

Groundedness: increasing coverage. To ensure the coverage of questions over the key elements in a text sentence T , we split the QA generation into two steps. We first prompt the LLM to index the visually groundable words in the sentence. For example, the sentence “A red colored dog.” is transformed into “A {1}{red colored} {2}{dog}.” Subsequently, using the text with annotated keywords $\{W'_i\}_{i=1}^N$ as input, we prompt the LLM again to generate a QA pair $\{q_i, a_i\}$ for each word labelled $\{w'_i\}$ in an iterative manner (see App. C for the prompting details). This two-step process ensures a more comprehensive and controllable QA generation process, particularly for complex or detailed text descriptions where the prompted LLM often selectively generates questions for specific segments of the text while overlooking others.

Groundedness: removing hallucination. LLMs can hallucinate (Bang et al., 2023; Guerreiro et al., 2023), leading to the generation of low-quality, unreliable QA pairs. We filter out hallucinated QA pairs by taking inspiration from previous work in NLP (Maynez et al., 2020; Kryściński et al., 2019): we employ a Natural Language Inference (NLI) model (Honovich et al., 2022) model for measuring the factual consistency between the text T and QA pairs $\{Q_i, A_i\}$. QA pairs with a consistency score lower than a threshold r are removed, ensuring that the remaining QAs are about the prompt.

Uncertainty: VQA score normalisation. Finally, we improve aggregation of scores from the VQA model. The reliance on binary judgement—strictly matching A'_i and A_i without considering the

Metrics	Gecko(R)			Gecko(S)		
	WL	Likert	DSG(H)	WL	Likert	DSG(H)
TIFA baseline	0.21	0.32	0.25	0.39	0.32	0.39
+ coverage	0.28	0.34	0.32	0.41	0.33	0.40
+ VQA score norm	0.32	0.42	0.37	0.43	0.37	0.41
+ NLI filtering	0.38	0.51	0.42	0.46	0.48	0.46

Table 5: **Validation of each component of the proposed Gecko metric on Gecko2K.**

We evaluate the utility of the three proposed improvements by adding them to the TIFA baseline one by one. They all bring higher correlation with human judgement across the board on Gecko2K.

predicted probability of A'_i —overlooks the inherent uncertainty in the predictions; a VQA model can predict a very similar score for two answers. If we simply take the max, then we lose this notion of uncertainty reflected in the scores. As a result, we normalise the scores as follows,

$$\text{Alignment}(T, I) = \frac{1}{N} \sum_{i=1}^N \frac{s_a}{\sum_i s_i}, \quad (2)$$

where the negative log likelihood of answer A'_i is s_i and the correct answer is A'_a with score s_a .

6 EXPERIMENTS ON AUTO-EVAL METRICS

We evaluate metrics across multiple prompt sets and templates to determine how they fare on the three tasks: (1) Do they give a good numeric measurement of overall alignment – **point-wise instance scoring**; (2) Are they good indicators on side by side comparisons – **pair-wise instance scoring**; (3) Can they predict **model ordering**. We demonstrate that the task *can* impact rankings but that our Gecko metric consistently performs best for Gecko(S)/(R) across tasks and on TIFA160. App. F.1 gives a thorough description of each task and intuitive examples for how they differ.

6.1 EXPERIMENTAL SETUP

Metrics. We benchmark two types of metrics. First, metrics that give a *single score*, including contrastive models (CLIP (Radford et al., 2021), PyramidCLIP (Gao et al., 2022) and 16 variants in Sec. F.4); (2) VNLI (Yarom et al., 2024); and (3) VQAScore (Lin et al., 2024). Second, interpretable QA/VQA based methods: TIFA (Hu et al., 2023), DSG (Cho et al., 2023a) and our metric Gecko.

Back-end models. For CLIP, we use a ViT-B/32 (Dosovitskiy et al., 2020) CLIP model and ViT-B/16 (Dosovitskiy et al., 2020) PyramidCLIP model. For VQAScore, we use a GeminiFlash (Reid et al., 2024) backend. For all the VQA-based metrics, we use PaLM-2 (Anil et al., 2023) as the LLM and PaLI (Chen et al., 2022) as the VQA models in all the metrics for fair comparison. When evaluating the Gecko metric, apart from using the LLM and VQA models above, we utilise a T5-11B model from Honovich et al. (2022) for NLI filtering and set the threshold r at 0.005. This threshold was determined by examining QA pairs with NLI probability scores below 0.05. We observed that QAs with scores below 0.005 are typically hallucinations. We re-use the original prompts from TIFA for generating QAs, and add coverage notation to their selected texts as described in Sec. 5. We additionally explore how the performance of point-wise instance scoring changes for the Gecko metric if we swap out the QA/VQA models for a stronger Gemini Flash model. Finally, some baseline models are trained with a maximum text input length L , e.g. $L_{\text{CLIP}} = 77$ and $L_{\text{VNLI}} = 82$. For these models, we only take the first L tokens from the text as input.

6.2 COMPARING AUTO-EVAL METRICS ON POINT-WISE INSTANCE SCORING

We first evaluate how well metrics measure T2I alignment at an instance level. We compute the Pearson and Spearman Ranked correlation between the auto-eval scores and human scores on all the instances in a prompt set. The evaluations are done on the Gecko benchmark and TIFA160.

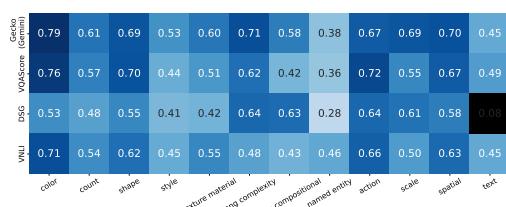


Figure 3: **Per skill results of different metrics.** Likert correlation for each skill; square black indicates p -values > 0.05 . Full results in App. F.

Metrics	QA	VQA	Spearman’s ρ	Kendall’s τ
ROUGE-L			0.33	0.25
METEOR	N/A	N/A	0.34	0.27
SPICE			0.33	0.23
CLIP			0.33	0.23
TIFA	GPT-3	BLIP-2	0.56	0.44
	GPT-3	MPLUG	0.60	0.47
	PALM	PaLI	0.43	0.32
DSG	PALM	PaLI	0.57	0.46
Gecko	PALM	PaLI	0.64	0.50

Table 6: Comparing different metrics by their correlation with human Likert ratings on TIFA160. The Gecko metric outperforms the others by a significant margin.

Metrics	WL Pearson	Likert	SxS Acc
VideoCLIP	0.18	0.21	28.0
VQAScore _{Gemini Flash}	0.30	0.33	52.0
Gecko _{Gemini Flash}	0.43	0.45	55.8

Table 7: Correlation between auto-eval metrics and human ratings for text-to-video evaluations. Gecko outperforms other auto-eval metrics on VBench overall consistency prompts, demonstrating the generality of the approach to other modalities.

Component validation on proposed Gecko metric. We validate the utility of the three key improvements we proposed: coverage, linear normalisation, and NLI filtering. Starting from our baseline TIFA, we include the improvements one at a time. The results in Table 5 uniformly demonstrate a positive impact. NLI filtering brings the largest boost among the three, underscoring the limitation of the PaLM-2 LLM in reliably generating high-quality and accurate QA pairs.

Results on Gecko benchmark We next compare auto-eval metrics. We start with metrics (CLIP and its variants, TIFA, DSG, Gecko) that do not rely on fine-tuning. As shown in Table 4, the Gecko metric outperforms other QA/VQA metrics using the same backend by a wide margin. Swapping out the backend of Gecko with a stronger GeminiFlash model leads to large improvements across the board. Contrastive models (e.g. CLIP variants) are worse than QA-based metrics, but VQAScore is a strong baseline. Finally, we compare Gecko with VNLI, our supervised baseline, as it is fine-tuned for text–image alignment on a mixed dataset containing COCO (which is used in Gecko(R)), while other metrics are zero-shot. It is worth noting that the correlation scores of different auto-eval metrics are generally higher on Gecko(S) than on Gecko(R). This validates that our skills-based benchmark has a more objective and balanced measure of alignment. We observe similar conclusions on the Gecko2K Reliable Prompts (Gecko2K-Rel) subset; results are in App. F.3.

TIFA160 results. We compare the Gecko metric with other metrics on TIFA160 (Hu et al., 2023), a set of 160 text–image pairs, each annotated with two Likert ratings. In Table 6, we list the results reported in Hu et al. (2023) and Cho et al. (2023a), and compare them with Gecko as well as our re-implementation of TIFA / DSG. Gecko has the highest correlation, with an average correlation 0.07 higher than that of DSG, when using the same QA and VQA models. This shows that the power of our proposed metric is from the method itself, not from the advance of models used.

Skill-based evaluation with Gecko. To better understand the differences between auto-eval metrics/annotation templates with respect to various skills, we visualise a breakdown of skills in Gecko(S) in Fig. 3 and App. E.1, F.5. The metrics have different strengths: e.g., we see that while Gecko, VQAScore, VNLI metrics are consistently good across skills, the Gecko metric is better on more complex and compositional language, DSG is best on compositional prompts, and VNLI is better on named entities. As with the overall results, these per skill conclusions seem to hold across templates.

Qualitative examples. We visualise examples in Fig. 4. For the negation example, the reason DSG(H) gives inconsistent results with WL/Likert here is that the question generation is confused by the negation (asking if there *are* cars as opposed to *no* cars). We can also see that VNLI and DSG mistakenly think none of the images are aligned. VNLI and DSG perform better on the shape prompt but VNLI scores Imagen incorrectly and DSG gives hard scores per question (0 or 1) and so it is sometimes not able to capture subtler differences in the human ratings.

6.3 COMPARING AUTO-EVAL METRICS ON PAIR-WISE INSTANCE SCORING

We measure how well an auto-eval metric is able to select between two generations given a prompt. We compare metrics’ predictions with the human choices we collected by computing accuracy—the percentage of times the metric gets the comparison right. Results are in the SxS column in Table 4. Although Gecko was the clear winner on point-wise instance scoring, single-score metrics are generally very good at SxS comparison. PyramidCLIP was worse than TIFA and DSG on point-wise instance scoring, but it has a much higher SxS accuracy, showing that different human annotation templates *do not* always give the same result, and single-score metrics can be a good estimator on the pair-wise instance scoring task. While VQAScore is better than the Gecko metric on SxS com-

Skill (subskill): Prompt:		lang/complexity (negation) A bridge with no cars on it.				Shape: (hierarchical) The number 0 made of smaller circles			
		Imagen	Muse	SDXL	SD1.5	Imagen	Muse	SDXL	SD1.5
WL: Likert: DSG(H):	0.5	1. 1. 0.5	1. 1. 0.5	1. 1. 0.5	1. 0.87 0.67	1. 0.92	1. 0.87 1.	0. 0.2 0.	0.67 0.67 0.89
Gecko: DSG: VNLI:	0.96 0.25 0.4	0.94 0.25 0.42	0.93 0.25 0.32	0.91 0.25 0.30	0.9 0.95 0.36	0.9 0.79	0.95 0.79 0.24	0.55 0. 0.24	0.75 0. 0.32

Figure 4: **Qualitative results.** Image generations of the four T2I models on prompts in Gecko(S), with the human annotation ratings and auto-eval scores.

parison on Gecko(S), Gecko is better on Gecko(R) and the Gecko metric is the only interpretable metric that has better or comparable performance with single-score metrics on SxS comparisons.

6.4 COMPARING AUTO-EVAL METRICS ON MODEL ORDERING

A good auto-eval metric should be able to give an overall model ordering for a set of prompts. To decide on a ground-truth ordering, we use Gecko2K-rel as it is the largest subset that has highest agreement across templates. We take the majority vote relationship in Fig. 13 as the ground truth. We compare these results to the significant relationships found using the auto-eval metrics in App. F.2 (we only use PaLM/PaLI-2 backends if there is a choice). We find that CLIP performs poorly, confusing wins with losses. All other auto-eval metrics perform well, never confusing a win with a loss but sometimes not finding significant relations when there is one or vice versa. Gecko correctly finds and predicts *all* significant relations, unlike the other metrics.

6.5 EXTENDING GECKO TO OTHER MODALITIES

To explore the generality of our approach on different modalities, we validate it on text-to-video generation. We choose a prompt set from VBench (Huang et al., 2024b) and compare the following models: Lumiere (Bar-Tal et al., 2024), Phenaki (Villegas et al., 2022) and WALT (Gupta et al., 2023). For human evaluation, we consider absolute (i.e., Likert, Word Level) and side-by-side templates. For automatic evaluation, we benchmark contrastive models (i.e., VideoCLIP; Xu et al. 2021) and VQA-based metrics. For VQA-based metrics, we extend the VQAScore and our fine-grained Gecko metric on videos using Gemini Flash, which can process long context multimodal inputs. We present the results in Table 7 and find that the Gecko metric agrees more closely with human judgement across all human templates than other metrics. See Appendix G for more details.

Takeaway: Although Gecko is the best metric on different human templates and modalities, we find that the ranking of different auto-eval metrics can change depending on whether they are evaluated on an instance-level template (e.g., Likert or DSG(H)), a comparative template (e.g. SxS) or for model ordering. It is important to evaluate metrics across a range of settings and in particular on one relative and one absolute template if under budget constraints.

7 CONCLUSIONS

We introduce the Gecko evaluation suite, a comprehensive set of prompts, human ratings across templates, and tasks to evaluate T2I models and alignment metrics. We find that looking at a single slice of the data (e.g., one annotation template, or one evaluation task) can give misleading observations of the relative benefits of one model or metric. Instead, we show that we need to use a comprehensive prompt set (or manually evaluated “reliable prompts”) to achieve consistent model orderings and thereby confidence in model rankings. Given this evaluation suite, we demonstrate that our Gecko metric performs consistently best across three tasks, measuring how metrics perform in scoring each image–text instance with respect to their alignment as well as ranking models. Our work highlights the importance of standardising the evaluation framework with respect to the prompt sets, the annotation templates, and metrics used. This is crucial when conducting research on models and metrics, and also to make informed decisions.

8 ETHICS STATEMENT

When gathering our dataset, we ensure that raters are compensated and provide consent as described in App. D.1.1. We also run safety filters over the generated images before giving them to the raters. This work is a step towards better evaluation of text-to-image models which are known to hallucinate. It gives tools to others developers and practitioners to properly understand and evaluate T2I models in the future.

9 REPRODUCIBILITY STATEMENT

We give extensive details of our setup in the Appendix. For human annotation, we visualise the templates used and give extensive detail on how these raw ratings are aggregated in App. D.1. For the dataset collation, we give the few shot prompts used to generate tags and templates: the few shot prompt for Gecko(R) is given in Listing 1. For Gecko(S), we give our full decomposition of skills in Table 8 with examples and an explanation of how we generated prompts for each specific skill in App. B.4 with sample few shot prompts. For metrics, we give full details of the baselines in Sec. 6.1 and the additional CLIP baselines in Sec. F.4. For Gecko metrics, we give the few shot prompt for generating coverage in Listing 3 and for generating the QAs in Listing 4.

Acknowledgements We thank Zi Wang, Miloš Stanojević, and Jason Baldridge for their feedback throughout the project. We are grateful to Andrew Zisserman for his feedback on the manuscript. We thank Aayush Upadhyay and the rest of the Podium team for their help in running models.

REFERENCES

- Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don’t just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4971–4980, 2018. [26](#)
- Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. [8](#)
- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*, 2023. [7](#)
- Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Yuanzhen Li, Tomer Michaeli, et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024. [10, 45](#)
- James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science*. <https://cdn.openai.com/papers/dall-e-3.pdf>, 2(3):8, 2023. [1, 3](#)
- Nitzan Bitton-Guetta, Yonatan Bitton, Jack Hessel, Ludwig Schmidt, Yuval Elovici, Gabriel Stanovsky, and Roy Schwartz. Breaking common sense: Whoops! a vision-and-language benchmark of synthetic and compositional images. In *ICCV*, 2023. [4](#)
- Emanuele Bugliarello, Laurent Sartran, Aishwarya Agrawal, Lisa Anne Hendricks, and Aida Ne-matzadeh. Measuring progress in fine-grained vision-and-language understanding. *arXiv preprint arXiv:2305.07558*, 2023. [3](#)
- Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Computer vision and pattern recognition (CVPR), 2018 IEEE conference on*. IEEE, 2018. [24](#)
- Huiwen Chang, Han Zhang, Jarred Barber, Aaron Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Patrick Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 4055–4075. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/chang23b.html>. [4](#)
- Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme, Andreas Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. [8](#)
- Xinyi Chen, Raquel Fernández, and Sandro Pezzelle. The bla benchmark: Investigating basic language abilities of pre-trained multimodal models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5817–5830, 2023. [24](#)
- Jaemin Cho, Yushi Hu, Roopal Garg, Peter Anderson, Ranjay Krishna, Jason Baldridge, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-image generation. *arXiv preprint arXiv:2310.18235*, 2023a. [1, 2, 3, 4, 7, 8, 9, 17, 18, 19, 26, 27, 40](#)
- Jaemin Cho, Abhay Zala, and Mohit Bansal. Dall-eval: Probing the reasoning skills and social biases of text-to-image generation models. In *ICCV*, 2023b. [3](#)

- Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. All that's 'human' is not gold: Evaluating human evaluation of generated text. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021. 1
- Dean C. Delis, Lynn C. Robertson, and Robert Efron. Hemispheric specialization of memory for visual hierarchical stimuli. *Neuropsychologia*, 24(2):205–214, 1986. ISSN 0028-3932. 23
- Ginger Delmas, Philippe Weinzaepfel, Thomas Lucas, Francesc Moreno-Noguer, and Grégory Rogez. Posescript: 3d human poses from natural language. In *ECCV*, 2022. 4
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 8
- Yuting Gao, Jinfeng Liu, Zihan Xu, Jun Zhang, Ke Li, Rongrong Ji, and Chunhua Shen. Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. In *NeurIPS*, 2022. 8, 40
- Robert Geirhos, Kristof Meding, and Felix A Wichmann. Beyond accuracy: quantifying trial-by-trial behaviour of cnns and humans by measuring error consistency. *Advances in Neural Information Processing Systems*, 33:13890–13902, 2020. 43
- Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *TACL*, 9: 346–361, 2021. 23
- Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022. 3
- Nuno M Guerreiro, Duarte M Alves, Jonas Waldendorf, Barry Haddow, Alexandra Birch, Pierre Colombo, and André FT Martins. Hallucinations in large multilingual translation models. *TACL*, 11:1500–1517, 2023. 7
- Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 24
- Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. *arXiv preprint arXiv:2312.06662*, 2023. 10, 45
- Andrew F Hayes and Klaus Krippendorff. Answering the call for a standard reliability measure for coding data. *Communication methods and measures*, 1(1):77–89, 2007. 5
- Cindy Yoonjoung Heo, Bona Kim, Kwangsoo Park, and Robin M Back. A comparison of best-worst scaling and likert scale methods on peer-to-peer accommodation attributes. *Journal of business research*, 148:368–377, 2022. 27
- Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *EMNLP*, pp. 7514–7528, 2021. 3
- Matthew Honnibal and Ines Montani. Spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *spacy.io*, 2017. 24
- Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. Q2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. *arXiv preprint arXiv:2104.08202*, 2021. 3

- Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. True: Re-evaluating factual consistency evaluation. *arXiv preprint arXiv:2204.04991*, 2022. 7, 8
- 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. *arXiv preprint arXiv:2303.11897*, 2023. 1, 2, 3, 4, 7, 8, 9, 40
- Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *NeurIPS*, 36, 2024a. 3
- Ziqi Huang, Yinan He, Jiahuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024b. 10, 45
- Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL <https://doi.org/10.5281/zenodo.5143773>. If you use this software, please cite it as below. 40
- Vasily Kostumov, Bulat Nutfullin, Oleg Pilipenko, and Eugene Ilyushin. Uncertainty-aware evaluation for vision-language models. *arXiv preprint arXiv:2402.14418*, 2024. 26
- Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *CVPR*, 2017. 4
- Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840*, 2019. 3, 7
- Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Teufel, Marco Bellagente, et al. Holistic evaluation of text-to-image models. *NeurIPS*, 36, 2024. 3
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML’23. JMLR.org, 2023. 40
- Weixin Liang, James Zou, and Zhou Yu. Beyond user self-reported likert scale ratings: A comparison model for automatic dialog evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1363–1374, 2020. 27
- Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, Feng Yang, et al. Rich human feedback for text-to-image generation. *arXiv preprint arXiv:2312.10240*, 2023. 4, 27
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 3
- Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *arXiv preprint arXiv:2404.01291*, 2024. 8
- Rosanne Liu, Dan Garrette, Chitwan Saharia, William Chan, Adam Roberts, Sharan Narang, Irina Blok, RJ Mical, Mohammad Norouzi, and Noah Constant. Character-aware models improve visual text rendering. *arXiv preprint arXiv:2212.10562*, 2022. 4
- Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22139–22149, 2024. 45
- Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016. 4

- Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*, 2020. 3, 7
- Shinri Ohta, Naoki Fukui, and Kuniyoshi L. Sakai. Computational principles of syntax in the regions specialized for language: integrating theoretical linguistics and functional neuroimaging. *Frontiers in Behavioral Neuroscience*, 2013. 24
- Roni Paiss, Ariel Ephrat, Omer Tov, Shiran Zada, Inbar Mosseri, Michal Irani, and Tali Dekel. Teaching clip to count to ten. *arXiv preprint arXiv:2302.12066*, 2023. 3, 4
- Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 4
- Jordi Pont-Tuset, Jasper Uijlings, Soravit Changpinyo, Radu Soricut, and Vittorio Ferrari. Connecting vision and language with localized narratives. In *ECCV*, 2020. 3, 4
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 7, 8
- Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittweiser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. 8
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 4
- Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 35:36479–36494, 2022. 1, 3, 19
- Michael Saxon, Fatima Jahara, Mahsa Khoshnoodi, Yujie Lu, Aditya Sharma, and William Yang Wang. Who evaluates the evaluations? objectively scoring text-to-image prompt coherence metrics with t2iscorescore (ts2). *arXiv preprint arXiv:2404.04251*, 2024. 3
- Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Workshop on Vision and Language (VL15)*, Lisbon, Portugal, September 2015. Association for Computational Linguistics. 24
- Robyn Speer. Python library: rspeer/wordfreq: v3.0, September 2022. URL <https://doi.org/10.5281/zenodo.7199437>. 20
- J Ridley Stroop. Studies of interference in serial verbal reactions. *Journal of experimental psychology*, 18(6):643, 1935. 23
- Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 40
- Yuxiang Tuo, Wangmeng Xiang, Jun-Yan He, Yifeng Geng, and Xuansong Xie. Anytext: Multilingual visual text generation and editing. *arXiv preprint arXiv:2311.03054*, 2023. 3
- Iulia Turc and Gaurav Nemade. Midjourney user prompts & generated images (250k), 2023. 4
- Cristina N. Vasconcelos, Abdullah Rashwan, Austin Waters, Trevor Walker, Keyang Xu, Jimmy Yan, Rui Qian, Shixin Luo, Zarana Parekh, Andrew Bunner, Hongliang Fei, Roopal Garg, Mandy Guo, Ivana Kajic, Yeqing Li, Henna Nandwani, Jordi Pont-Tuset, Yasumasa Onoe, Sarah Rosston, Su Wang, Wenlei Zhou, Kevin Swersky, David J. Fleet, Jason M. Baldridge, and Oliver Wang. Greedy growing enables high-resolution pixel-based diffusion models, 2024. URL <https://arxiv.org/abs/2405.16759>. 4

- Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *International Conference on Learning Representations*, 2022. 10, 45
- Zijie J Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models. *arXiv preprint arXiv:2210.14896*, 2022. 4
- Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2575–2584, 2020. 23
- Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6787–6800, 2021. 10, 45
- Michal Yarom, Yonatan Bitton, Soravit Changpinyo, Roee Aharoni, Jonathan Herzig, Oran Lang, Eran Ofek, and Idan Szpektor. What you see is what you read? improving text-image alignment evaluation. *NeurIPS*, 36, 2024. 3, 7, 8, 40
- Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *Transactions on Machine Learning Research*, 2022a. ISSN 2835-8856. URL <https://openreview.net/forum?id=Ee277P3AYC>. 40
- Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022b. 1, 3, 19
- Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *ICLR*, 2022. 3
- Antonia Zapf, Stefanie Castell, Lars Morawietz, and André Karch. Measuring inter-rater reliability for nominal data—which coefficients and confidence intervals are appropriate? *BMC medical research methodology*, 16:1–10, 2016. 5
- Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25994–26009. PMLR, 17–23 Jul 2022. URL <https://proceedings.mlr.press/v162/zeng22c.html>. 40
- Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *arXiv preprint arXiv:2303.15343*, 2023. 40
- Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 24
- Xiangru Zhu, Penglei Sun, Chengyu Wang, Jingping Liu, Zhixu Li, Yanghua Xiao, and Jun Huang. A contrastive compositional benchmark for text-to-image synthesis: A study with unified text-to-image fidelity metrics. *arXiv preprint arXiv:2312.02338*, 2023. 2, 3

APPENDICES

A OVERVIEW

In the Appendix, we give additional information on the benchmark, human annotation and corresponding results for T2I models, and experimental results for the auto-eval metrics.

Gecko Benchmark: For the benchmark, we give further information on how we automatically tag Gecko(R) and semi-automatically generate prompts in Gecko(S) in App. B.1 and B.3 respectively. We then give more detail about the skill breakdown in Gecko(R) in App. B.2. We define and give examples for the sub-skills in Gecko(S) in App. B.4.

Gecko Metric: We give further details on the Gecko metric in App. C.

Human Annotation: For the human annotation, we give additional details of our setup including screenshots of the annotation templates used and qualitative limitations of each setup in App. D.1. We further discuss more experimental results comparing inter-annotator agreement and the raw predictions under each template in App. D.2. Finally, we visualise the most and least reliable prompts in App. D.3, giving an intuition for the properties of the prompt that lead to more or less agreement across templates.

Additional results on T2I models: We give further results on using the annotated data to (1) compare T2I models by skill in App. E.1. We also compare how well prompts in TIFA160 are able to discriminate models under our human annotation setup in App. E.2 and find that they are less discriminative.

Additional results for auto-eval metrics: We give an intuitive explanation of each task as well as how they can lead to different metric orderings in App. F.1. We then give additional results for the auto-eval metrics on Gecko2K and Gecko2K-rel, including more correlation results in App. F.3 but we find that conclusions are the same irrespective of how we compute correlation or using the reliable subset or full set. We give the raw results for the model-ordering evaluation in App. F.2 and results for different CLIP variants in App. F.4. Finally, we explore results per skill for different auto-eval metrics in App. F.5, give additional visualisations in App. F.6 and demonstrate that we can use Gecko to evaluate the per-word accuracy of the metric (this is not possible with other auto-eval metrics) in App. F.7.

B GECKO2K: MORE DETAILS

As described in Sec. 3.1, we use automatic tagging in order to tag prompts with different skills in Gecko(R). However, this has a few issues: (1) it can be error prone; (2) we are limited by the tagging mechanism in the skills that we tag; (3) we do not tag sub-skills. As a result, we devise a semi-automatic approach to build Gecko(S) by few-shot prompting an LLM, as discussed in Sec. 3.2 and curate a dataset with a number of skills and sub-skills for each skill. This dataset covers more skills and sub-skills than other datasets, as shown in Fig. 6, 7.

B.1 AUTOMATIC TAGGING FOR GECKO(R)

As mentioned in Sec. 3.1, to obtain a better control for the skill coverage and prompt length, we resampled from the 10 datasets used in DSG1k (Cho et al., 2023a). To identify the categories covered in the prompts, we adopted an automatic tagging method similar to that used in DSG1K. This method utilizes a Language Model (LLM) to tag words in the text prompt, as shown in Listing 1. The only difference is that we also included named entities and landmarks to be the original categories, such as *whole*, *part*, *state*, *color* etc.

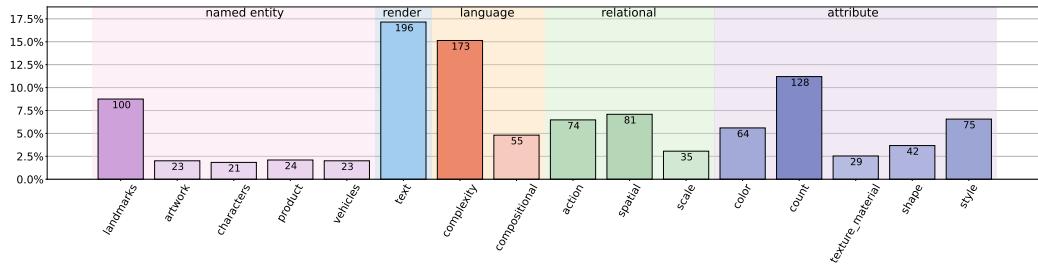


Figure 5: **Overview of Gecko(S).** The set of skills (coloured by the corresponding category) covered by the synthetic prompts. Note that we gather prompts by breaking each skill into sub-skills.

```

1 """
2
3 id: synthetic_v1_1
4 input: a man is holding an iPhone.
5 output: 1 | entity - whole (man)
6     2 | entity - named entity (iPhone)
7     3 | action - hold (man, iPhone)
8
9 id: diffusiondb_79
10 input: an hd painting by Vincent van Gogh. a bunch of zombified mallgoths hanging out at a hot topic store in
11 the mall.
12 output: 1 | global - style (hd painting)
13     2 | global - style (Vincent van Gogh)
14     3 | entity - whole (mallgoths)
15     4 | attribute - state (mallgoths, zombified)
16     5 | other - count (mallgoths, ==bunch)
17     6 | entity - whole (hot topic store)
18     7 | entity - whole (mall)
19     8 | relation - spatial (mallgoths, hot topic store, at)
20     9 | relation - spatial (hot topic store, mall, in)
21 ...
22 id: {image_id}
23 input: {text_input}
24 output: {LLM_output}
25 """

```

Listing 1: The prompt used to automatically tag skills given text prompts from the base datasets in DSG1K in order to generate a more balanced Gecko(R).

B.2 PROMPT DISTRIBUTION IN GECKO(R)

We resample 1000 prompts from the base datasets used in DSG1K and ensure a more uniform distribution over skills and prompt length. To sample more prompts featuring words from under-represented skills (e.g. TEXT RENDERING, SHAPE, NAMED IDENTITY and LANDMARKS), we use automatic tagging in App. B.1 to categorize the words in all prompts as pertaining to a given skill. We then resample, assigning higher weights to the under-represented skills. The resulting skill distribution is shown in Fig. 8. Although the resampling increases the proportion of under-represented skills, the overall distribution remains unbalanced. This underscores the necessity of acquiring a synthetic subset with a more controlled and balanced prompt distribution. To sample long prompts, we eliminate the constraint set in DSG1K (Cho et al., 2023a), which mandates that the sampled prompts should be shorter than 200 characters. This adjustment results in a more diverse prompt length distribution as shown in Fig. 8.

B.3 TEMPLATES TO FEW-SHOT AN LLM FOR GECKO(S)

As discussed in Sec. 3.2, we semi-automatically create prompts for Gecko(S) by few-shot prompting an LLM. We give an example for the TEXT RENDERING skill in Listing 2. In short, we define a set of properties based on the sub-skills we want included in our dataset. In this case, we define *text length* and *language* (we use *English* and *Gibberish* but we note this could be easily extended to more languages). We then create examples that have those properties to create our few-shot prompt. We can query the LLM as many times as we like to create a distribution of prompts across different text lengths and languages. We do a similar setup for each of the skills and sub-skills we define below.

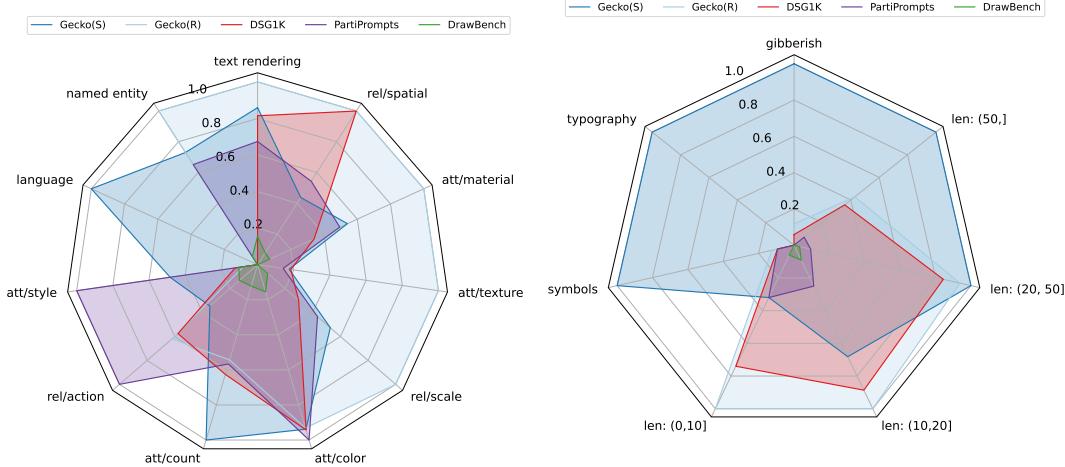
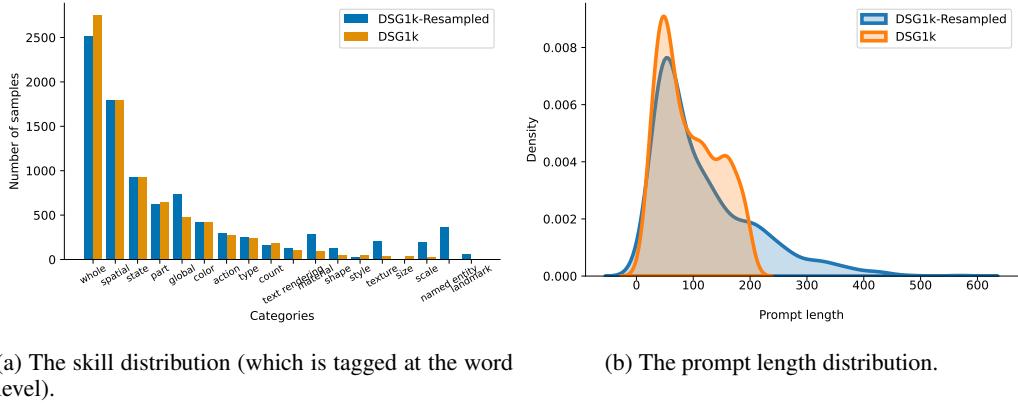


Figure 6: Distribution of skills. We visualise the distribution of prompts across different skills for Gecko(S)/(R), DSG1K (Cho et al., 2023a), PartiPrompts (Yu et al., 2022b) and DrawBench (Saharia et al., 2022). We use automatic tagging and, for each skill, normalise by the maximum number of prompts in that skill over all datasets. For most skills, Gecko2K has the most number of prompts within that skill.

Figure 7: TEXT RENDERING skill. We visualise the distribution of prompts across seven sub-skills explained in Table 2 ('len: ...' corresponds to bucketing different lengths of the text to be rendered). We normalise by the maximum number of prompts in the sub-skill (note that we only count unique texts to be rendered). The Gecko(S) dataset fills in much more of the distribution here than other datasets.



(a) The skill distribution (which is tagged at the word level).

(b) The prompt length distribution.

Figure 8: Prompt distribution in DSG1K-Resampled(Gecko-R) and DSG1K.

```

1 """
2 Generate captions for the given text of varying length. Be creative and imagine new settings.
3
4 Text length: 20
5 Language: English
6 Text: "look at that shadow!"
7 Caption: shadow of a stone, taken from the point of view of an ant, with the caption "look at that shadow!"
8
9 ...
10
11 Text length: {text_length}
12 Language: {language}
13 Text: {LLM_output}
14 Caption: {LLM_output}
15 """

```

Listing 2: Sample LLM template.

B.4 BREAKDOWN BY SKILL/SUB-SKILL IN GECKO(S)

An overview of Gecko(S) is given in Fig. 5 and comparisons to other datasets for skills and a given subskill in Fig. 6,7. In this section we give more information on the skills and sub-skills within Gecko(S). We provide a detailed breakdown of each prompt sub-skill, including examples and justifications. Skills and sub-skills are listed in Table 8. We aim to cover semantic skills, some of which have already been covered in previous work (e.g.*shapes*, *colors* or *counts*), while further subdividing each skill to capture its different aspects and difficulty levels. By varying the difficulty of the prompts within a challenge we ensure we are testing the models and metrics at different difficulty levels and can find where models and metrics begin to break.

Each skill (such as **SHAPE**, **COLOR**, or **NUMERICAL**) is divided into sub-skills, so that prompts within that sub-skill can be distinguished based on difficulty or, if applicable, some other criteria that is unique to that sub-skill (i.e.prompts inspired by literature in psychology). We create a larger number of examples and subsample to create our final 1K set of prompts to be labelled.

B.4.1 SPATIAL RELATIONSHIPS

This skill captures a variety of spatial relationships (such as *above*, *on*, *under*, *far from*, etc.) between two to three objects. In the most simple case, we measure a model’s ability to understand common relationships between two objects. The difficulty is increased by combining simpler entities and requiring the ability to reason about implicit relationships. We use an LLM as described in Sec. B.3 to create these prompts, subsample and manually verify prompts are reasonable.

B.4.2 ACTION

This skill examines whether the model can bind the right action to the right object, including unusual cases where we flip the subject and the object (i.e.*Reverse actions*). An example of a reverse setup is that we swap the entities in ‘A penguin is diving while a dolphin swims’ and create ‘A penguin is swimming while a dolphin is diving’. We vary the difficulty by increasing the number of entities. We use an LLM as described in Sec. B.3 to create these prompts, subsample and manually verify prompts are reasonable.

B.4.3 SCALE

We measure whether the model can reason about scale cues referring to commonly used descriptors such as *small*, *big* or *massive*. To reduce ambiguity, we typically refer to two objects, so that they can be compared in size. We test the ability to implicitly reason about scales by having *Comparative* prompts that contain several statements about objects, their relations and sizes. We use an LLM as described in Sec. B.3 to create these prompts, subsample and manually verify prompts are reasonable.

B.4.4 COUNTING

The simplest sub-skill *Simple modifier* contains a number (digits such as “2”, “3” or numerals such as “two”, “three”) and an entity. When selecting a vocabulary of words, we aimed to include words that occur less frequently in ordinary language (for example, “lemur” and “seahorse” occur less frequently than “dog” and “cat”) (Speer, 2022). We focus on numbers 1—10, or 1—5 in more complex cases. Complexity is introduced by combining simple prompts containing just one attribute into compositional prompts containing several attributes. For example, simple prompts “1 cat” and “2 dogs” are combined into a single prompt “1 cat and 2 dogs” in the sub-skill *Additive*. We also test approximate understanding of quantities based on linguistic concepts of *many* and *few* in the *Quantifiers and negations* sub-skill.

B.4.5 SHAPE

We test for basic and composed shapes, where composed shapes include objects arranged in a certain shape. *Hierachial shapes* are of the following type: “The letter H made up of smaller letters S”. This kind of challenge is used to study spatial cognition and the trade-off between global and local

Skill	Sub-skill	Examples
Spatial Relationships rel/spatial	Simple	A cat above a dog. The lemon is in the middle of the apples. A bus is behind a truck going down the highway.
	Composed	The cat is near the banana. The banana is below the horse. The horse is on the truck.
Action rel/action	1, 2, or 3 entities	A bear is running through a field. A basketball is passed to a team member.
	Reverse actions	A ladybug is riding on the back of a flying unicorn. A koala climbs a tree, an eagle stands on the branch and a penguin flies overhead.
Scale rel/scale	Single object	A small ship in a bottle. A giant couch in a field.
	Comparative	A garlic is next to an onion and a tomato on a cutting board. The onion is larger than the garlic. The tomato is smaller than the garlic.
Counting att/count	Same size	The table is the same size as the cake. The mouse is the same size as the dragon.
	Simple modifier	2 cats. Four lemurs.
	Additive	5 burgers and one bonsai. 1 baobab, 2 cats and 3 dogs.
	Quantifiers and negations	Some shirts and some pizzas. There are more shirts than pizzas. An image with fewer dogs than cats. An image with no flowers in the vase.
Shape att/shape	Basic shapes	A line. An octagon.
	Composed shapes	A star-shaped cookie. Strawberries arranged in a heart shape.
	Hierarchical shapes	A square made of smaller letters g. A smiley face made of strawberries.
Text Rendering render/text	Rendering	A creature with a clock shaped head with the words "flimflam, bishbash, gorp" written on it. The creature has a small body and two legs, and is pointing at the ground. There are two clocks on the ground, one showing the time of 7:24 and the other showing 3:30 p.m.
	Numerical	equation of "3+4 = 7" etched into a rock. "Lorem ipsum dolor sit amet, \$%&*(), 12345 + adipiscing elit" written on a chalkboard with a piece of chalk next to it.
	Font	"congratulations" written in fancy decorative cursive font on an antique 1920's typewriter. "happiness" written in decorative font with a happy face next to it.
Color att/color	Simple colors	A pink salad. A grey vase.
	Composed expressions	A pink unicorn, a white airplane, and a green potato. A yellow couch and a green cookie.
	Colors and abstract shapes	A red rectangle on a green background. A pink circle on a green background.
	Descriptive color terms	A pastel coloured train passing through the station. A rainbow-colored bicycle leaning against a wall.
	Stroop	Text saying "yellow" in blue letters. Text saying "black" in green letters.

Table 8: Breakdown by skill and sub-skill including examples of prompts.

Skill	Sub-skill	Examples
Surface Characteristics att/texture+material	Texture only	A fluffy floor in the bathroom. There is a silky fabric on a bumpy couch in the room.
	Material only	There is a metal lime in the bowl. A paper snake on the table.
	Combined	There is a soft floor made of wax and a shiny silver table in the room. A glossy diamond road.
Style att/style	style	A brightly colored canal in Venice, by Canaletto A cartoon of a cat by Goya.
	Visual medium	A sketch of a drawing of a flower in a pot. A glass vase in the style of el greco and the impressionists.
Language Complexity lang/complexity	Negation	A pencil sharpener without any pencils in it. A belt buckle with no belt. A leaflet with no text on it.
	Long prompt	An outdoors top-down view of purple sidewalk chalk on a concrete sidewalk reading, "Fear/ of/ Chores". The left side of the concrete slab has green algae growth that fades to the right. Small bits of smashed acorns are scattered across the slab.
	True paraphrase	The giraffe feeds the cat. The bystanders watch the dog.
	False paraphrase	The snake observes the kangaroo. The horse looks at the deer.
Compositional Language lang/compositional	Vary number of entities & attributes	An orange metal train. A plastic couch, a cyan blueberry, and 2 plates. 3 wooden pencils and 1 plastic fly. A brown plastic bus, a yellow plastic vase, and a yellow wooden salad.
Named Entities	Landmarks ne/landmarks	Ulvetanna Peak, Queen Maud Land, Antarctica during sunrise. Ashikaga Flower Park with beautiful wisteria in shades of violet and white. Burj Al Arab Jumeirah hotel with fireworks in the night and glowing water around.
	Animal Characters ne/characters	Grumpy Cat is sitting on a couch with a catnip toy. A cartoon of Laika playing in the snow. the lion cub named Simba is catching a ball.
	Vehicles ne/vehicles	A BMW M3 is on the road. A Ferrari is driving through roads in an Italian landscape. A Opel Ampera is upside down.
	Products ne/product	A bottle of Irn-Bru is sitting on a shelf. A Gucci bag with a red and white striped pattern. A Samsung Galaxy S III is on a car seat.
Artwork ne/artwork		Charles IV of Spain and His Family is being painted on an easel. A painting of The Milkmaid hangs on the wall of a living room. A painting of Bacchus and Ariadne hanging in a stone building.

attention in literature on higher-order cognition ([Delis et al., 1986](#)). We use an LLM as described in Sec. [B.3](#) to create these prompts, subsample and manually verify prompts are reasonable.

B.4.6 TEXT RENDERING

We investigate a model’s ability to generate text (both semantically meaningful and meaningless), including text of different lengths. We further test for the ability to generate symbols and numbers, as well as different types of fonts. We use an LLM as described in Sec. [B.3](#) to create these prompts, subsample and manually verify prompts are reasonable.

B.4.7 COLOUR

The simplest prompts in this skill include basic colours bound to objects. As before, we introduce complexity by combining several simpler prompts (either two or three objects bound with a colour attribute). To include diversity of possible colour attributes, we also test descriptive colour terms such as “pastel” or “rainbow-coloured”. Finally, the sub-skill *stroop* contains prompts of the type “Text saying ‘blue’ in green letters” similar to the incongruent condition in the Stroop task ([Stroop, 1935](#)) used to study interference between different cognitive processes.

B.4.8 SURFACE CHARACTERISTICS

Surface characteristics include texture and material. We first test for each sub-skill individually, and then combined. Generally, some prompts in this skill can be difficult to visualise as they might include descriptions that are typically of tactile nature (“abrasive” or “soft”).

B.4.9 STYLE

We divide prompts into two sub-skills: one depicting a style of an artist, and another to capture different visual mediums (such as *photo*, *stained glass* or *ceramics*). We use an LLM as described in Sec. [B.3](#) to create these prompts, subsample and manually verify prompts are reasonable.

B.4.10 NAMED ENTITY

This skill evaluates a model’s knowledge of the world through named entities, focusing on specific entity types such as *landmarks*, *artwork*, *animal characters*, *products*, and *vehicles*, which are free of personally identifiable information (PII).

For the landmark class, we choose landmarks from the Google Landmarks V2 dataset and ensure we cover different continents and choose landmarks with high popularity ([Weyand et al., 2020](#)). Given this set of landmarks, we use an LLM as described in Sec. [B.3](#) to create these prompts, subsample and manually verify prompts are reasonable.

For the other classes, to curate diverse named entities, we first gather candidates from Wikidata using SPARQL queries. A simple query (e.g., `instance of (P31) is painting (Q3305213)`) might yield an excessively large number of candidates. Therefore, we impose conditions to narrow down the query responses. See our criteria below.

Artwork : Created before the 20th century; any media; any movement

Animal Characters : Anthropomorphic/fictional animals; real animals with names

Products : Electric devices; food/beverage; beauty/health

Vehicle : Automobiles; aircraft

Once we have a candidate set for each entity class, we focus on selecting reasonably popular entities that are widely recognised and appropriate to present to models. We assess popularity using the number of incoming links to and contributors on their English Wikipedia pages as proxies ([Geva et al., 2021](#)). Finally, we manually curate the final set of named entities, selecting them based on their ranked popularity scores.

B.4.11 LANGUAGE COMPLEXITY

We evaluate models on prompts with “tricky” language structure / wording. For this skill, we include 4 sub-skills: *negation*, *long prompt*, and *true / false phrases*. We sampled 19 prompts for *negation* from LVIS (Gupta et al., 2019), COCO stuff (Caesar et al., 2018), and MIT Places (Zhou et al., 2017); 58 *true paraphrases* and 58 *false paraphrases* from BLA (Chen et al., 2023); and finally crowdsourced 38 for *long prompt* with the help from English major raters. It should be noted that while we do not cover the entire spectrum of complex language (e.g. passives, coordination, complex relative clauses, etc.), the subcategories included cover the most prominent pain points of image generation models per our experimentation.

We also include a language complexity metric which can be run over all prompts. Here we treat language complexity from two perspectives – semantic and syntactic.

- *Semantic complexity*. The quantity of semantic elements included in a prompt.
- *Syntactic complexity*. The level of complexity of the syntactic structure of a prompt.

Concretely, we define *semantic complexity* as the number of entities extracted from a prompt. Taking the visual relevance of the task into account, we apply *Stanford Scene Graph Parser* (Schuster et al., 2015) for entity extraction and count the number of unique entities as the proxy for semantic complexity. For *syntactic complexity*, we implement a modified variant of Ohta et al. (2013) to look for the deepest central branch in the dependency tree of a prompt (pseudo-code below)[§] to gauge the complexity of its syntactic structure.

```

1 def central_depth(node) -> Tuple[int, int]:
2     return (max(central_depth(child)[1]+1 in node.children if child.position < node.position),
3             max(central_depth(child)[0]+1 in node.children if child.position > node.position))

```

C GECKO METRIC: MORE DETAILS

C.1 LLM PROMPTING FOR GENERATING COVERAGE

```

1 """
2 Given a image description, label the visually groundable words in the description, and a score indicating how
3 visually groundable it is.
4 Classify each word into a type (entity, activity, attribute, counting, color, material, spatial, location,
5 shape, style, other).
6 Description:
7 Portrait of a gecko wearing a train conductor's hat and holding a flag that has a yin-yang symbol on it.
    Woodcut.
8 The visual-groundable words and their scores are labelled below:
9 {1}[Portrait, style, 0.8] of {2}[a, count, 1.0] {3}[gecko, entity, 1.0] {4}[wearing, activity, 1.0] {5}[a,
    count, 1.0] {6}[train conductor's hat, entity, 1.0] and {7}[holding, entity, 1.0] {8}[a, count, 1.0]
    {9}[flag, entity, 1.0] that has {10}[a yin-yang symbol, entity, 1.0] on it. {11}[Woodcut, material,
    1.0].
10
11 Description:
12 square blue apples on a tree with circular yellow leaves
13 The visual-groundable words and their scores are labelled below:
14 {1}[square, shape, 1.0] {2}[blue, color, 1.0] {3}[apples, entities, 1.0] {4}[on, spatial, 1.0] {5}[a, count,
    1.0] {6}[tree, entity, 1.0] with {7}[circular, shape, 1.0] {8}[yellow, color, 1.0] {9}[leaves, entity,
    1.0]
15
16 Description:
17 A small dog running on a beach happily on a sunny day
18 The visual-groundable words and their scores are labelled below:
19 {1}[A, count, 1.0] {2}[small, attribute, 0.3] {3}[dog, entity, 1.0] {4}[running, activity, 1.0] {5}[on,
    spatial, 1.0] {6}[a, count, 0.0] {7}[beach, entity, 1.0] happily on {8}[a, count, 0.0] {9}[sunny,
    attribute, 0.5] day.
20
21 Description:
22 acrylic drawing, illustration, multiple mushrooms and pink jello, naive, flat, sketchy, purple background.
23 The visual-groundable words and their scores are labelled below:
24 {1}[acrylic drawing, style, 1.0], {2}[illustration, style, 0.2], {3}[multiple, count, 0.5] {4}[mushrooms,
    entity, 1.0] and {5}[pink, color, 1.0] {6}[jell, entity, 1.0], {7}[naive, attribute, 0.1], {8}[flat,
    attribute, 0.8] {9}[sketchy, style, 0.8] {10}[purple, color, 1.0] {11}[background, entity, 1.0].
25
26 Description:
27 a girl with many braids, riding away on her bike through the city, children's book cover illustration,
    detailed background, vibrant colors

```

[§]As an implementation note, we implemented Schuster et al. (2015) and Ohta et al. (2013) with SpaCy 2 (Honnibal & Montani, 2017) as the workhorse parsing backend.

```

28 The visual-groundable words and their scores are labelled below:
29 {1}[a, count, 1.0] {2}[girl, entity, 1.0], with {3}[many, counts, 0.8] {4}[braids, entity, 1.0], {5}[riding
   away, activity, 1.0] on her {6}[bike, entity, 1.0] through the {7}[city, place, 0.8], {8}[children's
   book cover illustration, style, 0.8], {9}[detailed background, entity, 1.0] {10}[vibrant colors, colors
   , 1.0].
30
31 Description:
32 """

```

Listing 3: Sample LLM template for generating word coverage.

C.2 LLM PROMPTING FOR GENERATING QAs

```

1 """
2 Given a image description, generate one or two multiple-choice questions that verifies if the image
   description is correct.
3 Classify each concept into a type (object, human, animal, food, activity, attribute, counting, color, material
   , spatial, location, shape, other), and then generate a question for each type.
4
5 Description:
6 A man posing for a selfie in a jacket and bow tie.
7 The visual-groundable words and their scores are labelled below:
8 A {1}[Man, human] {2}[posing, activity] for a {3}[selfie, object] in a {4}[jacket, object] and a {5}[bow tie,
   object].
9 Generated questions and answers are below:
10 About {1}:
11 Q: is there a man in the image?
12 Choices: yes, no
13 A: yes
14 About {2}:
15 Q: is the man posing for the selfie?
16 Choices: yes, no
17 A: yes
18 About {3}:
19 Q: is the man taking a selfie?
20 Choices: yes, no
21 A: yes
22 About {4}:
23 Q: is the man wearing a jacket?
24 Choices: yes, no
25 A: yes
26 About {5}:
27 Q: is the man wearing a bow tie?
28 Choices: yes, no
29 A: yes
30
31
32 Description:
33 A horse and several cows feed on hay.
34 The visual-groundable words and their scores are labelled below:
35 A {1}[horse, animal] and {2}[several, count] {3}[cows, animal] {4}[feed, activity] on a {5}[hay, object].
36 Generated questions and answers are below:
37 About {1}:
38 Q: is there a horse?
39 Choices: yes, no
40 A: yes
41 About {2}:
42 Q: are there several cows?
43 Choices: yes, no
44 A: yes
45 About {3}:
46 Q: are there cows?
47 Choices: yes, no
48 A: yes
49 About {4}:
50 Q: are the horse and cows feeding on hay?
51 Choices: yes, no
52 A: yes
53 About {5}:
54 Q: is there hay?
55 Choices: yes, no
56 A: yes
57
58 Description:
59 ...
60
61 Description:
62 """

```

Listing 4: Sample LLM template for generating QAs.

C.3 DISCUSSION

Here we discuss the potential limitations of the models we rely on and how we mitigate those issues, as well as give quantitative results around how impactful those potential issues are in practice.

Note that we treat these models as black boxes (we do not consider fine-tuning or further calibration) which gives the benefit, as shown in Table 4, that as models improve (e.g. by swapping PALM-2/PaLI for Gemini Flash), so too will our metric, with no further effort.

Potential issue 1: Hallucination of the QA model. The QA model could hallucinate, generating erroneous questions that are not grounded in the prompt, leading to worse performance. There are two factors that help mitigate this: (1) the use of the NLI model and (2) the ability of the LLM used to not hallucinate in the first place. We quantify the accuracy of the NLI model. To do so, we randomly chose 1.8K question/answer pairs from Gecko(R)/(S) and annotated whether they are hallucinations or not. We find that the NLI model is $\sim 93\%$ accurate on the PALM-2 setup. The utility of the NLI model is further validated in Table 5, which shows that adding NLI filtering improves results. We also evaluate how often the NLI model removes questions for the older PALM-2 model as opposed to Gemini Flash. We find that the NLI model removes 13% of questions for PALM-2 but only 2% for Gemini Flash, indicating that a better model will hallucinate less. This result validates the finding in Table 4 that as models improve, so too does our metric. In all, we find that hallucination can be mitigated effectively through the use of the NLI model and that with better LLMs, the impact of this issue will be diminished.

Potential issue 2: Bias of the VQA model. The VQA model could be biased or give poor scores. We note several factors that indicate the scores are useful and that these models do not suffer from severe bias. First, Cho et al. (2023a) have validated that PaLI and even weaker models achieve high accuracy on such VQA style questions. We also use the largest model – PaLI-17B; prior work has found that versions with the largest language component are better calibrated (Kostumov et al., 2024). Second, we break down the VQA score into multiple questions and so we are more robust to incorrect scores arising from a single question. Third, we check for a strong ‘yes’ bias, which has been found in prior work evaluating VQA models (Agrawal et al., 2018) and is relevant, as many of our QAs are yes/no questions due to the few-shot prompt. We evaluate how Gemini Flash responds to ‘blind’ questions: given no image and a question, will the VQA model always output a given answer. We find that we obtain 20% ‘yes’ answers and 80% ‘no’, indicating that the model does not have a strong ‘yes’ bias. We also note that if a model is biased, it is equally biased for any input, which means that the relative comparison is valid.

Finally, our comprehensive results demonstrate that this potential bias is *not* a problem. First, we ablate the utility of the scoring component in Table 5 and find that it improves results. Second, we find that our metric performs well, obtaining 72/79% agreement with human preference and on average 0.53 correlation (see Table 4). If the VQA model were terribly biased, it would *ignore* visual input, leading to chance performance on the pair-wise instance scoring task. Thus the high performance on our comprehensive benchmark, as well as our ablations, demonstrate the utility of leveraging the scores from the VQA models.

D HUMAN ANNOTATION: MORE DETAILS AND EXPERIMENTS

D.1 ANNOTATION TEMPLATES

Likert scale. We follow the template of Cho et al. (2023a) and collect human judgements using a 5-point Likert scale by asking the annotators “How consistent is the image with the prompt?” where *consistency* is defined as how well the image matches the text description. Annotators are asked to choose a rating from the given scale, where 1 represents *inconsistent* and 5 *consistent*, or a sixth *Unsure* option for cases where the text prompt is not clear. Choosing this template enables us to compare our results with previous work, but does not provide fine-grained, word-level alignment information. Moreover, while Likert provides a simple and fast way to collect data, challenges such as defining each rating especially when used without textual description (e.g., what 2 refers to in terms of image–text consistency), can lead to subjective and biased scores (Heo et al., 2022; Liang et al., 2020).

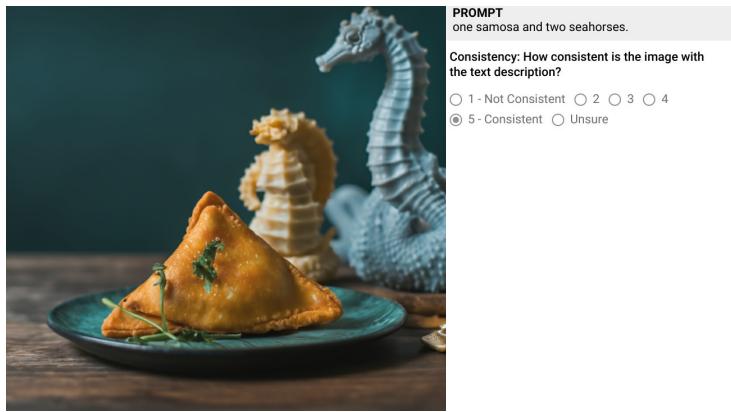


Figure 9: **Likert annotation template user interface.** Example depicting the interface shown to the annotators when performing evaluation tasks with the Likert template. Raters are given the prompt and image and asked to rate on a 5-point scale how consistent the image is with respect to the prompt. An *Unsure* option is also given the annotators.

Word-level alignment (WL). To collect word-level alignment annotations, we use the template of Liang et al. (2023) and define an overall image–text alignment score using the word-level information. Given a text–image pair, raters are asked to annotate each word in the prompt as *Aligned*, *Unsure*, or *Not aligned*. Note that for each text–image pair under the evaluation, the number of effective annotations a rater must perform is equal to the number of words in the text prompt. Although potentially more time consuming than the Likert template, we find that annotators spend ~ 30 s more to rate a prompt–image pair with WL than Likert.

We compute a score for each prompt–image pair per rater by aggregating the annotations given to each word. A final score is then obtained by averaging the scores of 3 raters.

DSG(H). We also use the annotation template of Cho et al. (2023a) that asks the raters to answer a series of questions for a given image, where the questions are generated automatically for the given text prompt as discussed in Cho et al. (2023a).

In addition, raters can mark a question as *Invalid* in case a question contradicts another one. The total number of *Invalid* ratings per evaluated generative model is given in App. D.1. Annotators could also rate a question as *Unsure*, in cases where they do not know the answer or find the question subjective or not answerable based on the given information.

For a given prompt, the number of annotations a rater must complete is given by the number of questions. We calculate an overall score for an image–prompt pair by aggregating the answers across all questions, and then averaging this number across raters to obtain the final score.

Side-by-side (SxS). We consider a template in which pairs of images are directly compared. The annotators see two images from two models side-by-side and are asked to choose the image that



Figure 10: **Word-level annotation template user interface.** Example depicting the interface shown to the annotators when performing evaluation tasks with the WL template. Raters are asked to click on words they find are not aligned with image, and double click on the words where they are unsure.

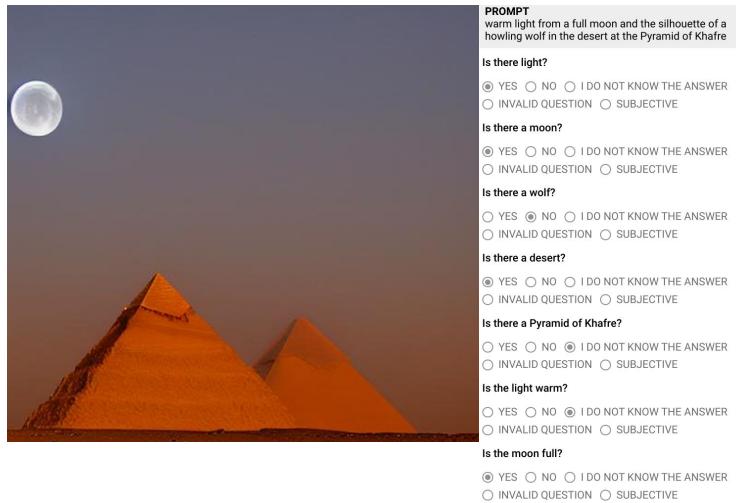


Figure 11: **DSG(H) annotation template user interface.** Example depicting the interface shown to annotators when performing evaluation tasks with the DSG(H) template. Raters are given the image, prompt, and respective automatically generated questions. There are 5 options for answering each question. In our analysis, both *I do not know the answer* and *Subjective* answers are considered as *Unsure*.

is better aligned with the prompt or select *Unsure*. We obtain a score for each comparison by computing the majority voting across all 3 ratings. In case there is a tie, we assign *Unsure* to the final score of an image–prompt pair.

D.1.1 DATA COLLECTION DETAILS.

We recruited participants ($N = 40$) through a crowd-sourcing pool. The full details of our study design, including compensation rates, were reviewed by our institution’s independent ethical review committee. All participants provided informed consent prior to completing tasks and were reimbursed for their time. Considering all four templates, both Gecko subsets, and the four evaluated generative models, approximately 108K answers were collected, totalling 2675 hours of evaluation.

D.1.2 PERCENTAGE OF UNSURE RATINGS FOR EACH ANNOTATION TEMPLATE/MODEL

One of the innovations of our human evaluation setup is to allow for annotators to reflect uncertainty in their ratings. In Table 9 we show the percentage of *Unsure* ratings for each absolute comparison



Figure 12: **Side-by-side comparison annotation template user interface.** Example depicting the interface shown to the annotators when performing evaluation tasks with the side-by-side template. Raters are given a pair of images from different models, the prompt used to generate them and asked to pick which one is more consistent with the prompt. An *Unsure* option is also given.

	Muse	SDXL	SD1.5	Muse	SDXL	SD1.5
	WL L D(H) SxS	WL L D(H) SxS	WL L D(H) SxS	WL L D(H) SxS	WL L D(H) SxS	WL L D(H) SxS
Imagen	<<<<	= = = =	>>>>	>>>>	>>>>	>>>>
G(S)	<<<<	= = = =	>>>>	>>>>	>>>>	>>>>
G(S)-rel	<<<<	= = = =	>>>>	>>>>	>>>>	>>>>
G(R)	<= = >	<< = >	<< >>	<< >>	< < = =	< < = =
G(R)-rel	<< = =	<< > >	<< > >	<< > >	= > > >	= > > >
G2K-rel	<<<<	= < = =	>>>>	>>>>	> = > >	> = > >

Figure 13: **Comparing models using human annotations.** We compare model rankings on Gecko(S), Gecko(R), their reliable subsets G(S)-rel, G(R)-rel and both subsets (G2K-rel). Each grid represents a comparison between two models. Entries in the grid depict results for WL, Likert (L), DSG(H) (D(H)), and side-by-side (SxS) scores. The > sign indicates the left-side model is better, worse (<), or not significantly different (=) than the model on the top. Green indicates cases where all results were the same across templates, yellow where templates didn't disagree with each other, and red cases where at least template disagreed with others.

annotation template. Overall, we find that evaluations with Gecko(R) yield a higher percentage of *Unsure* ratings in comparison to Gecko(S).

Gen. model	WL		Likert		DSG(H)	
	Gecko(R)	Gecko(S)	Gecko(R)	Gecko(S)	Gecko(R)	Gecko(S)
Imagen	43.52	18.46	20.25	2.07	30.90	29.55
Muse	41.09	23.91	18.10	2.42	33.47	32.65
SDXL	20.09	20.94	4.24	2.05	31.77	29.64
SD1.5	13.35	26.48	10.04	4.09	37.02	33.08

Table 9: **Percentage of *Unsure* ratings.** Overall, evaluation with Gecko(S) yields fewer *Unsure* ratings across all models and templates.

D.2 ADDITIONAL EXPERIMENTAL RESULTS

D.2.1 PAIRWISE MODEL COMPARISONS WITH RELIABLE PROMPTS

D.2.2 CORRELATION ACROSS TEMPLATES AND MODELS.

We show the correlation between templates and models for both Gecko(R) and Gecko(S) in Table 10.

Gen. models	Gecko(R)						Gecko(S)					
	Likert vs WL		Likert vs DSG(H)		WL vs DSG(H)		Likert vs DSG(H)		Likert vs WL		WL vs DSG(H)	
	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman
SD1.5	0.56	0.64	0.56	0.60	0.57	0.65	0.60	0.61	0.60	0.62	0.74	0.76
SDXL	0.60	0.63	0.52	0.57	0.56	0.61	0.60	0.50	0.56	0.62	0.78	0.79
Muse	0.67	0.62	0.61	0.63	0.61	0.66	0.51	0.52	0.51	0.53	0.77	0.75
Imagen	0.65	0.67	0.59	0.62	0.68	0.71	0.63	0.57	0.59	0.62	0.81	0.80

Table 10: **Correlation between all absolute comparison templates.** We compute Pearson and Spearman correlation coefficients for all pairs of templates for both Gecko(R) and Gecko(S). We find significant results with $p < 0.001$ for all cases and that scores of all metrics are at least moderately correlated, with the finer-grained templates, WL and DSG(H), being more correlated with each other in comparison to Likert.

D.2.3 DISTRIBUTION OF SCORES PER PROMPT-IMAGE PAIRS ACROSS ANNOTATION TEMPLATES.

We plot the distribution of scores per each evaluated prompt-image pair for all the absolute comparison templates. The violin plots in Fig. 14-15 show the distributions for Gecko(R) and Gecko(S), respectively. It is possible to notice that scores obtained for Muse with WL and DSG(H) are more concentrated in values closer to 1 for both templates, corroborating findings from Sec. 4.2 where results showed Muse was the overall best model.

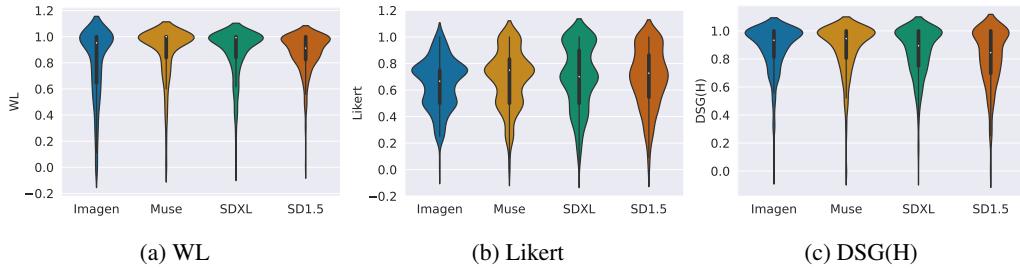


Figure 14: **Distribution of scores for Gecko(R).** We show violin plots for scores obtained with all absolute comparison templates.

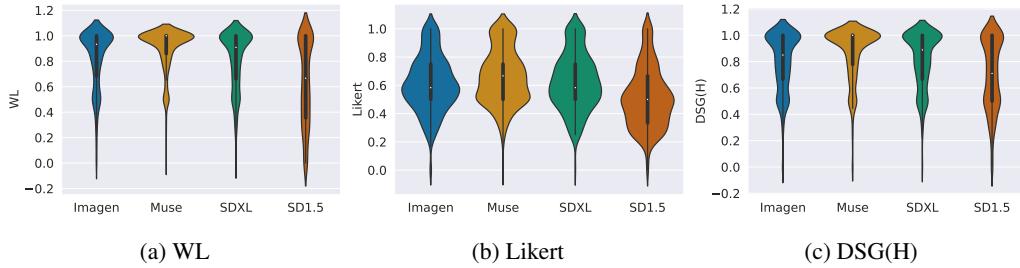


Figure 15: **Distribution of scores for Gecko(S).** We show violin plots for scores obtained with all absolute comparison templates.

D.2.4 SIDE-BY-SIDE TEMPLATE.

In Table 11 we show inter-annotator agreement results for the side-by-side annotation template for Gecko(R) and Gecko(S), along with the respective difference in agreement when using only the reliable prompts for both Gecko2K subsets. In Table 12 we present the results of the comparison between the side-by-side template and the absolute comparison ones.

	Gecko(R)	Gecko(R)-rel	Δ	Gecko(S)	Gecko(S)-rel	Δ
Imagen vs Muse	0.438	0.465	0.027	0.485	0.625	0.140
Imagen vs SDXL	0.440	0.425	-0.015	0.521	0.608	0.087
Imagen vs SD1.5	0.402	0.431	0.029	0.581	0.652	0.071
Muse vs SDXL	0.471	0.489	0.018	0.570	0.638	0.068
Muse vs SD1.5	0.539	0.592	0.053	0.600	0.617	0.017
SDXL vs SD1.5	0.389	0.438	0.049	0.522	0.562	0.040

Table 11: **Side-by-side template: inter-annotator agreement.** We compute Krippendorff’s α for Gecko(R) and Gecko(S) and the difference (Δ) in α when using only reliable prompts for both subsets of Gecko2K. In both cases, using the reliable subsets increases the overall inter-annotator agreement.

	WL	Likert	DSG(H)
Imagen vs Muse	0.736	0.755	0.728
Imagen vs SDXL	0.690	0.705	0.732
Imagen vs SD1.5	0.672	0.632	0.708
Muse vs SDXL	0.704	0.746	0.703
Muse vs SD1.5	0.759	0.740	0.749
SDXL vs SD1.5	0.693	0.700	0.689
Average	0.709	0.713	0.718

Table 12: **Comparing side-by-side and absolute templates on Gecko2K-rel.** We compare the side-by-side template with the absolute comparison ones by computing the accuracy obtained by WL, Likert, and DSG(H) scores when using them to compare pairs of images on Gecko2K-rel. In this case, the ground-truth is assumed to be the results obtained with the side-by-side template.

D.3 RELIABLE PROMPTS: EXAMPLES OF IMAGE-PROMPT PAIRS WITH HIGH HUMAN (DIS)AGREEMENT

In this section we show a representative list of prompts and corresponding images where human annotators were most likely to either agree or disagree in their ratings. The annotators agreed in ratings if they gave similar scores across for an image-text pair, meaning that the resulting mean variance was zero or close to zero. We refer to such prompts as “high agreement” prompts. In contrast, if annotators gave different ratings for a text-image pair, this would result in higher mean variance and we call such prompts “high disagreement” prompts.

To find prompts with high agreement across raters for all templates and all models, for each model-template combination we pick a subset of responses with low variance. Low variance is defined as the mean variance of a prompt-image pair for a model-template pair being below a certain threshold. The threshold is set as 10% of the maximum variance for that model-template set of ratings for both Gecko(R) and Gecko(S). Analogously, we also find a set of prompts with high disagreement; for this we find prompts that have mean variance above 1% of the maximum variance for a given template and for prompts from Gecko(R) and Gecko(S). The specific threshold value here is relevant only insofar as it captures at least 10 prompt-image pairs which we are interested in visualising. Then, we find prompts with high agreement by intersecting all model-template prompt sets where prompts have been selected based on the threshold. The procedure is analogous for low agreement prompts. For Gecko(R), both sets, namely the set of prompts with high agreement as well as the set of prompts with high disagreement have 34 prompts each. For Gecko(S), the set of prompts with high agreement has 62 prompts, while the set of prompts with high disagreement contains 85 prompts. A subset of 10 prompts for all different combinations is listed in Tables 13-16 and corresponding images are shown in the Figure 16-19.

Based on the analyses of such subsets, we observe several interesting trends. First, for Gecko(R) the prompts with higher agreement tend to be significantly shorter in length ($\mu = 54.32, \sigma = 32.18$) as measured by the number of characters, compared to the length of prompts with high disagreement ($\mu = 173.35, \sigma = 86.35$, Welch’s t-test $t(41.99) = -7.42$ ($p < 0.001$). The same observation holds for Gecko(S), where high agreement prompts were also significantly shorter ($\mu = 20.77, \sigma = 18.03$), than high disagreement prompts ($\mu = 82.48, \sigma = 95.17$, Welch’s t-

High Agreement Prompts (Gecko(R))	
1	three men riding horses through a grassy field
2	a small bathroom with a shower and a toilet
3	a wooden table with four wooden chairs in front of two windows
4	a large colgate clock is by the water
5	a slice of chocolate cake is on a small plate
6	a black cat sitting in a field of grass
7	The Statue of Liberty made of gold
8	a man riding skis down a snow covered slope
9	a vast, grassy field with animals in the distance
10	a plastic bento box filled with rice, vegetables and fresh fruit

Table 13: Selected prompts with a high level of agreement in scores among raters for Gecko(R).

test $t(92.20) = -5.80$ ($p < 0.001$). We further observe that prompts where raters tend to agree more are highly specific (i.e. they refer to one or just a few objects with few attributes), whereas prompts with high disagreement tend to describe more complex scenes with visual descriptors and often mentioning named entities or text rendering. Intuitively, this makes sense as longer prompts are more likely to require several skills.

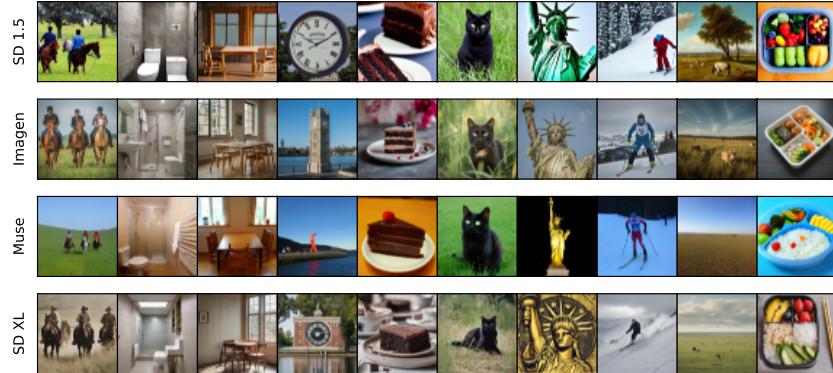


Figure 16: Images generated from Gecko(R) prompts with high level of agreement among raters. Prompts are listed in Table 13.

High Agreement Prompts (Gecko(S))	
1	A star-shaped cookie
2	a pastel coloured train passing through the station.
3	a green boat.
4	the cat wears a gray shirt and holds a frisbee
5	a red motorcycle.
6	a black fish.
7	a pink bottle.
8	two mushrooms.
9	five cats.
10	a dog named Balto is running on a beach.

Table 14: Selected prompts with a high level of agreement in scores among raters for Gecko(S).

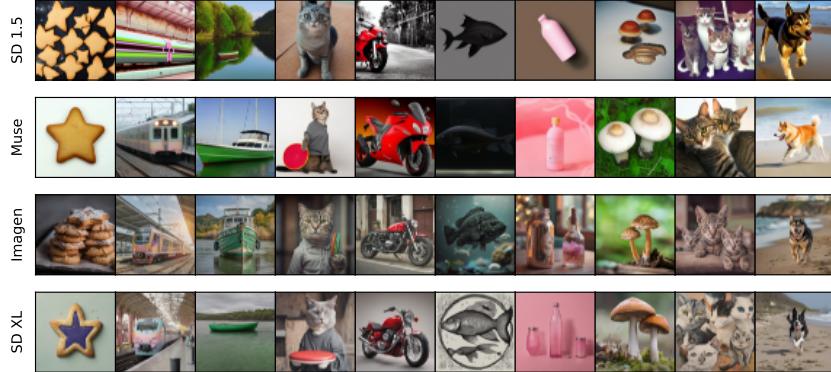


Figure 17: Images generated from Gecko(S) prompts with high level of agreement among raters. Prompts are listed in Table 14.

High Disagreement Prompts (Gecko(R))

- 1 Studio shot of sculpture of text 'cheese' made from cheese, with cheese frame.
- 2 vintage light monochrome six round and oval label set Illustration
- 3 There is a person snow boarding down a hill. There are tracks in the snow all around the snowboarder. There is a large rock in the snow next to them. There is a green pine tree in front of the snowboarder. The snowboarder is wearing blue ski pants and a blue and yellow jacket. They have a yellow snowboard on their feet.
- 4 pillow in the shape of words 'ready for the weekend', letterism, funny jumbled letters, [closeup]!!, breads, author unknown, flat art, swedish, diaper-shaped, 2000, white clay, surreal object photography
- 5 a sunflower field with a tractor about to run over a sunflower, with the caption 'after the sunflowers they will come for you'
- 6 a photo of a prison cell with a window and a view of the ocean, and the word 'freedom' painted on the glass
- 7 vehicle flying through a cyberpunk city 4 k, hyper detailed photograph
- 8 a scene with a city in the background, and a single cloud in the foreground, with the text 'contemplate the clouds' in rounded cursive
- 9 A pencil made of a tree branch with leaves
- 10 A wooden table that has a silver trophy in the middle of it. In front of the trophy are several bowls and dishes containing food. There is a loaf of bread on a block of wood at the front of the table.

Table 15: Selected prompts with a high level of disagreement in scores among raters for Gecko(R).

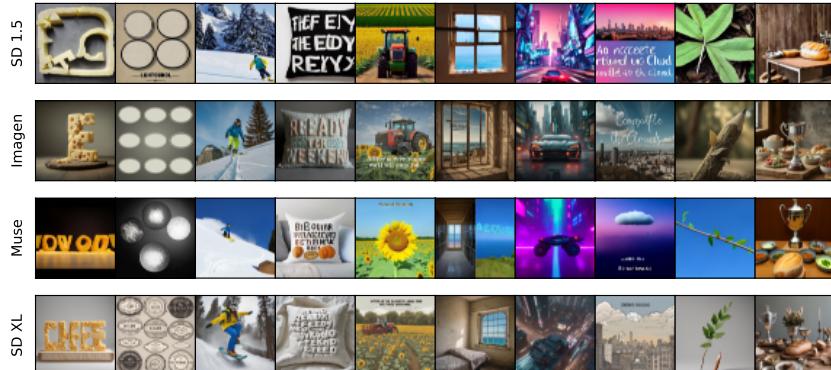


Figure 18: Images generated from Gecko(R) prompts with high level of disagreement among raters. Prompts are listed in Table 15.

High Disagreement Prompts (Gecko(S))									
1	the amazing view from the Halley Research Station in Antarctica on a clear night, the full Moon is rising and the sky is ablaze with the aurora australis, or polar lights.								
2	a futuristic sculpture made of smooth metal								
3	time lapse of sunrise over at the Hoover Dam								
4	A huge vase in the middle of a field towering over the lawn chairs.								
5	a bottle of Irn-Bru is sitting on a shelf.								
6	a lord howe island palm tree with a moon rising in the distance								
7	a long exposure image of the golden dunes at Playa del Ingles on the Canary Island, with a lone tourist								
8	The soup is behind the cheese platter, to the left of the wine glasses, and below the crackers.								
9	An alpaca and Chewbacca pose for a selfie at Machu Picchu								
10	the lion cub named Simba is catching a ball.								

Table 16: Selected prompts with a high level of disagreement in scores among raters for Gecko(S).

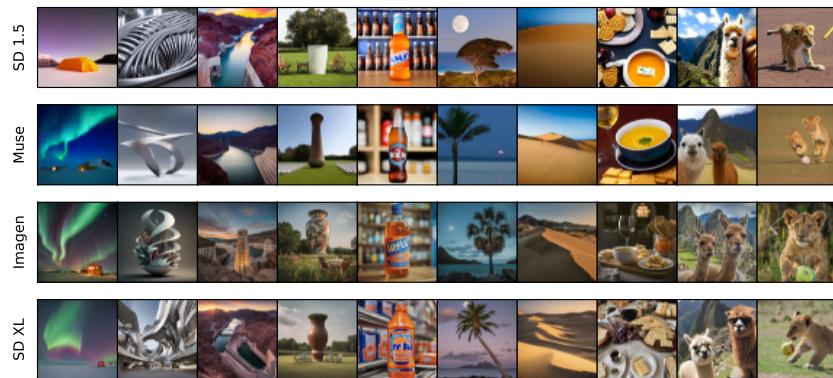


Figure 19: Images generated from Gecko(S) prompts with high levels of disagreement among raters. Prompts are listed in Table 16.

D.4 HUMAN EVALUATION TEMPLATES: CHALLENGING CASES

In Figs. 20, 21, and 22 we show examples of challenging cases for the absolute comparison templates.

Image	Ratings
	A Nexus One is placed on a bench. A Nexus One is placed on a bench. A Nexus One is placed on a bench.
	Some shirts and some pizzas. There are more shirts than pizzas. Some shirts and some pizzas. There are more shirts than pizzas. Some shirts and some pizzas. There are more shirts than pizzas.

Figure 20: **Examples of challenges for WL.** We show two examples of evaluated images, respective prompts and annotations from three raters. Each word is coloured according to the score given by the rater: green indicates *Aligned*, red *Not aligned*, and yellow *Unsure*. Both examples show that WL can be sensitive to words that are not relevant to the alignment evaluation. **Top:** All raters seem to agree it is not possible to tell whether a bench is represented in the image (hence the word is evaluated as *Unsure*). In spite of that, one of the raters disagrees on how to rate the “on a” preposition. **Bottom:** All raters seem to agree the quantity of shirts in the image does not reflect the prompt, but their ratings vary in terms of which words are rated as not aligned.

Image	Prompt	Rater 1	Rater 2	Rater 3
	A giraffe stands in the field.	4-Mostly consistent	5-Consistent	5-Consistent
	The raccoon holds the cat.	2-Mostly inconsistent	3-Somewhat consistent	4-Mostly consistent

Figure 21: **Examples of challenges for Likert.** **Top:** Raters might take into account other aspects of the images besides alignment when evaluating a prompt-image pair. In this example, although the image is perfectly consistent with the prompt, one of the raters penalised its score. We hypothesise they took into account the fact the generated image is in grey scale. **Bottom:** “Uncalibrated” scores across raters. The scores of all three raters reflect the imperfect consistency between prompt and image, but each rater penalised the score with different intensity.

Image	Prompt	Questions
	A church without a steeple.	Is there a church? Does the church have a steeple? Is the steeple missing?
	A wood carving of an owl.	Is there an owl? Is there a wood carving? Is the wood carving made of wood?

Figure 22: **Examples of challenges for DSG(H).** **Top:** Language complexity–Negation. As also shown in Fig. 4, the question generation is confused by the negation (asking if the church *has* a steeple as opposed to *does not have* a steeple). **Bottom:** Coverage. The question generation fails to capture that the owl should be represented as a wood carving.

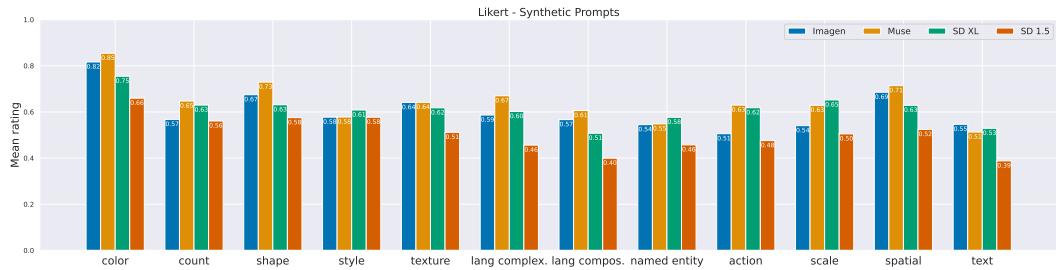


Figure 23: **Per skill results - Likert.** Muse scores the best in nine out of the twelve categories, and SD1.5 performs the worst in all categories. Focusing on Muse, SDXL, and Imagen, the models score above 0.5 on all categories. Recalling that the Likert scale is symmetric (0.0 being inconsistent, and 1.0 being consistent), we see that these three models are more consistent than inconsistent on average (albeit only slightly for skills such as ‘lang compos.’, ‘named entity’ and ‘text’).

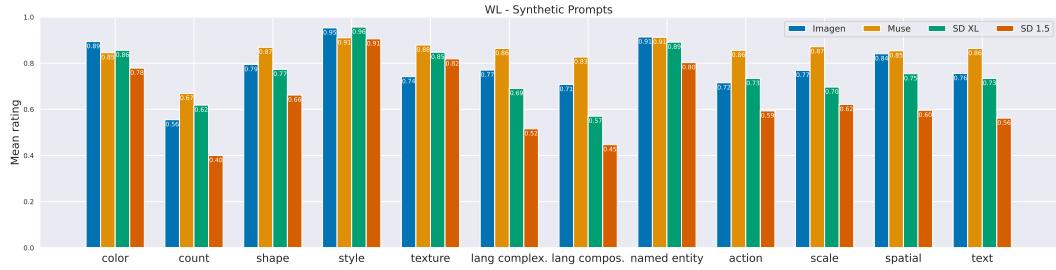


Figure 24: **Per skill results - WL.** Muse scores the best in ten out of the twelve skills, and SD1.5 performs the worst in all skills. Moreover, Muse scores higher than the other models by a noticeable margin (≥ 0.1) for the skills ‘lang compos.’, ‘action’, ‘scale’ and ‘text’. In this case, analysing the results by skill shows that we can contribute Muse’s higher average score (over the whole prompt set) mostly to these skills.

E T2I MODELS: ADDITIONAL COMPARISONS

E.1 ANALYSING MODEL RATINGS PER SKILL

In Figures 23, 24 and 25, we plot the mean ratings in different skills for Likert, WL and DSG(H), respectively. We focus on Gecko(S) because we have a skill/sub-skill label for each prompt. Our goal is to understand how the trends in model performance on the whole prompt set relate to their performance in individual skills. We provide an overview of the results in the captions for the plots. Overall, the results broken down by skill are consistent with the averages over the whole prompt set. In other words, if a model is better or worse on the full prompt set, this is generally true for the individual categories as well. Another observation is that COUNTING and COMPLEX LANGUAGE seem to be the most difficult skills judging by WL and DSG, but this is not as clear from Likert (where many categories seem just as difficult).

Further Breaking Down Skills. We can gain more insight into the skills of the models by looking at variation within a skill. Figure 26 shows sub-skills of the COLOUR prompts. We find that two sub-skills are more challenging, corresponding to prompts that require the models to combine multiple skills when generating the image (i.e., colour plus either composition or text rendering). For example, the ‘colour:composed’ sub-skill (COMPOSED EXPRESSIONS) includes prompts such as ‘A brown vase, a white plate, and a red fork.’ with variations in the colors/objects. The sub-skill ‘color:stroop’ (STROOP) contains prompts like ‘Text saying “green” in white letters.’ where the word in quotes differs from the color of the letters.

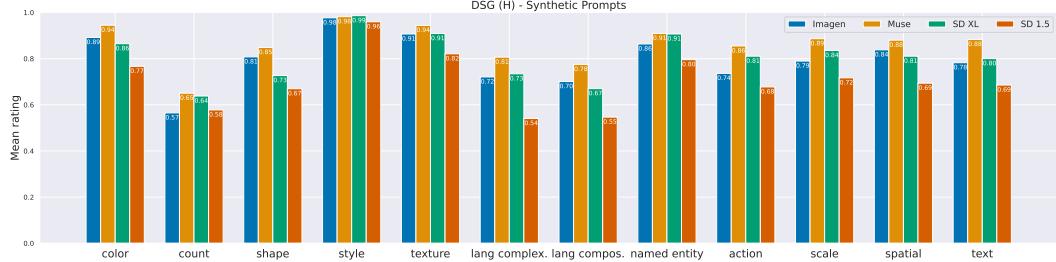


Figure 25: **Per skill results - DSG(H).** Muse performs well across the skills, being the best eleven out of twelve times (scoring very close to the top for ‘style’). On the other hand, SD1.5 scores the worst in all the skills. This is consistent with the average scores on the overall prompt set. We see that counting is the most difficult skill for Muse, SDXL, and Imagen. Aside from counting, the hardest skills for Muse are the language ones (‘lang complex.’ and ‘lang compos.’). This relative skill deficiency is not evident from the Likert and WL ratings, and therefore, the DSG ratings are better able to capture model shortcomings for prompts with more complex linguistic structure.

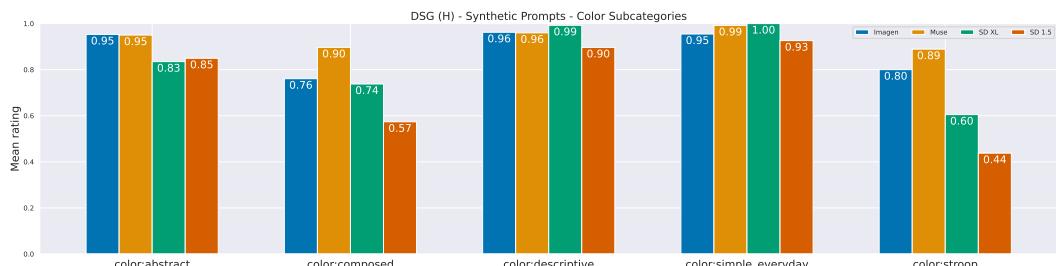


Figure 26: **Color sub-skill results - DSG (H).** We further break down the prompts in the color skill into five sub-skills. One observations is that two of the sub-skills (‘composed’ and ‘stroop’) are noticeably more difficult for all the models. This can be explained by the fact that they combine multiple skills: ‘composed’ includes prompts with multiple colors/objects, and ‘stroop’ includes text rendering in a certain color. Hence, while models may perform well on a skill overall, the sub-skills can illuminate where they struggle in generating images aligned with more complex prompts. On the other hand, models perform well with both abstract and everyday colors, likely because these are more commonly seen during training.

E.2 MODEL COMPARISONS WITH TIFA160

We augment the experiment from Fig. 13 in Sec. 4.2 by performing a similar analysis with TIFA160. We generate images with this set of prompts and perform human evaluation following the same protocol as for Gecko2K and its subsets. In Fig. 27, we show the results of pairwise model comparisons with TIFA160 carried out with the Wilcoxon signed-rank test with $p < 0.001$. Results show that all three versions of Gecko prompts are able to better distinguish models by finding more significant comparisons between them.

	Muse	SDXL	SD1.5	Muse	SDXL	SD1.5	SD1.5			
	WL	L	D(H)	WL	L	D(H)	WL	L	D(H)	
Imagen	G(S)-rel	<	<	<	=	=	=	>	>	>
	G(R)-rel	<	<	=	<	<	>	<	<	>
	G2K-rel	<	<	<	=	<	=	>	>	>
TIFA160	=	=	=	=	=	>	>	>	>	>

Figure 27: **Comparing models using human annotations.** We compare model rankings on the reliable subsets of Gecko(S) (G(S)-rel), Gecko(R) (G(R)-rel), both subsets (G2K-rel), and TIFA160. We perform the Wilcoxon signed-rank test for all pairs of models ($p < 0.001$) and post-hoc comparison based on average ratings. Each grid represents a comparison between two models. Entries in the grid depict results for WL, Likert (L), and DSG(H) (D(H)) scores. The $>$ sign indicates the left-side model is better, worse ($<$), or not significantly different ($=$) than the model on the top. All three versions of Gecko prompts are able to better distinguish models by finding more significant comparisons between them.

F AUTO-EVAL METRICS: ADDITIONAL EXPERIMENTS ON T2I

F.1 INTUITIVE EXPLANATION OF EACH TASK

We use this section to give an intuitive explanation of each task – *point-wise instance scoring*, *pair-wise instance scoring*, *model ordering* – as well as how they can lead to different outcomes in terms of metric ranking on a toy setting. We evaluate the different auto-eval metrics on our actual dataset in Sec. 6, where we find that *in practice* metrics do achieve different rankings on these tasks.

Point-wise instance scoring. Point-wise instance scoring evaluates how well a metric ranks generations for a single model. Assume we have a set of generations for Model A over a prompt set \mathcal{P} and metric m that calculates scores $m(A(p), p)$ for a generation $A(p)$ for a prompt $p \in \mathcal{P}$. We also have a human rating $h(A(p), p)$ on some numerical scale between $[0, 1]$ where 1 indicates a perfect generation and 0 a terrible one. This evaluation compares how well the metric scores correlate with the human ones over that prompt set for Model A . If we have multiple models, we average the correlation coefficients obtained for each model.

Pair-wise instance scoring. Pair-wise instance scoring compares two metrics given a pair of generations for a prompt. Assume we have two generations $A(p), B(p)$ for the prompt p for Model A and Model B . We also have a human preference (e.g. Model $A >$ Model B), and a metric m which gives a score (e.g. $m(A(p), p)$) for a generation. Pairwise instance scoring would evaluate whether the relationship between the scores (i.e. $m(A(p), p) > m(B(p), p)$) matches that of the human rating. If the relationship matches, then we say m is correct for that example, else it is incorrect. To get an average accuracy for a given metric within a dataset, we count the number of examples for which m predicts the correct relationship for all prompts / model pairs and divide by the number of comparisons.

Model ordering. For model ordering, we compare two metrics over a set of prompts and corresponding generations. We average the scores for a metric across all prompts to get $m_{avg}(A) = \sum_p m(A(p), p) / |\mathcal{P}|$. We then evaluate (for all model pairs) how often the model ordering from human evaluation matches that obtained when comparing a given model pair: e.g. $m_{avg}(A)$ vs $m_{avg}(B)$ for Model A and Model B .

Model 1	Model 2	'GT'	Gecko	DSG	TIFA	CLIP	VNL	VQAScore
Imagen	Muse	<	<	<	<	<	<	<
Imagen	SDXL	--	<	<	--	<	--	<
Imagen	SD1.5	>	>	>	>	<	>	>
Muse	SDXL	>	>	--	--	<	--	--
Muse	SD1.5	>	>	>	>	>	>	>
SDXL	SD1.5	>	>	>	>	>	>	>

Figure 28: **Comparing model ordering obtained from humans and auto-eval metrics on G2K-rel.** We show the ‘GT’ human ordering and the predicted ones for auto-eval metrics. < means Model 1 < Model 2, > Model 1 > Model 2 and -- that no significant relation was found. While CLIP performs poorly, mistaking wins with losses, no other metric confuses a win with a loss.

Toy example. Why these different evaluation procedures can give different results for a metric is subtle. Consider the following toy setting. We have two models: *A* and *B*, and 3 prompts. For prompt p_1 , Model *A* is clearly much better than Model *B*, which is terrible, but for prompts p_2, p_3 , Model *A* is ever so slightly worse but both are reasonable. Intuitively, Model *A* is better than Model *B* on this prompt set.

1. *Example 1:* Now imagine we have a metric that can do a good job of doing pairwise instance scoring but is not well calibrated across prompts (e.g. a score does not indicate an overall notion of alignment – 0.7 for one prompt could correspond to a poor generation but a score of 0.5 on another denotes a high quality generation). So in this toy example, that metric gives the following scores for Model *A*: $p_1 = 0.1, p_2 = 0.1, p_3 = 0.1$ and for Model *B*: $p_1 = 0.05, p_2 = 0.8, p_3 = 0.8$. The comparisons are all right, but the average score for Model *A* is 0.1 and Model *B* is 0.55, which does not match the intuition that Model *A* is actually better than Model *B*.
2. *Example 2:* We can conversely have a metric that is well calibrated across prompts but not able to reliably pick up on subtle differences. Such a metric could have the following scores for Model *A*: $p_1 = 0.8, p_2 = 0.7, p_3 = 0.72$ whereas Model *B*: $p_1 = 0.1, p_2 = 0.69, p_3 = 0.71$. While the model ordering is correct, the pairwise comparisons for prompts p_2, p_3 are not.

These examples demonstrate how a metric can be good at pairwise comparison (e.g. a metric like CLIP) but be poor at model ordering, i.e. a metric can give scores that are not well calibrated across prompts. Similarly, a metric can be good at model ordering but bad at pairwise instance scoring because it does not capture subtle differences. We can use a similar logic to understand why the point-wise instance scoring and pair-wise instance scoring tasks can achieve differing results as well as the model ordering and point-wise instance scoring tasks.

F.2 MODEL-ORDERING EVALUATION

We provide detailed results for the model ordering experiment. The goal is to determine the each auto-eval metric can predict the significant relations found from human annotation. We use G2K-rel as it is the largest subset with agreement among annotation templates across models. We take the majority vote to determine the relationship and note that there is no template that disagrees with this vote but one template may find a significant relation where the others do not, or vice versa. For each model pair, we compare distributions of auto-eval metric predictions using the same Wilcoxon signed rank test to get the relationship predicted by the metric. We plot in Table 28 the results. CLIP performs poorly, confusing wins with ties. However, no other auto-eval metric confuses a win with a tie showing that these auto-eval metrics are already robust for this task. Of these metrics, we see that TIFA and Gecko are able to get the most number of significant relationships right.

Metrics	FT	Gecko(R)			Gecko(S)			Gecko(R)-Rel			Gecko(S)-Rel		
		WL	Likert	DSG(H)	SxS	WL	Likert	DSG(H)	SxS	WL	Likert	DSG(H)	SxS
		SpearmanR			Acc	SpearmanR			Acc	SpearmanR			Acc
<i>Interpretable (QA/VQA)</i>													
TIFA _{PALM-2/PALI}	X	0.26	0.34	0.28	41.7	0.39	0.32	0.39	53.2	0.34	0.37	0.33	47.3
DSG _{PALM-2/PALI}	X	0.35	0.47	0.42	49.6	0.45	0.45	0.45	58.1	0.47	0.50	0.50	53.9
Gecko _{PALM-2/PALI}	X	0.41	0.55	0.46	62.1	0.47	0.52	0.45	74.6	0.52	0.58	0.53	71.3
Gecko _{Gemini Flash}	X	0.43	0.58	0.48	72.2	0.54	0.59	0.56	78.8	0.56	0.62	0.58	74.0
<i>Uninterpretable (single score)</i>													
CLIP	X	0.14	0.16	0.13	54.4	0.25	0.18	0.26	67.2	0.11	0.09	0.08	59.7
PyramidCLIP	X	0.26	0.27	0.26	64.3	0.22	0.25	0.23	70.7	0.26	0.26	0.23	65.8
VQAScore _{Gemini Flash}	X	0.42	0.54	0.45	73.1	0.51	0.57	0.49	76.5	0.51	0.59	0.52	73.9
VNL	✓	0.37	0.49	0.42	54.4	0.45	0.55	0.45	72.7	0.49	0.57	0.46	65.6
Gecko+VQAScore _{Gemini Flash}	X	—	—	—	81.0	—	—	—	86.3	—	—	—	82.7
Gecko+VQAScore _{Gemini Flash}	X	—	—	—	81.0	—	—	—	86.3	—	—	—	87.4

Table 17: **Correlation between VQA-based, contrastive, and fine-tuned (FT) auto-eval metrics and human ratings across annotation templates on Gecko2K and Gecko2K-Rel.** We observe a similar trend in Gecko2k and Gecko2K-Rel: Gecko performs the best across the board, and it can be improved by using a better language/VQA backend.

Metrics	FT	Gecko(R)			Gecko(S)			Gecko(R)-Rel			Gecko(S)-Rel		
		WL	Likert	DSG(H)	SxS	WL	Likert	DSG(H)	SxS	WL	Likert	DSG(H)	SxS
		Pearson			Acc	Pearson			Acc	Pearson			Acc
<i>Interpretable (QA/VQA)</i>													
TIFA _{PALM-2/PALI}	X	0.21	0.32	0.25	41.7	0.39	0.32	0.39	53.2	0.27	0.35	0.27	47.3
DSG _{PALM-2/PALI}	X	0.28	0.41	0.38	49.6	0.43	0.42	0.44	58.1	0.39	0.43	0.44	53.9
Gecko _{PALM-2/PALI}	X	0.38	0.51	0.42	62.1	0.46	0.48	0.46	74.6	0.50	0.55	0.51	71.3
Gecko _{Gemini Flash}	X	0.40	0.53	0.47	72.7	0.52	0.58	0.56	77.9	0.53	0.57	0.54	74.0
<i>Uninterpretable (single score)</i>													
CLIP	X	0.15	0.18	0.16	54.4	0.26	0.19	0.25	67.2	0.13	0.12	0.10	59.7
PyramidCLIP	X	0.29	0.30	0.26	64.3	0.28	0.27	0.25	70.7	0.31	0.28	0.25	65.8
VQAScore _{Gemini Flash}	X	0.35	0.44	0.36	73.1	0.42	0.53	0.41	76.5	0.41	0.47	0.42	73.9
VNL	✓	0.34	0.48	0.39	54.4	0.41	0.55	0.42	72.7	0.25	0.41	0.22	65.6
Gecko+VQAScore _{Gemini Flash}	X	—	—	—	81.0	—	—	—	86.3	—	—	—	82.7
Gecko+VQAScore _{Gemini Flash}	X	—	—	—	81.0	—	—	—	86.3	—	—	—	87.4

Table 18: **Pearson correlation between VQA-based, contrastive, and fine-tuned (FT) auto-eval metrics and human ratings across annotation templates on Gecko2K and Gecko2K-Rel.** Similar to the comparisons on Spearman correlation, Gecko again outperforms other auto-eval metrics (both QA/VQA and single score) with higher overall Pearson correlation. Swapping for the GeminiFlash backend leads to consistent performance improvement across templates.

F.3 ADDITIONAL PAIR-WISE INSTANCE SCORING AND POINT-WISE INSTANCE SCORING RESULTS ON GECKO2K, GECKO-REL

We report the Spearman Rank correlation of different auto-eval metrics on Gecko2K in Table 4. Here we report the Pearson correlation in Table 18 as well and both Spearman and Pearson on Gecko-Rel in Table 17. Results follow those in the paper: the Gecko metric is consistently best though VQAScore is a strong baseline. While DSG/TIFA perform better than CLIP on the absolute templates, CLIP performs better on SxS.

In SxS comparison, to investigate how the interpretable and uninterpretable metrics can be combined to achieve better results, we also take the samples on which Gecko and VQAScore agree, and compute the accuracy of prediction on them. We found that we can improve agreement to >80% for this subset.

F.4 RESULTS FOR ADDITIONAL CLIP METRICS ON THE GECKO BENCHMARK

We compare the Gecko metric with several score-based auto-eval metrics (Li et al., 2023; Ilharco et al., 2021; Yu et al., 2022a; Sun et al., 2023; Zhai et al., 2023; Gao et al., 2022; Zeng et al., 2022), as well as QA-based metrics such as TIFA (Hu et al., 2023) and DSG (Cho et al., 2023a), and VNLI models (Yarom et al., 2024) in Table 19.

Generally, our Gecko metric outperforms the others and shows a higher correlation with most of the human annotation templates. DSG is the second best metric, except on SxS where it ranks third. It outperforms TIFA by a clear margin but falls behind Gecko. Finally, we note that Gecko even shows

Metrics	Gecko (R)						Gecko (S)					
	WL		Likert		DSG(H)		WL		Likert		DSG(H)	
	Pearson	Spearman-R										
BLIP-2 _{ITM}	0.25	0.22	0.24	0.19	0.23	0.21	0.28	0.23	0.13	0.16	0.25	0.23
CLIP-B/32	0.15	0.14	0.18	0.16	0.16	0.13	0.26	0.25	0.19	0.18	0.25	0.26
CLIP-B/32 _{LAION-2B}	0.26	0.21	0.24	0.22	0.26	0.21	0.28	0.24	0.23	0.22	0.26	0.24
CLIP-B/16	0.16	0.11	0.15	0.12	0.17	0.12	0.26	0.22	0.15	0.14	0.24	0.22
CLIP-L/14	0.18	0.16	0.16	0.14	0.18	0.15	0.26	0.23	0.17	0.16	0.25	0.24
CLIP-H/14 _{LAION-2B}	0.29	0.24	0.25	0.22	0.27	0.23	0.29	0.24	0.23	0.22	0.27	0.25
CLIP-g/14 _{LAION-2B}	<u>0.30</u>	0.24	0.26	0.23	0.28	0.23	<u>0.30</u>	0.25	0.24	0.23	0.28	0.25
CLIP-G/14 _{LAION-2B}	0.29	0.23	0.25	0.21	0.27	0.23	<u>0.30</u>	0.24	0.25	0.23	0.28	0.25
CoCa-L/14	0.28	0.23	0.26	0.22	0.26	0.21	0.29	0.25	0.24	0.22	0.28	0.26
EVA-02-CLIP-L/14	0.27	0.24	0.23	0.21	0.24	0.22	<u>0.30</u>	<u>0.26</u>	0.21	0.20	0.28	<u>0.27</u>
EVA-02-CLIP-E/14	0.28	0.23	0.24	0.20	0.27	0.22	0.28	0.23	0.23	0.22	0.26	0.24
EVA-02-CLIP-E/14+	<u>0.30</u>	0.24	0.24	0.21	0.27	0.23	0.29	0.24	0.25	0.23	0.28	0.25
SigLIP-B/16	0.26	0.21	0.22	0.18	0.27	0.21	0.29	0.25	0.22	0.21	<u>0.29</u>	0.26
SigLIP-L/16	0.28	0.24	0.26	0.22	<u>0.29</u>	0.25	0.29	<u>0.26</u>	0.23	0.22	<u>0.29</u>	<u>0.27</u>
PyramidCLIP-B/16	0.29	<u>0.26</u>	<u>0.30</u>	<u>0.27</u>	<u>0.29</u>	<u>0.26</u>	0.28	0.22	<u>0.27</u>	<u>0.25</u>	0.25	0.23
X-VLM _{16M}	0.17	0.11	0.21	0.09	0.25	0.15	0.26	0.23	0.23	0.16	0.24	0.23
TIFA _{PALM-2/PALI}	0.21	0.26	0.32	0.34	0.25	0.28	0.39	0.39	0.32	0.32	0.39	0.39
DSG _{PALM-2/PALI}	0.28	0.35	0.41	0.47	0.38	0.42	0.43	0.45	0.42	0.45	0.44	0.45
Gecko _{PALM-2/PALI}	0.38	0.41	0.51	0.55	0.42	0.46	0.46	0.47	0.48	0.52	0.46	0.45
Gecko _{Gemini Flash}	0.40	0.42	0.53	0.57	0.47	0.47	0.52	0.54	0.58	0.60	0.56	0.56
VQAScore _{Gemini Flash}	0.35	0.42	0.44	0.54	0.36	0.45	0.42	0.51	0.53	0.57	0.41	0.49
VNLI	0.34	0.37	0.48	0.49	0.39	0.42	0.41	0.45	0.55	0.55	0.42	0.45

Table 19: **Correlation between auto-eval metrics and human ratings across three annotation templates on Gecko2K.** Best results per model type are underlined; best results are in **bold**.

higher correlation than the supervised VNLI model. By using a stronger, Gemini Flash backend, Gecko performs best by a significant margin consistently.

Looking at efficient, score-based metrics, we find that PyramidCLIP achieves competitive correlations. Moreover, a larger pre-training corpus leads to better metrics; e.g., as seen by comparing CLIP-B/32 (trained on 400M images) and CLIP-B/32_{LAION-2B} (trained on 2B images). Finally, larger models are often better (e.g., SigLIP-L/16 vs. SigLIP-B/16), although these trends are less consistent (e.g., EVA-02-CLIP-L/14+ ≈ EVA-02-CLIP-E/14).

F.5 ANALYSING AUTO-EVAL METRIC RESULTS PER SKILL.

We present the per-skill Spearman Ranked Correlation between different auto-eval metrics and human annotation templates in Fig. 29. We observe a similar trend across the three plots, as we discussed in Sec. 6: Gecko is the best on handling prompts with “language complexity”, which can be attributed to the coverage tagging and filtering steps in its pipeline that make Gecko less prone to errors when processing long and complicated prompts. DSG is better on “compositional prompts”, as it can leverage its utilization of dependency graphs. VNLI and VQAScore demonstrate advantages in assessing “shape”, “color”, and “text” (e.g., TEXT RENDERING) prompts though we note they are both worse on the more complex prompts (e.g., “compositional” and “language complexity”). When leveraging a better QA/VQA model (e.g., GeminiFlash) for Gecko, we see improvements across the board; Gecko with GeminiFlash performs consistently the same or better than VNLI/VQAScore. TIFA and CLIP consistently perform poorly and worse than the other metrics. It is also worth noting that all metrics exhibit relatively poor performance on “text”, “style”, and “named identity”, highlighting the current lack of OCR and named recognition ability in existing contrastive, NLI and VQA models.

F.6 ADDITIONAL VISUALISATIONS.

We visualise additional examples in Fig. 30 for different categories (spatial, counting, text rendering, linguistic complexity, etc.). These examples demonstrate both differences arising from different annotation templates and also different metrics.

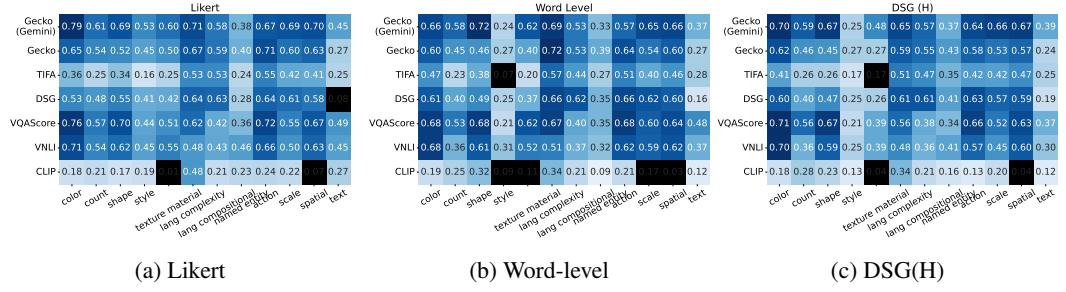


Figure 29: **Per skill results by metric.** We visualise the correlation for each skill. Where p-values are > 0.05 , we color the square black.

	Prompt:							
	A snake is on the elephant.							
	Imagen	Muse	SDXL	SD1.5	Imagen	Muse	SDXL	SD1.5
APIC:	1.	1.0	0.44	0.33	0.53	0.83	0.47	0.21
Likert:	1.	0.80	0.6	0.4	0.6	0.6	0.6	0.4
DSG (H):	1.	0.50	1.	0.5	0.92	0.8	0.71	0.25
Gecko:	0.98	0.84	0.73	0.26	0.85	0.58	0.86	0.21
DSG:	1.0	1.0	1.0	0.33	0.67	0.50	0.89	0.11
VNLI:	0.94	0.93	0.14	0.23	0.19	0.30	0.34	0.27

	Prompt:							
	There are 5 apples, two of them are yellow and two are black but none are red.							
	Imagen	Muse	SDXL	SD1.5	Imagen	Muse	SDXL	SD1.5
APIC:	0.88	0.73	0.57	0.61	1.0	1.0	1.	0.4
Likert:	0.73	0.60	0.53	0.60	0.8	0.93	1.	0.53
DSG (H):	0.80	0.53	0.5	0.40	0.75	1.0	1.	0.25
Gecko:	0.88	0.91	0.84	0.69	0.98	0.97	0.97	0.7
DSG:	1.0	1.0	0.9	0.8	1.	0.83	1.	0.17
VNLI:	0.23	0.20	0.19	0.16	0.9	0.95	0.94	0.17

	Prompt:							
	A fortune cookie that has the fortune "the best way to predict the future is to create it."							
	Imagen	Muse	SDXL	SD1.5	Imagen	Muse	SDXL	SD1.5
APIC:	0.35	0.39	0.	0.02	1.0	0.88	1.0	0.63
Likert:	0.53	0.47	0.4	0.27	0.73	0.67	0.73	0.53
DSG (H):	1.0	0.5	0.67	0.17	.83	1.0	0.83	0.83
Gecko:	0.96	0.79	0.75	0.62	0.92	0.88	0.87	0.70
DSG:	0.88	0.25	1.0	0.25	0.93	0.93	0.86	0.79
VNLI:	0.37	0.27	0.25	0.35	0.81	0.65	0.55	0.31

Figure 30: **Additional visualisations of scores from different auto-eval metrics.** We show the image generations by the four generative models given two prompts from Gecko(S), with the alignment scores predicted by human annotators and auto-eval metrics respectively.

F.7 RESULTS PER WORD FOR WL

Here we evaluate how well the Gecko metric (with the PALI/PALM-2 backend) can identify whether words are or are not grounded as rated by WL. This experiment can *only* be done for Gecko as *no other* metric gives word level annotations that can be traced back to the original words in the prompt. We note that this is much more challenging than giving an overall image rating. In order to perform this experiment, we first parse the coverage prediction to ensure we can match words in the original prompt with those in the coverage prediction. For example, if we have the original prompt ‘a red-colored dog’ and ‘a {1}[red-colored] {2}[dog]’ as the generated coverage one, we can map from the word index (e.g. ‘{1}’ to the phrase ‘red-colored’).

Gecko(R)			Gecko(S)			
	# Words evaluated	Accuracy	Error Consistency (κ)	# Words evaluated	Accuracy	Error Consistency (κ)
SD1.5	3727	75.5	0.20	2729	77.5	0.52
SDXL	3901	80.0	0.13	3304	79.5	0.34
Muse	2792	82.5	0.24	3298	82.0	0.19
Imagen	2549	76.5	0.29	3472	80.2	0.39

Table 20: **Word level results comparing Gecko to the WL annotation template.** We can see that Gecko achieves high accuracy but also that results are not simply explained by chance, as $\kappa > 0$ and in general indicates fair agreement.

We then take all word level predictions where all annotators either annotated the word as grounded or not grounded (we removed those for which a subset of annotators annotated the word as ‘unsure’). For these words, we take the ones where the coverage model predicts that it should be covered (e.g. in the example above, even if ‘a’ was always annotated as grounded, we would ignore it as it is not considered groundable by the coverage step). Given this final set of words, we look at whether the VQA prediction was accurate and compare this to whether the annotators thought that the word was grounded or not.

We report three numbers in Table 20: (1) the number of words we are left with, (2) the accuracy and (3) the error consistency κ (Geirhos et al., 2020), equation (1),(3). We report error consistency as many of the words ($\sim 90\%$) are rated as grounded. Accuracy does not account for the fact that a metric which predicts ‘grounded’ $\sim 90\%$ of the time would actually get $\sim 80\%$ accuracy by chance. Error consistency takes this into account such that $\kappa = -1$ means that two sets of results never agree, $\kappa = 0$ that the overlap is explained by chance and $\kappa = 1$ means results agree perfectly. As shown by the results, Gecko is able to predict grounding at the word level with reasonable accuracy. Moreover, results are not simply explained by chance (as $\kappa > 0$); the error consistency results indicate in general some but not substantial agreement.

G EXTENDING GECKO TO MORE MODALITIES: TEXT-TO-VIDEO GENERATION

Video evaluations are more challenging, in that there are multiple aspects that are encapsulated in the text-to-video consistency, including stylistic, temporal, semantic and overall fidelity. In order to extend our approach, we follow a similar two step process, in which question-answer pairs are generated using a few-shot prompt that outlines which aspects of the video need to be covered by those pairs. The corresponding few-shot prompt used for question-answer pair generation with sufficient coverage of the groundable words in the prompt is shown in Listing 5.

```

1 """
2 Given a video description and the groundable words in it, generate multiple-choice questions that verify if
3 the video description is correct.
4 The goal is to ask questions about entities, objects, attributes, actions, colors, spatial relations, temporal
5 relations, styles and scenes, when these are present in the description.
6 Make sure that all options are substantially different from each other and only one option can be the correct
7 one based on the description. Do not include other parts of the description as a non correct option.
8 Justify why the other options cannot be true based on the description and question. Also, make sure that the
9 question cannot be answered correctly only based on common sense and without reading the description.
10 Each generated question should be independent of the other ones and it should be able to be understood without
11 knowing the other questions; avoid referring to entities/objects/places from previous questions.
12 Finally, avoid asking very general questions, such as ‘What is in the video?’, or ‘Name a character in the
13 video’.
14 Generate the multiple-choice questions in the exact same format as the examples that follow. Do not add
15 asterisks, white spaces, or any other reformatting and explanation that deviate from the formatting of
16 the following examples.
17
18 Description:
19 A fat rabbit wearing a purple robe walking through a fantasy landscape.
20 The visual-groundable words and their scores are labelled below:
21 A {1}[fat, attribute, 1.0] {2}[rabbit, entity, 1.0] {3}[wearing a {4}[purple, color, 1.0] robe, attribute,
22 {1.0} {5}[walking, action, 1.0] through a {6}[fantasy landscape, scene, 1.0].
23 Generated questions and answers are below:
24 About {1}:
25 Q: What is the most appropriate description for the animal of the video?
26 Choices: thin, regular, slim, fat
27 A: fat
28 Justification: the rabbit in the video is fat ({1}). The options thin and slim are opposite of the attribute
29 mentioned in the description and the regular adjective checks whether it is obvious that the rabbit has
30 a weight above normal.
31 About {2}:
32 Q: Who wears a robe in the video?
33 Choices: rabbit, hare, squirrel, rat

```

```

23 A: rabbit
24 Justification: the rabbit is the animal that wears a robe in the video ((2)). Hare is an animal very similar
   to rabbit, and the other two options (squirrel and rat) are also similar but not true according to the
   description.
25 About {3}:
26 Q: What is the rabbit wearing in the video?
27 Choices: nothing, dress, robe, jumpsuit
28 A: robe
29 Justification: the rabbit is wearing a robe ((3)). Nothing is what normally an animal is wearing, and the
   options dress and jumpsuit are similar to the robe but not true according to the description.
30 About {4}:
31 Q: What is the color of the clothing that the rabbit wears in the video?
32 Choices: purple, blue, pink, green
33 A: purple
34 Justification: the rabbit is wearing a purple robe ((4)). the options blue, pink and green are colors similar
   to purple.
35 About {6}:
36 Q: What is the rabbit doing in the video?
37 Choices: running, walking, standing, jumping
38 A: walking
39 Justification: the rabbit is walking through a fantasy landscape ((5), (6)). The options running and standing
   are similar to walking, and jumping is an action that could be performed by a rabbit, but not true
   according to the description.
40 About {7}:
41 Q: Where is the video taking place?
42 Choices: fields, countryside, fantasy landscape, mountains
43 A: fantasy landscape
44 Justification: the rabbit is walking through a fantasy landscape ((6)). The options fields, countryside, and
   mountains are different types of landscapes, but they are real-world scenes instead of fantasy ones.
45
46 Description:
47 A beautiful coastal beach in spring, waves lapping on sand by Hokusai, in the style of Ukiyo
48 The visual-groundable words and their scores are labelled below:
49 A {1}[beautiful coastal beach, scene, 1.0] {2}[in spring, temporal relation, 1.0], {3}[waves, scene, 1.0] {4}
   [lapping, action, 1.0] {5}[on sand, spatial relation, 1.0] {6}[by Hokusai, style, 1.0], {7}[in the style
   of Ukiyo, style, 1.0]
50 Generated questions and answers are below:
51 About {1}:
52 Q: Where is the video taking place?
53 Choices: cliffs, harbor, coastal park, coastal beach
54 A: coast beach
55 Justification: the main scene is a beautiful coastal beach ((1)). The options cliffs, harbor, and coastal park
   are similar to coastal beach but not true according to the description.
56 About {2}:
57 Q: Which season is most likely during the video?
58 Choices: spring, summer, autumn, winter
59 A: spring
60 Justification: the video shows a coastal beach in spring ((2)). The options summer, autumn and winter are
   other seasons that are not true according to the description.
61 About {3}:
62 Q: What is the level of movement of the sea during the video?
63 Choices: calm, wavy, slightly moving, ripply
64 A: wavy
65 Justification: the sea is wavy ((3)). The options calm, slightly moving, and ripply are different levels of
   movement of the sea and they are all different enough from wavy.
66 About {4}:
67 Q: What is the movement of the sea during the video?
68 Choices: gentle waves are coming to the shore, there is a tide, waves are lapping on the shore, there are sea
   ripples
69 A: waves are lapping on the shore
70 Justification: the sea is lapping on the shore ((4)). The other provided options are either of less intesity (
   gentle waves are coming to the shore, there are sea ripples) or the exact opposite (there is a tide).
71 About {5}:
72 Q: Where does the sea move to during the video?
73 Choices: sand, rocks, cliffs, pebbles
74 A: sand
75 Justification: the waves are lapping on sand ((5)). The options pebbles, rocks, and cliffs are different types
   of ground typically by the sea and have different levels of solidity.
76 About {6}:
77 Q: Whose artist is the theme of the scene similar to?
78 Choices: Utamaro, Hokusai, Hiroshige, Yoshitoshi
79 A: Hokusai
80 Justification: the theme of the scene resembles a painting of Hokusai. The other options are other Japanese
   artists that are similar to Hokusai.
81 About {7}:
82 Q: Which Japanese painting style is most similar to the video?
83 Choices: Ukiyo, Nihonga, Sumi, ink calligraphy
84 A: Ukiyo
85 Justification: the video scene is in the style of Ukiyo ((7)). The other options are other types of Japanese
   painting styles that are not similar to the video according to the description.
86
87 Description:
88 ...
89 """

```

Listing 5: Sample LLM template for generating QAs with coverage for videos.

Evaluation setup. As described in Section 6.5, we choose a prompt set that is appropriate for measuring alignment between the description and the video and a set of text-to-video models to collect human annotations using different templates. We consider a subset of 94 prompts from the

Model 1	Model 2	'GT'	Gecko	VideoCLIP	VQAScore
Lumiere	Phenaki	>	>	--	--
Lumiere	WALT	>	>	--	>
Phenaki	WALT	--	--	--	>

Figure 31: **Comparing model ordering obtained from humans and auto-eval metrics on VBench.** We show the ‘GT’ human ordering and the predicted ones for auto-eval metrics. < means Model 1 < Model 2, > Model 1 > Model 2 and -- that no significant relation was found.

VBench benchmark (Huang et al., 2024b) manually tagged with “overall consistency” for evaluating overall text-to-video alignment by the curators of the benchmark. We compare the following text-to-video models: Lumiere (Bar-Tal et al., 2024), Phenaki (Villegas et al., 2022) and WALT (Gupta et al., 2023). For human evaluation, we consider both absolute comparison templates (i.e., Likert and Word Level) and a template for relative pairwise comparisons (i.e., SxS), and for automatic evaluation, we again benchmark two types of auto-eval metrics: contrastive models (i.e., VideoCLIP; Xu et al. 2021) and VQA-based metrics. For VQA-based metrics, since there is no prior work on text-to-video generation, we extend the VQAScore and our fine-grained Gecko metric on the video domain using Gemini Flash as our video question answering model, which can process long context multimodal inputs.

Model ordering. In addition to the Pair-wise instance scoring and point-wise instance scoring results presented in Table 7, we also compare model rankings per metric in Figure 31 using the Wilcoxon signed-rank test for all pairs of models. To obtain the ground-truth (GT) model ordering, we average human preferences across the three templates (Likert, WL, SxS) via majority voting and consider valid model rankings only when pairwise differences are significant ($p < 0.05$). We find that the model ranking provided by Gecko agrees with the human rankings for all three model comparisons. In contrast, VideoCLIP, which is often used as part of tool use for text-to-video evaluation (Huang et al., 2024b; Liu et al., 2024), does not provide *any* information about the relative performance of video models in terms of alignment. VQAScore agrees with the ground truth comparisons only for one model pair (Lumiere vs WALT).