CROC: Evaluating and Training T2I Metrics with Pseudo- and Human-Labeled Contrastive Robustness Checks

Christoph Leiter[1,*,+] and **Yuki M. Asano**[2] and **Margret Keuper**[1,3] and **Steffen Eger**[1,2,+]
[1]University of Mannheim, [2]University of Technology Nuremberg,
[3]Max Planck Institute for Informatics, Saarland Informatics Campus
[*]christoph.leiter@uni-mannheim.de
[+]NLLG (https://nl2g.github.io/)

Abstract

The assessment of evaluation metrics (meta-evaluation) is crucial for determining the suitability of existing metrics in text-to-image (T2I) generation tasks. Human-based meta-evaluation is costly and time-intensive, and automated alternatives are scarce. We address this gap and propose CROC: a scalable framework for *automated **C**ontrastive **Ro**bustness **C**hecks* that systematically probes and quantifies metric robustness by synthesizing contrastive test cases across a comprehensive taxonomy of image properties. With CROC, we generate a pseudo-labeled dataset (CROC^{syn}) of over 1 million contrastive prompt–image pairs to enable a fine-grained comparison of evaluation metrics. We also use this dataset to train CROCScore, a new metric that achieves state-of-the-art performance among open-source methods, demonstrating an additional key application of our framework. To complement this dataset, we introduce a human-supervised benchmark (CROC^{hum}) targeting especially challenging categories. Our results highlight robustness issues in existing metrics: for example, many fail on prompts involving negation, and all tested open-source metrics fail on at least 24% of cases involving correct identification of body parts.[1]

1 Introduction

The multimodal task of text-to-image (T2I) generation has advanced rapidly: from text-conditioned GANs (e.g., Reed et al., 2016) to state-of-the-art architectures like diffusion transformers (e.g., Esser et al., 2024). T2I models produce images conditioned on textual prompts including desired styles, actions, relationships, or attributes.

Evaluating T2I outputs is challenging because multiple images can validly satisfy a prompt, making judgments subjective. Since human evaluation is slow and expensive, automatic metrics are used to assess various quality dimensions (Hartwig et al., 2025). These metrics rank outputs, benchmark systems, filter training data, and guide fine-tuning and re-ranking (Hartwig et al., 2025; Leiter et al., 2024). Although numerous evaluation metrics have been proposed, their meta-evaluation (evaluating the metrics themselves) is less researched and typically relies on correlations with human labels (e.g., Hu et al., 2023; Cho et al., 2024; Wiles et al., 2025). However, this strategy has several shortcomings: (1) it is costly and time-consuming; (2) it offers limited coverage of image properties; (3) older datasets pose a risk of data leakage; and (4) certain correlation measures may unfairly favor metrics (Deutsch et al., 2023).

Previous work in machine translation has demonstrated the utility of generated test cases (e.g. Sai et al., 2021; Karpinska et al., 2022; Chen and Eger, 2023) for evaluating the robustness of evaluation metrics. Motivated by their findings, we investigate the viability of automatically generating meta-evaluation datasets for T2I metrics. In particular, we focus on assessing their **robustness** by identifying which T2I properties and content types each metric handles well and which it does not. To this end, we propose **CROC**: automated **C**ontrastive **Ro**bustness **C**hecks (see Figure 1).

Each CROC sample tests a specific property for text-image alignment (e.g., "can metrics detect the color *white*?"). It contains an original text (e.g., "a white sheep"), a contrastive text (e.g., "a blue sheep"), and one or more images per text. Texts are generated by LLMs and images by diffusion models. Under accurate generation, matching text-image pairs should, by definition, score higher than mismatched ones, thereby minimizing human supervision. Cases that models cannot re-

[1]Our framework is available at https://github.com/Gringham/CROC/tree/main.

Transactions of the Association for Computational Linguistics, vol. 14, pp. 1119–1139, 2026. https://doi.org/10.1162/TACL.a.699
Action Editor: Deyi Xiong. Submission batch: 5/2025; Revision batch: 10/2025; Published 6/2026.

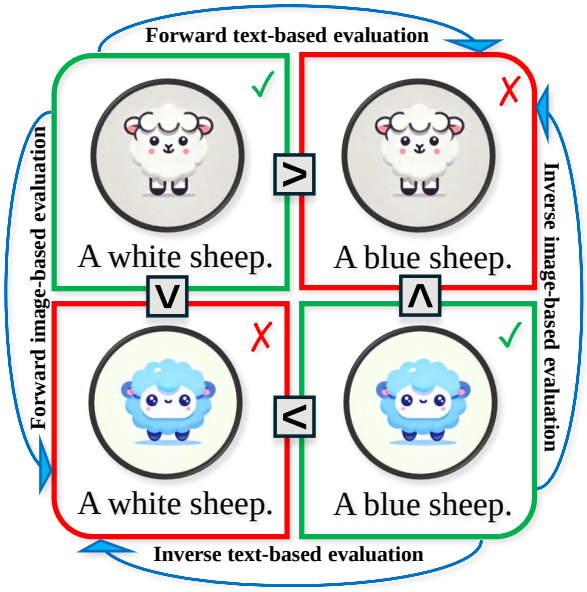

Figure 1: Contrastive evaluation of T2I metrics. Given a text-to-image metric that assigns quality scores to text-image pairs, matching pairs (green) should receive higher scores than non-matching pairs (red). In *text-based* evaluations, the original text is replaced with a contrastive text, while in *image-based* evaluations, the original image is replaced. In *inverse* evaluations, the matching pair is defined by the contrastive text and image used in the *forward* evaluations.

liably generate still require human oversight.

Based on a comprehensive taxonomy of properties, domains, and entities, we construct a large pseudo-labeled (synthetic ground-truth through contrastive generation) dataset, CROC$^{\text{syn}}$, and benchmark recent metrics on it. We also create a human-supervised set, CROC$^{\text{hum}}$, focused on particularly challenging generation categories. As a further use case, we train a new metric, **CROCScore**, on CROC$^{\text{syn}}$. In summary, we make the following contributions:

- ✓ **Meta-evaluation framework:** We introduce CROC, automated **C**ontrastive **Ro**bustness **C**hecks, an adaptable framework for T2I metric meta-evaluation. To our knowledge, CROC is the first metric meta-evaluation framework that *jointly* uses LLM-generated, property-wise perturbations and pairwise contrastive checks to verify that metric scores move in the expected direction, reducing human labeling and enabling fine-grained robustness analysis.
- ✓ **Datasets (CROC$^{\text{syn}}$ and CROC$^{\text{hum}}$):** We construct the large-scale, pseudo-labeled dataset CROC$^{\text{syn}}$, providing $>$ 1M prompt–image pairs for training and metric comparison. To validate its utility as a benchmark, we conduct a human evaluation and show that evaluated metrics still fall short of human accuracy. Further, we curate CROC$^{\text{hum}}$, a human-supervised core set targeting cases that current T2I models struggle to generate.
- ✓ **CROCScore:** We use CROC$^{\text{syn}}$ to train a new metric CROCScore, achieving state-of-the-art performance among open-source metrics on CROC$^{\text{hum}}$, GenAi-Bench (Li et al., 2024), Winoground (Thrush et al., 2022) and TIFA (Hu et al., 2023). This demonstrates another benefit of the pseudo-labeled data.
- ✓ **Benchmark and robustness findings:** We evaluate 6 metrics on CROC$^{\text{syn}}$ and find that, while VQAScore (Lin et al., 2024) performs best in most evaluation settings, embedding-based metrics like AlignScore (Saxon et al., 2024) and BLIP2-ITM (Li et al., 2023) outperform it in some cases, consistent with a previous meta-evaluation (Saxon et al., 2024). On the more challenging CROC$^{\text{hum}}$, performance gaps widen, particularly for image-based evaluation. On this dataset, we additionally evaluate VQAScore with GPT-4o backend (best overall) and our metric CROCScore (best among open-source metrics). Fine-grained analyses reveal robustness issues in specific properties. For example, many metrics mishandle negation and all open-source metrics confuse body parts in ca. 25% of cases.

2 Related Work

This section briefly reviews prior work on T2I metrics and their fine-grained meta-evaluation.

T2I evaluation metrics The survey by Hartwig et al. (2025) groups T2I metrics into two families: (1) *embedding-based* approaches, e.g., CLIPScore (Hessel et al., 2021), measure text-image similarity in a shared embedding space. (2) *Content-based* approaches, e.g., BVQA (Huang et al., 2023), decompose the evaluation process, for example, by reformulating the input text into questions (VQA-based) that a multimodal LLM answers. In this work, we distinguish *tuned* metrics

as a third family: these metrics, e.g., PickScore (Kirstain et al., 2023), are trained on human quality judgments. Although most tuned methods build on embedding-based approaches, their training signal and typical failure modes differ.

Several works use these metrics to benchmark *T2I models* with fine-grained prompt suites (e.g., DrawBench and DALL-EVAL (Lee et al., 2023; Cho et al., 2023)). They share our emphasis on property-wise prompts, but their goal is model evaluation. Our goal is to *meta-evaluate the metrics* themselves using contrastive checks.

Metric meta-evaluation Datasets such as TIFA (Hu et al., 2023) provide sample-level, human-labeled scores for text-image pairs. Recent benchmarks, including Gecko (Wiles et al., 2025) and GenAI-Bench (Li et al., 2024), extend this by providing human-labeled scores over diverse sets of images, enabling fine-grained analysis of metric behavior. Our approach instead uses *generated, property-wise contrastive examples*. For each text, we construct matched and contrasted pairs that differ on a single property, enabling targeted robustness analyses while minimizing manual labeling because labels are implied by construction.

T2IScoreScore (Saxon et al., 2024) evaluates whether metric scores decrease along controlled degradations that are injected into aligned pairs. Our property-based perturbations target fine-grained behavior across specific properties, which is enabled through the contrastive design.

Winoground (Thrush et al., 2022) and follow-up benchmarks like Winoground-T2I (Zhu et al., 2023) also use contrastive setups to probe compositionality of multimodal models. The latter includes a small-scale meta-evaluation via correlation with human Likert scores. In contrast to these, we set a clear focus on the fine-grained, property-wise meta-evaluation of T2I metric robustness through generated, pseudo-labeled data.

Compared to the above benchmarks, CROC$^{\text{hum}}$ also contains more human-verified text-image pairs and CROC$^{\text{syn}}$ is on a much larger scale (see Table 1). We also introduce CROCScore, a strong open-source metric trained on CROC$^{\text{syn}}$.

More broadly, our work relates to metrics that exploit contrasts in model representations (e.g., Wang et al., 2025) and to contrastive pre-training of joint text-image encoders (e.g., Jia et al., 2021). We differ by using a fine-grained generated dataset to meta-evaluate and fine-tune T2I metrics.

Benchmark	Texts	T-I Pairs
TIFA	160	800
Gecko	2k	8k
GenAI-Bench	1,6k	12,8k
T2IScoreScore	165 err. graphs	2,8k
Winoground	800	1,6k
Win.T2I (S-100)	200	800
CROC$^{\text{hum}}$	560	∼23,8k
CROC$^{\text{syn}}$	∼220k	>1M

Table 1: Comparison of related dataset sizes.

3 Methodology

In this section, we describe the evaluation setup and generation framework of CROC and the training approach of CROCScore.

3.1 Evaluation directions

A T2I model $G(T) = I$ generates an image I based on an image generation prompt T. Accordingly, a T2I metric $M(T, I) = s$ assigns a score $s \in \mathbb{R}$ that indicates how well I follows T (i.e., their alignment, higher is better). Further, we refer to a T-I pair where I correctly follows T as a **matching pair**. Likewise, a pair where I does not follow T is referred to as a **contrast pair**.

Inspired by related work in the domain of MT evaluation (Chen and Eger, 2023), we meta-evaluate the robustness of T2I metrics in a contrastive setup (see Figure 1). That means, we test whether the condition

$$M(\text{matching pair}) > M(\text{contrast pair}) \quad (1)$$

is correctly fulfilled. Specifically, we design examples with two contrasting prompts T_O and T_C, as well as images I_O and I_C (**O** for **O**riginal and **C** for **C**ontrast). Here, T_O-I_O and T_C-I_C are matching pairs, while T_O-I_C and T_C-I_O are contrast pairs. For example, in Figure 1 the green cells show matching T-I pairs (e.g., the text "A white sheep" and the image of a white sheep) that should receive higher metric scores than the red cells that show contrast pairs (e.g., the text "A white sheep" and the image of a blue sheep). Notably, by controlling T_O and T_C so that they differ solely in specific features (such as color or position), we facilitate fine-grained robustness tests. There are four possible evaluation directions:

1. Image-based evaluations compare a matching pair to a contrast pair that changes the image

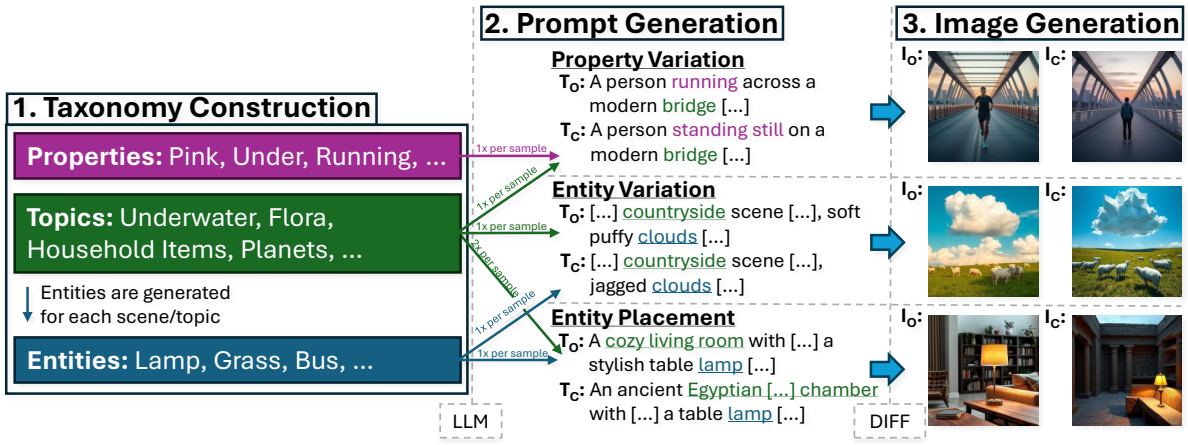

Figure 2: Overview of the CROC^{syn} data generation process. First, we construct our taxonomy based on related work and with LLM support. The image shows example categories. Second, we pass selected categories of the taxonomy to an LLM (e.g., "running" and "bridges and infrastructure") to generate the original text T_O and the contrast text T_C for one sample. Then these are passed to a diffusion model to generate the original image I_O and the contrast image I_C.

(column-wise comparison in Figure 1). For example, we can evaluate whether $M(T_O, I_O) > M(T_O, I_C)$ is true. This evaluation corresponds to the question: "Which image was more likely generated from T_O?"

2. Text-based evaluations compare a matching pair to a contrast pair that changes the text (row-wise comparison in Figure 1). For example, we can evaluate whether $M(T_O, I_O) > M(T_C, I_O)$ is true. This evaluation corresponds to the question: "Which prompt is more likely to have generated I_O?" This evaluation type is related to image captioning. The difference is that the images were generated from the prompt and not vice versa.

3. + 4. In our prompt creation process (see §3.2), we create T_C by changing an *original* prompt T_O. Therefore, T_C is more likely to feature unusual scenarios. When the matching pair of the comparison is T_O-I_O, we refer to the evaluation as **forward evaluation**. In contrast, if the matching pair is T_C-I_C, we refer to the evaluation as **inverse evaluation**.

3.2 Contrastive Dataset Generation

We propose two complementary dataset-construction pipelines: (1) a *pseudo-labeled* pipeline that generates **CROC^{syn}**, a large-scale dataset designed to broadly cover text-to-image (T2I) alignment use cases; and (2) a *human-supervised* pipeline that curates **CROC^{hum}**, a suite of evaluation tests targeting especially challenging properties of image generation.

Pseudo-labeled data generation The *pseudo-labeled* pipeline (see Figure 2) creates comprehensive contrastive samples (T_O, T_C, I_O, I_C) in three steps: (1) *taxonomy construction*, (2) *text generation* and (3) *image generation*. The taxonomy is hierarchical and contains 64 image properties (e.g., relations, attributes, colors), 51 topics (e.g., nature, people, architecture) and 158 entities (e.g., eagle, bridge, bus). Each element is paired with a short textual description. The taxonomy of properties, example topics and example entities is shown in Appendix A. To construct this taxonomy, we first consolidated initial properties from Wiles et al. (2025), Hartwig et al. (2025), Foote (2018) and Chen and Eger (2023). Then, we manually define and interactively generate (with GPT-4o) a broad hierarchy of further subclasses, topics and respective descriptions. Entities are created for every topic leaf node, e.g., *grass* for *flora* and *deer* for *fauna*. Based on this taxonomy, we create three types of test prompts (one that is based on properties and two that are based on entities):

1. Property variation: Here, T_O features a property selected from the taxonomy and T_C strongly changes this property. Referring back to Figure 1, the property *white* in T_O "A white sheep" is changed to *blue*. Property variation is the main test type in our robustness evaluation, as it allows for a fine-grained check of metric capabilities.

2. Entity placement: In an initial experiment, we generated contrastive text-image pairs from T2ICompBench (Huang et al., 2023) and found

that human annotators can rate inverse evaluation directions (with unexpected prompts) better than metrics (see Table 8). Motivated by this, we evaluate the capability of metrics to correctly rate unexpected matching text-image pairs higher than expected contrast pairs[2]. Here, T_O describes an entity with its source topic, i.e., where it naturally occurs. For example, this could be a *sheep* on a *field*. T_C then places the entity in an unusual topic, e.g., a *sheep* in a *city*.

3. Entity variation: As another way of testing the possible unexpectedness bias of T2I metrics, we generate T_O such that an entity is described naturally. Then, T_C describes the entity with an altered description. For example, this could be "a sheep with two eyes" vs. "a sheep with three eyes".

In the final step, the images I_O and I_C are generated for T_O and T_C, respectively.

Practical considerations: CROC$^{\text{syn}}$ In practice, we set $I_O = G(T_O)$ and $I_C = G(T_C)$. However, just like T2I evaluation metrics, the text generation models and T2I models are imperfect and may not correctly follow the prompt. This presents a chicken-and-egg problem: normally, T2I evaluation metrics evaluate T2I models. In our setup, we meta-evaluate T2I evaluation metrics **with** T2I models. T2I evaluation metrics are not perfect and their quality can be quantified in terms of human correlation. Similarly, our automatic evaluation setup for T2I evaluation metrics is not perfect. Therefore, (1) we measure its quality with human annotations (see §5.2), and (2) we compare the results with CROC$^{\text{hum}}$ (see §3.2). Further, (3) for each prompt (T_O resp. T_C), we generate $n > 1$ images ($I_O^{1,\dots,n}$ resp. $I_C^{1,\dots,n}$). Then, for the **forward text-based** setups, we evaluate:

$$j^* = \underset{i=1,\dots,n}{\operatorname{argmax}} M(T_O, I_O^i),$$
$$M(T_O, I^{j^*}) > M(T_C, I^{j^*}) \quad (2)$$

where i and j^* are indices for the images. That means, if at least one image $I_O^{j^*}$ follows T_O correctly, the metric should correctly pick that image (assign the highest score to $M(T_O, I^{j^*})$), otherwise it is an error of the metric and not an error of the setup (see Appendix C for a detailed example).

For $n = 5$ images, this setup raises the accuracy of a random-scoring baseline to $\frac{5}{6} \approx 83.\overline{3}\%$, since among $n+1$ i.i.d. draws (i.e., 5 options from the matching side, given through the maximum, and 1 option from the non-matching side) each is equally likely to be the maximum, giving $1 - \frac{1}{n+1}$. We use this later to scale the accuracies for comparability. For the **forward image-based setup**, we compare:

$$\max_{i=1,\dots,n} M(T_O, I_O^i) > \max_{j=1,\dots,n} M(T_O, I_C^j) \quad (3)$$

That means, the I_O^i that matches T_O best should be rated higher than the I_C^j that matches T_O best. Here, the method is different because for text-based evaluation we have many images per prompt. On the other hand, for image generation, we often do not have multiple prompts per image because the prompt generation is less restricted and prompts can contain different content even though they have the same topic and property. For **inverse** evaluations, T_O is swapped with T_C and I_O^i is swapped with I_C^i (see Appendix B). For this setup, the accuracy of a random baseline is 50%. While there is no guarantee that every single sample is generated correctly, these considerations ensure that evaluation on our dataset provides meaningful fine-grained comparisons between metrics. Another circularity that may occur is that, just like LLM-based evaluators can have biases that favor their own generations (Panickssery et al., 2024), the mLLM-based metrics that we are testing may be biased if the data-generation LLMs are related to the metric backbone. We argue that this effect should be partly mitigated, because bias in a contrastive setup would affect the rating of both, matching and non-matching text-image pairs.[3]

Human-supervised data generation Supplementary to the pseudo-labeled pipeline of CROC$^{\text{syn}}$, we construct the human-supervised dataset, CROC$^{\text{hum}}$. It addresses selected categories that are especially challenging for T2I models. These categories are critical because evaluation metrics should remain reliable when T2I models fail. Since learned metrics often differ from T2I models in architecture and training objectives, it is not obvious that they fail on the same instances. CROC$^{\text{hum}}$ allows us to probe

[2]This is to some degree also inherent to *property variation*, because our prompt generation process prompts an LLM to first construct the original text and then change something. But due to the large variety of tested properties, both texts may be unexpected.

[3]As a further mitigation, in our experiments we only select generation LLMs from different model families than the metric-backbones. Also, we do not evaluate CROCScore on the dataset it was trained on (CROC$^{\text{syn}}$).

where metrics and the models they evaluate share weaknesses. It is built in a five-step process: (1) *category selection*, (2) *template-based prompt construction*, (3) *grammar check*, (4) *image generation*, and (5) *human validation*.

In *step one*, we *choose eight challenging categories* (see Appendix D). The categories *body parts* and *parts of things* are motivated by persistent difficulties many T2I models have with rendering hands (e.g., Narasimhaswamy et al., 2024). Further, the categories *counting*, *shapes*, *size relation*, *spatial relation*, *action* and *negation* were already included in prior correlation-based benchmarks (e.g., Wiles et al., 2025; Li et al., 2024). We differ in our contrastive setup and in our prompt construction.

In *step two*, we generate original prompts T_O and contrast prompts T_C for each selected category. For action, body parts and parts of things, we interactively create T_O and T_C with GPT-4o(OpenAI et al., 2024). For example, we use colors to change a highlighted body part between T_O (e.g., *A red ring finger*) and T_C (e.g., *A red thumb*). For the remaining five categories, we randomly select one or two entities from the CROC$^{\text{syn}}$ taxonomy and one property from a predefined list, for example, the entities *person* and *car*, and the property *left of* for spatial relations. We then fill fixed templates to form simple prompts such as "a person left of a car" or "a person and no car" and use GPT-4o to verify the grammar (*step three*). A detailed example for *body parts* is described in Appendix C (example 2). Once all prompts are generated, we generate I_O and I_C (*step 4*). Negation is a special category, because we can use a trick to generate high quality I_C images: instead of generating an image of T_C ="A and not B", we simply generate an image of the alternative prompt "A" that does not contain the negation, but still compare T_C during the evaluation.

In the *final step*, we perform human verification of every text-image pair. To facilitate simple and fast verification of a large number of examples, we (1) construct short prompts[4], (2) generate a large number of images (in our experiments $n = 100$) per prompt and (3) verify by filtering, i.e., annotators see the prompt and all images in a scrollable list (e.g., a file explorer) and remove all images that do not fit the respective text. The exact annotation setup is described in §4.

[4]In CROC$^{\text{syn}}$, we opt for long, descriptive prompts, to increase the quality and diversity of the generated images. In contrast, in CROC$^{\text{hum}}$, we opt for short prompts, which might reduce image quality, but allow human annotators to verify the T2I alignment of the images with more certainty in a shorter amount of time.

Practical considerations: human-sup. dataset Due to the human verification through filtering, CROC$^{\text{hum}}$ has strong prompt–image alignment, so it is unnecessary to apply the same selection strategies used for CROC$^{\text{syn}}$. However, CROC$^{\text{hum}}$ contains a varying number of images per prompt (varying i and j). Therefore, we treat all prompts equally by averaging the sample-wise performance. For the forward text-based evaluation, for each prompt T_O, we compute the ratio of cases where $M(T_O, I_O^i) > M(T_C, I_O^i)$ across all i. For the forward image-based evaluation, we compute the ratio of cases where $M(T_O, I_O^i) > M(T_O, I_C^j)$ across all i, j. The final performance for a category is then obtained by averaging these ratios across prompts. For inverse evaluation, we swap O and C accordingly.

CROCScore The basis for CROCScore is inspired by VQAScore (Lin et al., 2024), which prompts a multimodal LLM (mLLM) to answer a simple question like "Does this image show {prompt}" and uses the probability of the answer "Yes" as a metric score. We extend this approach and ask a question like "Does this image show the following content:'{prompt}'? Answer with Yes or No.". Then the score is computed as $p(\text{Yes}) - p(\text{No})$. This matches our contrastive setup, in which non-matching pairs should have a high No-probability (and a low Yes-probability). During training, for each sample in our dataset, we randomly select either the matching or contrast pair and further fine-tune an mLLM to generate *Yes* for matching pairs and *No* for contrast pairs.

4 Experiment Setup

Models and infrastructure We run computations on a Slurm cluster with Nvidia A40 and A100 graphics cards (Table 7 compares metric runtimes). For pseudo-labeled prompt generation, we use the two models, DeepSeek-R1-Distill-Qwen-14B (DeepSeek-AI, 2025) and QwQ-32B (Qwen Team, 2025), because of their strong performance on text generation leaderboards and relative cost-effectiveness. The runtime is ca. three hours per model (2 resp. 4 GPUs). For image generation, we use FLUX.1-schnell

(Black Forest Labs, 2024) and Stable-diffusion-3.5-large-turbo (StabilityAi, 2024a) that are fine-tuned to require fewer generation steps (with the trade-off of lower generation quality) to ease the computational requirements. The image generation took an average of ca. 5h on 120 GPUs.

T2I metrics We compare several classes of T2I metrics (see §2 for details on the metrics and Table 7 for their configuration). We use the **embedding-based metrics** CLIPScore (Hessel et al., 2021) and AlignScore (Saxon et al., 2024). We also evaluate BLIP2-ITM, which was trained with an image-text matching objective (Li et al., 2023). Further, we test the **trained** metric PickScore (Kirstain et al., 2023) and the more recent **visual question-answering (VQA) metrics** VQAScore (Lin et al., 2024) and BVQA (Huang et al., 2023). On the human-supervised dataset, we also evaluate VQAScore with a closed-source GPT-4o backend. We only evaluate it here because of its costly inference (ca. $150 on the supervised dataset).

The CROC Datasets For **CROC$^{\text{syn}}$**, we generate up to 20 prompt-contrast pairs per *input description* (five for each LLM-T2I model combination).[5] As we described earlier, we place each prompt type (property variation, entity variation and entity placement) with one of 51 topics (e.g., medieval, landscapes). Thus, the *input description* can be one of three types: (1) The first type is a property variation pair; for these, we generate all 51 combinations. (2) The second type is an entity variation sample; here, we choose ten random topics for each entity. (3) The third type is an entity placement sample; in this case, we select ten random combinations of topic and alternative topic for each entity. For example, for entity placement, we choose ten setups that place the entity "eagle" from a natural topic in a randomly chosen alternative topic. In total, CROC$^{\text{syn}}$ contains 25,693 unique entity variation, 28,723 unique entity placement and 56,984 unique property variation prompt-contrast pairs. Further, we generate $n = 5$ different images per prompt to increase the robustness of our setup. Appendix C shows a full example for one property. The prompts in this setup have a maximum token count of 180. The prompting guides that we use in the prompt templates suggest detailed prompts to increase the quality of generated images. However, the generated prompts can be long and difficult to process in human experiments. Also, CLIP-based models like CLIPScore have a disadvantage on long prompts with more than 77 tokens, where they are cut off (see Figure 9 for an analysis).

For **CROC$^{\text{hum}}$**, we create 280 prompts and respective contrast prompts (four test categories with 50 prompts and four with 20 prompts). For each prompt and contrast prompt, we create 50 images with the Stable Diffusion model and 50 images with the Flux model. This means, in total, we create 56,000 text-image pairs. In Appendix C we show a full example of the human-supervised data creation. As described in §3.2, we manually filter out images that do not fulfill the category we are evaluating. For example, for the category "counting" and a prompt "four fingers", we manually remove images that do not show exactly four fingers. During this human verification, we keep images with visual distortions if the matching property is fulfilled and the non-matching one is not fulfilled. For example, for "part of things" prompts about *calculators*, we accept images of calculators that show incorrect button ordering if the contrasting prompts "a blue button" vs. "a blue display" are fulfilled. For the prompt "a blue 9-button", we only accept images that verifiably indicate the button through text or position. For spatial relations, like "left of", we consider the position of the viewer.

As we deliberately choose hard cases for image generation, some prompts do not generate any valid image. For these, we augment our data to have at least three valid images by interactively generating matching images with GPT-4o (for body parts) and GPT5 (OpenAI, 2025) (others) in multi-turn conversations. After filtering, the dataset contains 12,090 Flux and 11,497 Stable Diffusion images out of 56,000 images (+245 GPT images). This means, over half of the images are removed, due to unsuccessful generation for these challenging categories. The filtering was conducted by one person with fluent English skills in approximately 25 hours. While this may introduce a degree of subjectivity, for most prompts it is resolved through the contrastive setup: even if a match is subjective, it will be more matching than the contrast sample. We further verify the validity of CROC$^{\text{hum}}$ with a **human evaluation** in §5.2. Here, 3 annotators annotate the same 480 exam-

[5] If the output is not in a valid format, we drop it. We choose JSON for simple output parsing.

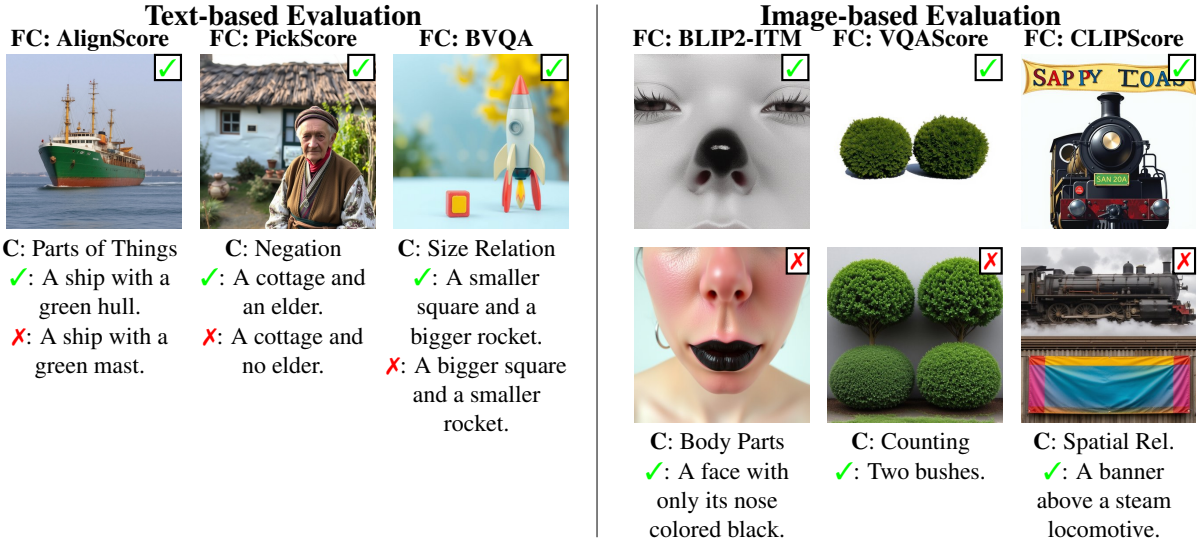

Figure 3: Selected metric failure-cases CROC$^{\text{hum}}$. *FC* shows a metric that failed on this example and *C* shows the category of the example. ✓ indicates the matching text-image pair. For text-based evaluation, the metric falsely rates the text with ✗ higher than the text with ✓. For image-based evaluation, the metric falsely rates the image with ✗ higher than the image with ✓.

ples (60 per category, stratified across evaluation direction) and have to select matching text-image pairs over non-matching ones. Notably, because we use 280 prompts and contrast prompts these 480 samples cover at least one original or contrasting prompt of every data sample.

Human evaluation of CROC$^{\text{syn}}$ To assess the quality of the pseudo-labeled generation process, we conduct a human study on CROC$^{\text{syn}}$ using Prolific annotators.[6] For every annotation, workers are presented with one text (T_O or T_C) and five images ($I_O^{\{1,\dots,5\}}$ or $I_C^{\{1,\dots,5\}}$). Their task is to assign a continuous 1–5 alignment rating (slider with two decimal digits) that indicates how well the best matching image is aligned with the text (1 indicates no alignment, while 5 indicates perfect alignment). We choose continuous scores to reduce the amount of ties among scores for low or high quality images. This setup mirrors the max-over-images operation used in our image-based aggregation. We select 500 samples that are stratified across properties, entities, text- and image-generation models. For every sample, we evaluate all four text-image directions: (1) T_O–$I_O^{\{1,\dots,5\}}$, (2) T_C–$I_C^{\{1,\dots,5\}}$, (3) T_O–$I_C^{\{1,\dots,5\}}$, and (4) T_C–$I_O^{\{1,\dots,5\}}$, yielding 2,000 text-image pairs (each consists of one text and five candidate images). We split the 2,000 pairs into 26 batches. Each text-image pair receives scores from three Prolific workers (amounting to 77 annotators, because one annotator annotated two batches). Within each batch, the four image-pairs (one per direction) of the same sample are shown in random order. In addition, a subset of 380 pairs is labeled by three in-house annotators and analyzed separately. For aggregation, we normalize ratings per annotator via z-scoring (subtract the mean and divide by standard deviation) and average the normalized scores to obtain a single score for each item–direction. We then compute the four evaluation directions identically to the automatic metrics. The total annotation cost is ca. $1025 with annotation times ranging between 20min. up to 1h20min. per batch. To increase annotation quality, ca. every 12 samples we display a simple attention check asking to select a score from a certain range hidden within the text.

Training and evaluating CROCScore To train CROCScore, we fine-tune the vision encoder and LoRA of phi4-multimodal-instruct (Microsoft, 2025) on two H100 GPUs. We first select a subset of CROC$^{\text{syn}}$ that includes three data samples for every property variation-topic combination and a total of 200 entity variation prompts. We do not include entity placement, as our analysis in §5.1 shows that metrics already perform strongly on it. Further, we restrict the dataset to contain only

[6] www.prolific.com. We target fluent English speakers residing in the UK or USA and, for ∼80% of tasks, require at least a master's degree.

P	Embedding-based Align	BLIP2	CLIP	Tuned Pick	VQA-based BVQA	VQAS
Forward Text-Based						
EV	0.022	-0.037	-0.308	0.039	-0.017	**0.552**
EP	0.922	0.883	-0.214	0.796	0.868	**0.968**
PV	0.044	0.041	-0.309	0.004	-0.034	**0.442**
Inverse Text-Based						
EV	**-0.032**	-0.103	-0.383	-0.099	-0.261	-0.164
EP	**0.904**	0.874	-0.201	0.737	0.505	0.841
PV	**0.186**	0.019	-0.281	-0.034	-0.100	-0.043
Forward Image-Based						
EV	0.541	**0.585**	0.402	0.564	0.438	0.580
EP	0.958	**0.959**	0.662	0.906	0.872	0.946
PV	0.700	0.706	0.456	0.637	0.641	**0.720**
Inverse Image-Based						
EV	0.622	**0.657**	0.304	0.529	0.554	0.642
EP	0.979	0.965	0.589	0.918	0.949	**0.984**
PV	0.659	0.684	0.372	0.564	0.603	**0.733**

Table 2: Scaled metric accuracy on our supervised dataset. Metrics are abbreviated as **Align**Score, **CLIP**Score, **BLIP2**-ITM, **Pick**Score, **BVQA** and **VQAS**core. The **P** column denotes the prompt type (abbreviated as: EV = entity variation, EP = entity placement, PV = property variation). The scores are scaled such that 0 indicates random picking, -1 indicates preference for contrast pairs, and 1 indicates correct preference for matching pairs. We bold the highest score for each row if it is higher than 0.

DeepSeek and Flux generated text-image pairs to reduce potentially confounding prompting styles. For each data point, we include two random evaluation directions, where the overall distribution is 40% matching and 60% non-matching with the intention to make the model slightly more critical towards errors. For the training, we use 12k text-image pairs from this data sample. Detailed parameters can be found in Appendix E. Future work could further optimize the hyper-parameter selection, or explore training all model parameters. We evaluate CROCScore on (1) CROC$^{\text{hum}}$, (2) GenAI-Bench (Li et al., 2024), (3) Winoground (Thrush et al., 2022), and (4) the TIFA benchmark (Hu et al., 2023) with original annotations and additional annotations by Cho et al. (2024).

5 Results & Analysis

In this section, we analyze the metrics' performance on our datasets. Further, we discuss the human evaluations and the implications on the usage of auto-generated datasets to evaluate T2I metrics.

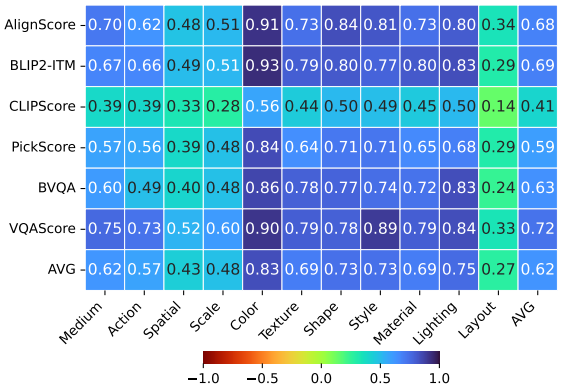

Figure 4: Scaled image-based accuracy per metric on the top-level properties of CROC$^{\text{syn}}$.

5.1 Quantitative Results

CROC$^{\text{syn}}$ Table 2 shows the scaled accuracy each metric reaches (averaged across text and image generation models). "Scaled" means that we scale the accuracy such that 0 indicates random choice, negative values indicate a preference towards contrast pairs and positive values indicate a preference towards matching pairs.[7] For example, the first table cell indicates that AlignScore reaches a scaled accuracy of 0.022 in successfully rating the pair T_O-I_O higher than T_C-I_O. Here, the **P**rompt column indicates the experiment type is *entity variation (EV)*. The metric VQAScore (with a scaled accuracy between -0.164 and 0.984) performs best in 6/12 evaluation setups. Notably, this includes 3/4 property variation directions. Second-best are the embedding-based metrics BLIP2-ITM (3/12) and AlignScore (2/12). That means, similar to the findings of Saxon et al. (2024), embedding-based metrics like AlignScore perform competitive to VQA-based metrics in some settings. Compared to the VQA-based approaches, the other types have the benefit of being more resource-efficient (see Table 7). All scaled accuracies for inverse text-based entity variation in the table are negative. We hypothesize that the chosen image-generation models were not able to sufficiently generate the entities with changed definitions. The highest accuracy is achieved for entity placement. This is reasonable, due to the clear distinction of envi-

[7] As described in §3.1, due to the random chance of the evaluation setups, text-based evaluation on CROC$^{\text{syn}}$ is scaled such that 0.83 is mapped to 0, while 0.5 is mapped to 0 for image-based evaluation.

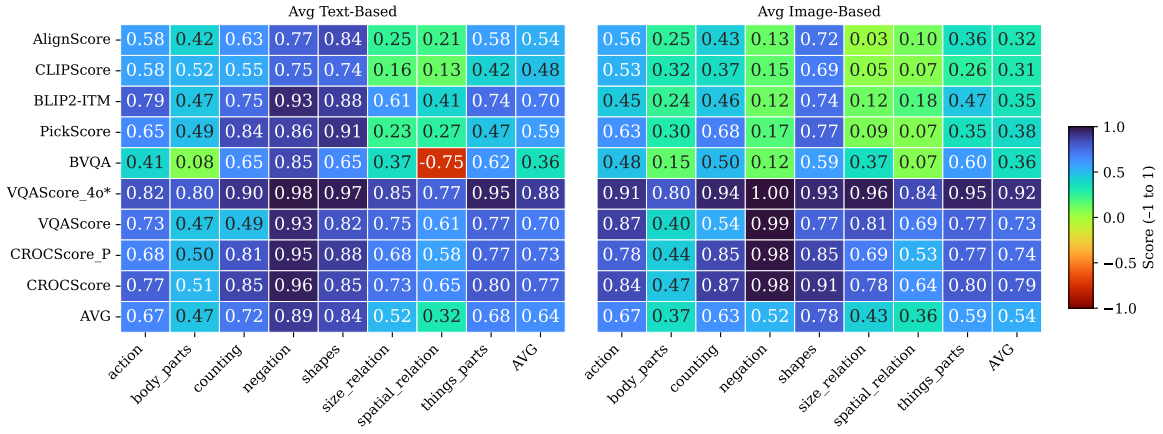

Figure 5: Scaled metric accuracy per category and evaluation direction for CROC^{hum}. For 1, a metric correctly rated all matching pairs higher than the contrast, for 0 it is random and for -1 it rated all contrast pairs higher than the matching pair. The tables show the cell-wise average of the forward and inverse evaluations. *VQAScore_4o may exhibit mild bias, because a small number of images was created interactively with GPT-4o and GPT-5. Excluding these rows only changes its average by 0.01.

ronments the entities are placed in. Notably, in text-based evaluation both entity placement and variation exhibit lower inverse than forward accuracies, whereas the relationship reverses under image-based evaluation. This directional asymmetry suggests that the metrics are more robust in judging unexpected text-image pairs when comparing between images than for texts. The Kendall (Kendall, 1945) correlation between the accuracies for the two text generation models is 0.872, i.e., the setup is very stable in that regard.

Fine-grained analysis - CROC^{syn} Figure 4 shows a heatmap of scaled accuracies for the top-level properties of our taxonomy for image-based evaluation, i.e., considering property variation. All values, besides *Layout*, range between 0.28 and 0.93. As in Table 2, VQAScore performs best. *Color* is the easiest category, with an average of 0.83, while *Layout* is the hardest, with 0.28. As this is CROC^{syn}, errors may reflect both metric failures and generation failures, although our aggregation mitigates the latter to some extent. However, we can also see that for many cases, some metrics perform better than others. In other words, these can act as an upper bound for performance (unless surpassed in human evaluation). For example, in the *Layout* category, AlignScore surpasses PickScore by 0.05 and in the *Action* category, VQAScore surpasses AlignScore by 0.11 (Figure 9 discusses the effect of prompt lengths). These comparisons can guide metric selection for specific use-cases. Future work might explore the usage for metric comparison in targeted domains, such as industry applications.

Results on CROC^{hum} Figure 3 shows an example of six failure cases identified by our setup and Figure 5 gives an overview of the scaled metric accuracy on CROC^{hum}. We have averaged the results for the forward and inverse setting, because only for *action* (where the contrast includes more unusual actions) and *negation* (where the contrast includes the negation) the inverse direction is expected to perform much differently. In this evaluation, we include our own tuned Metric CROCScore and its base metric (Phi4 with contrastive prompting), indicated as CROCScore$_P$. We do not evaluate and compare CROCScore on CROC^{syn} because it was trained on it. Additionally, we add VQAScore with GPT-4o backend, which is much more cost and time intensive than the other metrics. Overall, the most difficult categories are *spatial relations* and *body parts*. The simplest categories are *shapes* and text-based *negation*. VQAScore with GPT-4o wins against the open-source metrics by 0.11 and 0.13 points for the text-based and image-based evaluation respectively. Among the open-source metrics, CROCScore has the highest average accuracy, which is 0.04 and 0.05 points above CROCScore_P. Notably, CROCScore uses only 5.6B parameters compared to VQAScore's CLIP-Flan-T5-XXL with 11.3B parameters, i.e., it is also more resource-efficient. An interesting case is BVQA for spatial relations, which shows a neg-

ative scaled accuracy, indicating that the metric has learned to inverse spatial relations. Only the VQAScore and CROCScore metrics can adequately handle the tricky inverse image-based negation case, where the metric needs to decide that something is not in the picture (all other metrics are lower than 0.17 for image-based negation). Interestingly, AlignScore does not perform as well as on the CROC^{syn} dataset. The underlying Align model (Jia et al., 2021) was trained on noisy, contrastive data. Perhaps, this makes the model more in-domain on CROC^{syn}, while fine image details, like body parts, are less considered. Likewise, BLIP2-ITM only performs competitive on the text-based setup. For VQAScore, 35% of the setups have a value below 0.6, that is, (because of scaling) at least one out of 5 samples failed. For CLIPScore, it is 81.3% of the setups. The GPT-4o VQAScore is below 0.6 in none of the cases. Overall, we can see (uniform colors) that the weaker block of the upper five and the block of the lower four metrics (VQAScore and CROCScore) have similar failure rates. In Figure 8, we underline this finding with pairwise metric correlations. For image-based evaluation, all metrics in the upper block have averages below or equal to 0.4 scaled accuracy, while the block of VQAScore and CROCScore has at least 0.7. For text-based evaluation, this gap is much smaller: BLIP2-ITM reaches an average of 0.7. In other words, image-based evaluation is more difficult for embedding-based metrics. On the other hand, VQA-based metrics achieve about 0.03 points less for text-based evaluation. For action and shape, the embedding-based metrics even outperform the VQA-based metrics in text-based evaluation. The fine-tuning of CROCScore improves the untuned CROCScore$_p$ in 14 of 16 categories, i.e., it increases the robustness of the metric (for example, action is increased by 0.09 and counting by 0.04). Further, CROC^{syn} contains images generated by models with performance-runtime tradeoff. Future work could further explore the use of stronger T2I models to tune advanced metrics.

CROCScore on further benchmarks To verify the strong performance of CROCScore, we evaluate it on (1) GenAI-Bench (Table 3), (2) Winoground (Table 4), and (3) the TIFA benchmark (Table 5). On most of these benchmarks, CROCScore achieves higher scores than the other open-source metrics. Only for the Kendall score of the DSG TIFA annotations and for *Image* on Winoground, other metrics win. This highlights the value of CROC^{syn} for tuning new metrics.

Metric	**Kendall** τ_B (Basic/Adv/Overall)	**Pairwise Acc.** (Basic/Adv/Overall)
Align	0.072 / 0.113 / 0.131	0.481 / 0.503 / 0.514
BLIP2	0.081 / 0.485 / 0.147	0.141 / 0.522 / 0.516
VQAS	0.403 / 0.310 / 0.398	0.637 / 0.596 / 0.641
CROC_P	0.443 / 0.386 / 0.452	0.623 / 0.624 / 0.650
CROC	**0.446 / 0.401 / 0.454**	**0.649 / 0.631 / 0.660**

Table 3: Results of **Align**Score, **BLIP2**-ITM, **VQAS**core, **CROC**Score$_{plain}$ and **CROC**Score on GenAi-Bench (Li et al., 2024), where plain refers to the metric without fine-tuning. Basic, Advanced and Overall are categories in GenAi-Bench. The reported measures are Kendall correlation and Pairwise Accuracy (Deutsch et al., 2023).

Metric	Text	Image	Group
Align	0.353	0.105	0.0825
BLIP2	0.428	0.223	0.183
VQAS	0.605	**0.575**	0.458
CROC_P	0.587	0.522	0.428
CROC	**0.615**	0.558	**0.468**

Table 4: Results of **AlignScore**, **BLIP2**-ITM, **VQAScore**, **CROCScore**$_{plain}$, and **CROCScore** on Winoground (Thrush et al., 2022). Winoground also contains contrastive text-image pairs; the reported values are accuracies for distinguishing matched from mismatched pairs in both *Text* directions, both *Image* directions or all four directions (*Group*).

Comparison: CROC^{syn} vs. CROC^{hum} To evaluate how correlated CROC^{syn} is with CROC^{hum}, we first select property groups in CROC^{syn} that are similar to the properties in CROC^{hum}: (1) action, (2) spatial, (3) size, (4) shape and (5) negation. For each combination of these five groups and the 6 metrics evaluated on CROC^{syn}, we compute the average accuracy. Then, we compare these accuracies, i.e., these five rankings, with the corresponding categories in Figure 5. To do so, we compute the Kendall correlation of the flattened accuracy tables. For text-based evaluation, this yields a Kendall agreement of 0.40 ($p<0.003$) and for image-based it yields a Kendall agreement of 0.52 ($p<0.0001$). In other words, even though (1) the categories are not exactly the same, (2) the prompts of CROC^{syn} are more detailed and (3) these categories can be difficult to generate, the metric rankings per category are similar. The text-

Metric	Original (K/P)	DSG (K/P)
Align	0.299/0.408	0.276/0.393
BLIP2	0.294/0.409	0.233/0.357
VQAS	0.532/0.657	0.527/0.665
CROC_P	0.549/0.645	**0.538**/0.641
CROC	**0.550/0.680**	0.532/**0.675**

Table 5: Results of **Align**Score, **BLIP2**-ITM, **VQAS**core, **CROC**Score$_{\text{plain}}$ and **CROC**Score on TIFA (Hu et al., 2023). We show the **K**endall and **P**earson correlations. The *Original* column shows the correlation to scores in the original TIFA paper by Hu et al. (2023). The *DSG* column shows the correlation to the re-annotated data by Cho et al. (2024).

Metric	F. T.	I. T.	F. I.	I. I.	Avg
Human$_p$	**0.659**	**0.522**	0.590	0.526	**0.574**
Align	0.145	0.120	0.707	0.695	0.417
BLIP2	0.036	0.048	0.647	0.727	0.364
VQAScore	0.421	-0.039	**0.711**	**0.755**	0.462
Human$_i$	**0.832**	**0.684**	**0.768**	**0.705**	**0.738**
Human$_p$	0.600	0.411	0.495	0.495	0.500
Align	-0.027	0.053	0.663	0.684	0.334
BLIP2	-0.065	0.053	0.558	0.642	0.355
VQAScore	0.305	-0.091	0.705	0.579	0.380

Table 6: **Human**, **Align**Score, **BLIP2**-ITM and **VQAScore** scaled accuracy on CROC$^{\text{syn}}$ in the directions **F**orward **T**ext-Based, **I**nverse **T**ext-Based, **F**orward **I**mage-Based and **I**nverse **I**mage-Based. The upper part shows scores on 2000 text-image pairs annotated by Prolific annotators (Human$_p$). The lower part shows scores on 380 text-image pairs that were additionally annotated by in-house annotators (Human$_i$).

based agreement might be lower because of this different complexity in dataset texts.

5.2 Human evaluation

Table 6 shows the results of our human evaluation on CROC$^{\text{syn}}$. On a set of 2,000 text-image pairs, Prolific annotators achieve higher scaled accuracy than VQAScore on text-based items but lower scaled accuracy on image-based items. Krippendorff's alpha per batch ranges from -0.05 to 0.656 (median 0.404) across 26 batches, indicating that our attention checks only partially mitigated variability. To obtain an additional higher-quality reference, we label a subset of 380 image pairs with three in-house annotators. They outperform all automatic metrics in every category. Their agreement is $\alpha = 0.604$. TIFA (0.67) and GenAI-Bench (0.72) report comparable agreements as substantial. The human in-house performance indicates that CROC$^{\text{syn}}$ can successfully reveal metric shortcomings, because metrics have not reached the upper boundary.

In addition, we measure the disagreement for each property on the in-house sample as the *mean absolute difference* between per-sample annotator scores. The highest disagreement occurs for the properties *right-of* and *close proximity*. One sample, e.g., swaps *the position of musician in a concert*. We verify that images are generated as intended. However, the prompt features 93 words, so the disagreement might be caused by the contrast either being overlooked or being rated differently in relation to the other, correctly generated, parts of the prompt.

We also perform a human evaluation on **CROC$^{\text{hum}}$**. Here, the three annotators achieve a Krippendorff's Alpha of 0.957 and an average scaled accuracy (from -1 to 1) of 0.949, highlighting the quality of the dataset. Notably, this human accuracy is largely better than the metric performance in Figure 5, i.e., about 0.1 points better than VQAScore with GPT-4o.

6 Conclusion

We introduce CROC, a meta-evaluation framework for T2I metrics that enables fine-grained robustness tests based on contrastive text-image pairs. We use this framework to generate CROC$^{\text{syn}}$, a novel large-scale dataset that we use to evaluate existing metrics and to train our new metric CROCScore. The evaluation shows that our approach can be used to successfully compare metric performance. Further, we introduce CROC$^{\text{hum}}$, a contrastive human-supervised dataset of challenge categories that allows a fine-grained analysis of metric failure cases. Both datasets show that all tested metrics still exhibit blindspots and do not reach human performance. This is especially true for body parts and spatial relations, although VQAScore with a GPT-4o backend is already much more robust. Our metric CROCScore achieves state-of-the-art results among the tested open-source metrics on CROC$^{\text{hum}}$ dataset and GenAI-bench (Li et al., 2024). Further, it provides a cost advantage over GPT-4o. This opens an interesting future research path of improving VQA-based T2I metrics by tuning on fine-grained generated contrastive datasets.

Acknowledgements

The NLLG group gratefully acknowledges support from the Federal Ministry of Education and Research (BMBF) via the research grant "Metrics4NLG" and the German Research Foundation (DFG) via the Heisenberg Grant EG 375/5-1. Further, we thank the annotators of our human evaluations. Also, we thank the action editor and the reviewers for their constructive feedback that has helped us to improve our work. The authors also acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1597-1 FUGG.

References

Black Forest Labs. 2024. Flux. https://github.com/black-forest-labs/flux.

Yanran Chen and Steffen Eger. 2023. MENLI: Robust evaluation metrics from natural language inference. *Transactions of the Association for Computational Linguistics*, 11:804–825.

Jaemin Cho, Yushi Hu, Jason Michael Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. 2024. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *The Twelfth International Conference on Learning Representations*.

Jaemin Cho, Abhay Zala, and Mohit Bansal. 2023. DALL-Eval: Probing the reasoning skills and social biases of text-to-image generation models. In *ICCV*.

DeepSeek-AI. 2025. Deepseek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *CoRR*, abs/2501.12948v1.

Daniel Deutsch, George Foster, and Markus Freitag. 2023. Ties matter: Meta-evaluating modern metrics with pairwise accuracy and tie calibration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12914–12929, Singapore. Association for Computational Linguistics.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Melissa Cheyenne Foote. 2018. Design guides. https://guides.lib.berkeley.edu/design. Accessed: 2025-03-21, University of California, Berkeley Library.

Sebastian Hartwig, Dominik Engel, Leon Sick, Hannah Kniesel, Tristan Payer, Poonam Poonam, Michael Glöckler, Alex Bäuerle, and Timo Ropinski. 2025. A survey on quality metrics for text-to-image generation. *IEEE Transactions on Visualization and Computer Graphics*, 31(10):9464–9483.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. 2023. TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20406–20417.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. 2023. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Pro-*

ceedings of Machine Learning Research, pages 4904–4916. PMLR.

Marzena Karpinska, Nishant Raj, Katherine Thai, Yixiao Song, Ankita Gupta, and Mohit Iyyer. 2022. DEMETR: Diagnosing evaluation metrics for translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9540–9561, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Maurice G. Kendall. 1945. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251.

Kyungtae Kim. 2024. Flux.1 prompt guide. https://www.giz.ai/flux-1-prompt-guide/. Accessed: 2025-03-21.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. 2023. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 36652–36663. Curran Associates, Inc.

Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Teufel, Marco Bellagente, Minguk Kang, Taesung Park, Jure Leskovec, Jun-Yan Zhu, Fei-Fei Li, Jiajun Wu, Stefano Ermon, and Percy S Liang. 2023. Holistic evaluation of text-to-image models. In *Advances in Neural Information Processing Systems*, volume 36, pages 69981–70011. Curran Associates, Inc.

Christoph Leiter, Piyawat Lertvittayakumjorn, Marina Fomicheva, Wei Zhao, Yang Gao, and Steffen Eger. 2024. Towards explainable evaluation metrics for machine translation. *Journal of Machine Learning Research*, 25(75):1–49.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Emily Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. 2024. GenAI-bench: A holistic benchmark for compositional text-to-visual generation. In *Synthetic Data for Computer Vision Workshop @ CVPR 2024*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. 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*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. 2024. Evaluating text-to-visual generation with image-to-text generation. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part IX*, page 366–384, Berlin, Heidelberg. Springer-Verlag.

Microsoft. 2025. microsoft/Phi-4-multimodal-instruct.

Supreeth Narasimhaswamy, Uttaran Bhattacharya, Xiang Chen, Ishita Dasgupta, Saayan Mitra, and Minh Hoai. 2024. HanDiffuser: Text-to-image generation with realistic hand appearances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2468–2479.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad

Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson,

Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. GPT-4o system card. *CoRR*, abs/2410.21276v1.

OpenAI. 2025. Gpt5. https://openai.com/gpt-5/.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems*, volume 37, pages 68772–68802. Curran Associates, Inc.

Qwen Team. 2025. QwQ-32B: Embracing the power of reinforcement learning.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1060–1069, New York, New York, USA. PMLR.

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michael Saxon, Fatima Jahara, Mahsa Khoshnoodi, Yujie Lu, Aditya Sharma, and William Yang Wang. 2024. Who evaluates the evaluations? objectively scoring text-to-image prompt coherence metrics with t2IScorescore (TS2). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

StabilityAi. 2024a. Stable diffusion 3.5. https://stability.ai/news/introducing-stable-diffusion-3-5. Accessed: 2025-03-21.

StabilityAi. 2024b. Stable diffusion 3.5 prompting guide. https://stability.ai/learning-hub/stable-diffusion-3-5-prompt-guide. Accessed: 2025-03-21.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. 2022. Winoground: Probing vision and language models for visio-linguistic compositionality. In *CVPR*.

Xiao Wang, Daniil Larionov, Siwei Wu, Yiqi Liu, Steffen Eger, Nafise Sadat Moosavi, and Chenghua Lin. 2025. ContrastScore: Towards higher quality, less biased, more efficient evaluation metrics with contrastive evaluation. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 3045–3060, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Olivia Wiles, Chuhan Zhang, Isabela Albuquerque, Ivana Kajic, Su Wang, Emanuele Bugliarello, Yasumasa Onoe, Pinelopi Papalampidi, Ira Ktena, Christopher Knutsen, Cyrus Rashtchian, Anant Nawalgaria, Jordi Pont-Tuset, and Aida Nematzadeh. 2025. Revisiting text-to-image evaluation with Gecko: on metrics, prompts, and human rating. In *The Thirteenth International Conference on Learning Representations*.

Xiangru Zhu, Penglei Sun, Chengyu Wang, Jingping Liu, Zhixu Li, Yanghua Xiao, and Jun Huang. 2023. A contrastive compositional benchmark for text-to-image synthesis: A study with unified text-to-image fidelity metrics. *CoRR*, abs/2312.02338.

A Taxonomy Properties

Here, we give an overview of all taxonomy properties and example topics (with 2 example entities each):

Example Topics: *Nature:* Landscapes [Mountain, River], Flora [Tree, Flower], Fauna [Deer, Eagle], Weather Phenomena [Lightning Bolt, Snowflake], Underwater [Sea Turtle, Coral]; *People:* Portraits [Adult Human, Child], Groups [Friends, Crowd], ...; *Animals:* Wild Animals [Lion, Elephant], Domestic Animals [Dog, Cat], ...; *Architecture:* Residential Buildings [House, Cottage], Commercial Buildings [Skyscraper, Shopping Mall], ...

Property - *Medium*: *Photography*, *Illustration*, *3D Rendering*, *Anime*, *Mixed Media*, *Painting*

Property - *Relation*

Action: *Gesture:* Pointing, Waving, Facial Expression, Nodding; *Full-Body Movement:* Running, Dancing, Jumping, Swimming

Spatial: *Foreground/Background:* Foreground Emphasis, Midground Placement, Background Silhouette; *Proximity/Overlap:* Close Proximity, Overlapping Forms, Left-of, Right-of, Above, Below, Inside

Scale: *Exaggerated:* Giant Figures, Miniature Objects, Distorted Perspective; *Realistic Scale:* Life-Size Representation, Proportional Figures, Consistent Depth

Property - *Attribute*

Color: *Monochrome*, *Vibrant*, *Red*, *Blue*, *Green*, *Yellow*, *Purple*, *Orange*, *Pink*, *Brown*, *Black*, *White*

Texture: *Smooth*, *Rough*, *Reflective*

Shape: *Geometric*, *Organic*

Style: *Realistic*, *Impressionistic*, *Minimalist*

Material: *Metallic*, *Wooden*, *Fabric*, *Plastic*, *Glass*, *Stone*, *Paper*

Lighting: *Natural Light*, *Artificial Light*, *High Contrast*

Layout: *Centered*, *Rule of Thirds*, *Asymmetrical*

B Inverse Equations

Here we show the equation that we apply for the evaluation of inverse text-base samples. M is a metric, T_O and I_O are the original text and image, T_C and I_C are the contrast text and image. i and j are indices for one of multiple images:

$$j^* = \underset{i=1,\dots,n}{\operatorname{argmax}} M(T_C, I_C^i),$$
$$M(T_C, I^{j^*}) > M(T_O, I^{j^*}) \quad (4)$$

This is the respective equation for inverse image-based evaluation:

$$\max_{i=1,\dots,n} M(T_C, I_C^i) > \max_{i=1,\dots,n} M(T_C, I_O^i) \quad (5)$$

C Detailed examples for generation and evaluation

In the following, we present one example of data construction and evaluation with our pseudo-labeled generation process (for CROC$^{\text{syn}}$) and one example with our human-supervised generation process (for CROC$^{\text{hum}}$).

Pseudo-Labeled Generation - Property Variation

1. **Property and subject selection** This is an example for *property variation, where we* select the property "red" and the subject "Transportation".

2. **Prompt generation** In this example, we generate an image with stable diffusion. Hence, we load the stable diffusion guide. Then we fill the *property variation* prompt template from Appendix F with our data and pass it to an LLM, here the Deepseek model, to generate 5 outputs. One valid output is the following JSON:
{ "prompt (T_O)": "A majestic red steam locomotive chugging through a mountain valley[...]",
"contrast_prompt (T_C)": "A majestic blue steam locomotive chugging through a mountain valley[...]" }

3. **Image generation** Then, we generate 5 images from the extracted prompt and contrast prompt each (see Figure 6).

4. **Metric computation** Next, we compute the metric scores for all Text-Image combinations.

5. **Metric evaluation** Finally, we evaluate the metric(s) based on the score. For example, for **forward text-based** evaluation we first select the highest $M(T_O, I_O^i)$ (green box in Figure 6 and then compare it to the respective $M(T_O, I_C^i)$ score (red box). In the example $M(T_O, I_O^2) = 14.4$ is smaller than $M(T_C, I_O^2) = 14.7$, therefore the metric did not pass the test case.[8]

Human-Supervised Generation Example

1. **Supervised prompt construction** In this example, we choose a prompt and contrast prompt of the category *body parts* that was created through interactive querying of GPT-4o. **Prompt (T_O):** "A hand with only its index finger colored red." **Contrast** (T_C): "A hand with only its ring finger colored red."

[8] Metric scores are not always scaled between 0 and 1.

Metric	Type	Version and Model(s)	Runtime
CLIPScore	Embed	torchmetrics_1.6.2; `clip-vit-large-patch14`	ca. 275 Sec.
BLIP2-ITM	Embed	VQAScore (Implementation) from 3.25; `blip2-itm-vit-g`	ca. 112 Sec.
ALIGNScore	Embed	T2IScoreScore 1.25;`align-base`	ca. 70 Sec.
PickScore	Tuned	`PickScore_v1`	ca. 61 Sec.
VQAScore	VQA	VQAScore from 03.25;`clip-flant5-xxl`	ca. 40 Min. (no batching)
VQAScore_4o	VQA	VQAScore from 03.25;`GPT-4o (04.25)`	ca. 55 Min.
BVQA	VQA	T2I-CompBench from 03.25	ca. 22.7 Min.
CROCScore	Tuned VQA	microsoft/Phi-4-multimodal-instruct, batch size=4	ca. 11 Min.

Table 7: Overview of the metrics we evaluate, with a brief description, key configuration details, and their runtime on 1 000 "body parts" samples from CROC^{hum}. Metrics that directly return the quality score are faster than VQA based metrics, because they do not require auto-regressive generation.

2. **Image generation** Next, we generate 100 images for each prompt. Figure 7 shows examplary generations.

3. **Supervised image filtering** Then, we manually remove all images that are not matching the prompts. In Figure 7, these are titled "Invalid image".

4. **Metric computation** Next, we compute the metric scores.

5. **Metric evaluation** Here, we demonstrate image-based evaluation. To calculate the accuracy, we first compare all $M(T_O, I_O^i)$ with all $M(T_O, I_C^i)$. That means, for the four valid images in Figure 7 we compare each score of the original outputs (left) with each score of the contrast outputs (right): (1) $18.3 > 17.2$, (2) $18.3 > 18.2$, (3) $16.8 > 17.2$ and (4) $16.8 > 18.2$. Because two of these four conditions are true, the final score is $\frac{2}{4}$.

D Categories of CROC^{hum}

Table 9 gives an overview of the 8 selected categories for CROC^{hum}. Notably, this list is not exhaustive and other image generation failure cases exist, but are not covered in this work.

E Training Parameters for CROCScore

We used the following training parameters: optim=adamw, adam beta1=0.9, adam beta2=0.95, adam epsilon=1e-7, max grad norm=1.0, lr scheduler type='linear', warmup steps=100, logging steps=10, lr=$5.0e^{-}6$, weight decay = 0.01. Additionally, we have saved the evaluation directions that were used in our dataset.

	AlignScore	CLIPScore	PickScore
F. T.	0.648	0.021	0.431
I. T.	-0.277	-0.048	-0.308

Table 8: **Initial Experiment:** Scaled accuracy on contrastive text-image pairs generated from T2ICompBench (Huang et al., 2023). We generate paraphrases and contrast prompts for each original prompt with GPT4 and generate images with Flux. Therefore, contrast prompts are more unexpected. On a small human-annotated subset, humans perform above random (0.17) on the inverse set, while the tested metrics do not (e.g., AlignScore with -0.26). This created our hypothesis that metrics might have a bias to rate unexpected matching pairs higher than contrasting pairs with a natural prompt. We test this with the categories *entity placement* and *entity variation* in our unsupervised dataset.

F Templates for prompt generation

Table 10 shows the templates we use for prompt generation. "" is copied from the first prompt. The prompting guides that we adapted are written by Kim (2024) for FLUX and by StabilityAi (2024b) for Stable Diffusion. We chose them because of their structured breakdown and qualitative example pictures. In a second step, we further streamlined them for prompt usage in an interactive conversation with GPT-4o.

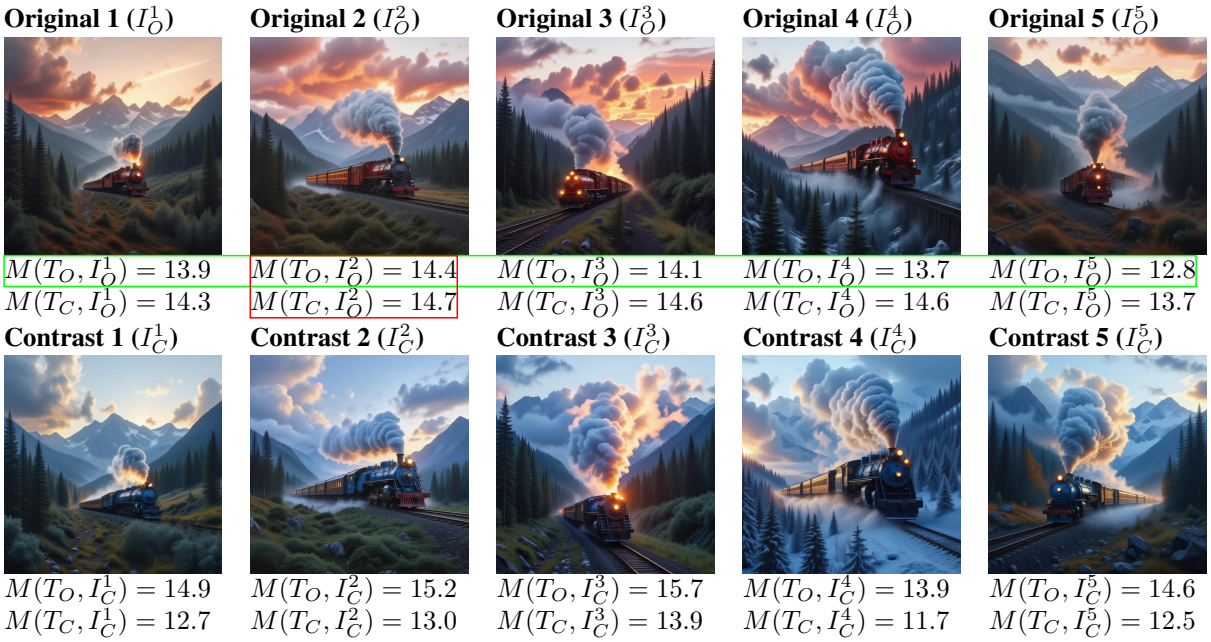

Figure 6: Generated original and contrast images for property variation with subject *Transportation* and property *Red*. T_O: *A majestic red steam locomotive chugging through a mountain valley[...]*. T_C: *A majestic blue steam locomotive chugging through a mountain valley[...]*. Further, we display the metric scores for AlignScore for all combinations. In **text-based forward** evaluation, we first find the highest value for T_O-I_O^i pairs (green box) and then compare it to the respective T_C-I_C^i pair (red box).

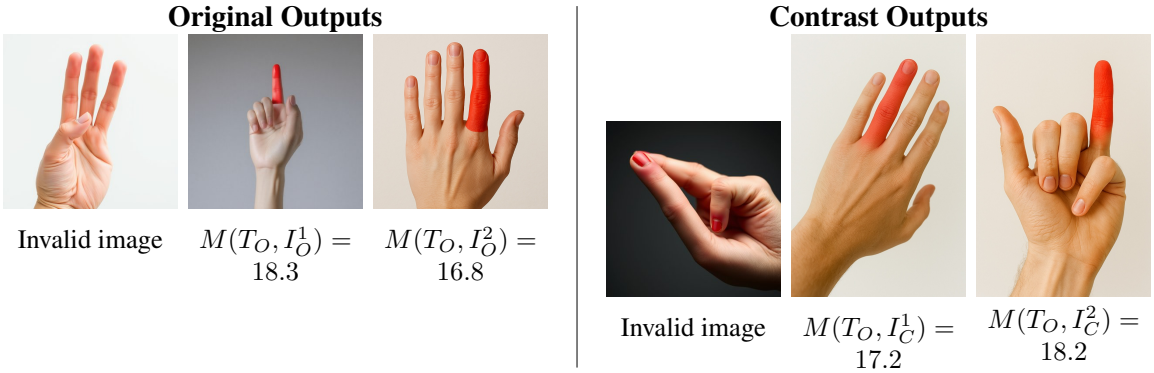

Figure 7: Example images of the *body parts* test case (per-default GPT-4o sometimes generates non-square image dimensions). T_O: *A hand with only its **index finger** colored red*. T_C: *A hand with only its **ring finger** colored red*. Further, we display the AlignScore scores necessary for forward image-based evaluation where we compare each matching score of the original images (left side) with each contrast score of the contrast outputs (right side). That means, 18.3 is (1) higher than 17.2 and (2) higher than 18.2, but 16.8 is (3) lower than 17.2 and (4) lower than 18.2. Therefore the overall accuracy is $\frac{2}{4}$.

Property	Description
Action	The action of an entity is changed. For example, “A ball bounces” vs. “A ball sits”.
Body Parts	The highlighted small body part is changed. For example, “A hand with only its ring finger colored red” vs. “A hand with only its index finger colored red”.
Counting	The count of an entity is changed. For example, “Two apples” vs. “Four apples”.
Negation	One entities is negated. For example, “A phoenix and a flag” vs. “A phoenix and no flag”
Shapes	The shape of an entity is changed. For example, “An apple in the shape of a cube” vs. “An apple in the shape of a torus”.
Size Relation	The size relation between two objects is changed. For example “A bigger giraffe and a smaller child” vs. “A smaller giraffe and a bigger child”.
Spatial Relation	The spatial relation between two entities is changed. For example “A fish left of a car” vs. “A fish right of a car”
Parts of things	The highlighted part of a thing is changed. For example, “A bike with a blue saddle” vs. “A bike with a blue handlebar”

Table 9: Properties of CROC^{hum}

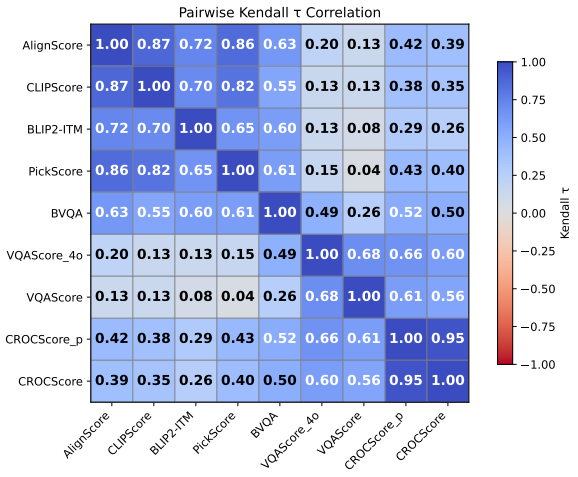

Figure 8: Kendall correlation between row-wise metric accuracies on CROC^{hum} (Figure 5).

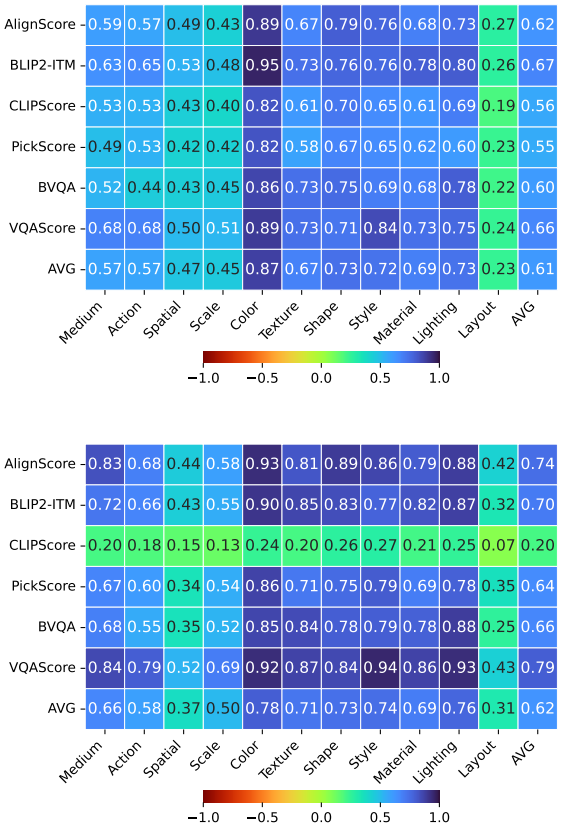

Figure 9: Scaled image-based accuracy per metric on the top-level properties of CROC^{syn}. *Left*, samples are filtered to only include prompts smaller than or equal to 280 characters (heuristic for 77 tokens). *Right*, only prompts larger than 280 characters are included. The performance of CLIPScore is degraded for long prompts (due to its token limit) and the performance of all other metrics is increased). Notably, with AlignScore outperforming BLIP2-ITM.

Property Variation: Consider the following guide on writing a good prompt with {model_name}:
{guide}
Write a prompt for {model_name} that describes a specific scene about "{subject_name}" that involves the concept "{property_name}" ({property_description}).
Additionally, write a contrast prompt that strongly contrasts the original prompt in terms of the concept "{property_name}" ({property_description}), but keeps the wording and content of the prompt the same as far as possible.

For example, if the concept is a color the contrast prompt may use a different color.
Pay attention not to use unusual words and make sure that the contents can be displayed as images. Use simple and understandable language. Do not use phrases like "the same" in the contrast prompt.
Write the prompts very short, concise and clear. Do not write more than a single line. Do not write more than 30 words. Think step by step, then return your output in the following format: {{ "prompt": "Your prompt here",
"contrast_prompt": "Your contrast prompt here" }}

Entity Variation: " "
Write a prompt for {model_name} that describes a specific scene about "{subject_name}" involving the entity {entity_name} (Definition: {entity_description}).
Additionally, write a contrast prompt that strongly changes parts of the entity definition {entity_name} (Definition: {entity_description}), but keeps the wording and content of the prompt the same as far as possible.
For example, if the entity is a human that has two arms, the contrast prompt may change the number of arms to three.

" " {{ "prompt": "Your prompt here",
"varied_definition": "Strongly changed definition of {entity_name} (Definition: {entity_description}. The definition needs to be displayable as an image and it should change the visual appearance of the entity in an unexpected way, ideally not by adding external elements, for example by changing the shape, color or changing numbers.)"
"contrast_prompt": "Your contrast prompt here" }}

Entity Placement: " "
Write a prompt for {model_name} that describes a specific scene about "{subject_name}" with the entity {entity_name} (Definition: {entity_description}).
Additionally, write a contrast prompt that places the entity {entity_name} in a picture about "{alt_subject_name}", but keeps the wording and content of the prompt the same as far as possible.
" "

Table 10: Templates for Prompt Generation