Appendix

A. Cycle Consistency and Point-wise Mutual Information

Let X and Y be random variables that take on realizations x and y, respectively. In Section 3 X and Y represent images and texts, but note how our cycle consistency score (Equation 3) and preference creation (Equations 4) are general to any X and Y. We now focus on the general case.

In Equation 3, we define $s(x \to y)$ and $s(y \to x)$ with respect to fixed backward mappings $G : Y \to X$ and $F : X \to Y$ respectively. If F, G are stochastic mappings, then we can view G as sampling some $x' = G(y)$ from the distribution $p_G(X|Y = y)$ - a distribution which is determined by G. Symmetrically, we can view F as sampling $y' = F(x)$ from the distribution $p_F(Y|X = x)$ determined by F. We then argue that distributionally,

$$\begin{aligned} s(x \to y)_d &:= \log p_G(x|y) \\ s(y \to x)_d &:= \log p_F(y|x) \end{aligned} \tag{8}$$

If the two distributions p_F and p_G sample from the same underlying distribution p, we can define joint distributional cycle consistency score. This may be the case if F and G are trained on the same dataset or with sufficient examples to model the same distributions.

$$\begin{aligned} s(x, y)_d &:= s(x \to y)_d + s(y \to x)_d \\ &= \log p(x|y) + \log p(y|x) \quad x, y \sim p(X, Y) \end{aligned} \tag{9}$$

Mutual Information Following the connection that previous work [45] has made between cycle consistency and mutual information, we rewrite the joint reward as follows:

$$\begin{aligned} s(x, y)_d &= \log p(x|y) + \log p(y|x) \\ &= \log \frac{p(x, y)}{p(y)} + \log \frac{p(x, y)}{p(x)} \\ &= \log \frac{p(x, y)^2}{p(x)p(y)} \\ &= \log p(x, y) + \text{PMI}(x, y) \end{aligned} \tag{10}$$

Therefore, we can view the joint cycle consistency score as measuring both the likelihood of the pairing $p(x, y)$ and the pointwise mutual information. In turn, CycleReward prefers x, y pairings which are both high probability and informative of each other.

B. Benefits from Reward Modeling

Because our reward model is trained with preferences from cycle consistency, it is natural to assume that the performance of raw cycle consistency scores $s(x \to y)$ and $s(y \to x)$ would be an upper bound for our reward model. In contrast, our trained reward models outperform raw cycle consistency on all benchmarks reported in Section 5 in both mapping directions.

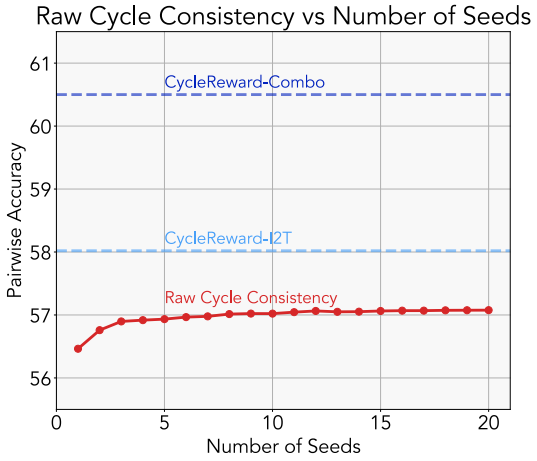

Figure 5. **Raw cycle consistency performance with increasing number of samples.** We plot DetailCaps-4870 benchmark performance (Pairwise Accuracy) for raw cycle consistency calculated over multiple samples (random seed sampling). Despite the increasing number of seeds, raw cycle consistency performance does not come close to reward model performance.

Albeit computationally slow, averaging raw cycle consistency scores over multiple reconstructions as in Equation 11 could provide more accurate alignment measurements than just a single forward pass. We define the mean image-to-text cycle consistency as follows:

$$s^*(x \to y) = \frac{1}{N} \sum_{n=1}^{N} ||x - g(y, z_n)|| \qquad z_n \sim \mathcal{N}(0, I) \tag{11}$$

This measurement averages $s(x \to y)$ scores over N decoder reconstructions. In practice, we sample reconstructions by using different random seeds for the SD3 decoder. Note we can define a symmetric mean cycle consistency score for $s(y \to x)$, but focus on the image-to-text direction in this section.

Figure 5 plots DetailCaps-4870 benchmark performance against the number of samples N used for mean cycle consistency score. Although using more seeds generally benefits raw cycle consistency, improvement tapers off around $N = 5$ and never reaches the performance of CycleReward.

Figure 7 qualitatively compares alignment computed by raw cycle consistency against our reward model. From the rich visual descriptions in our dataset, the reward model has learned that the image of the red bird corresponds best with the text description. In contrast, raw cycle consistency attempts to reconstruct the original input from the input prompt. Due to the lack of fine-grained visual information in the text, the reconstruction is more of a typical, object-

Figure 6. **Examples of CyclePrefDB**. Preferred samples are in blue and rejected samples are in red. (Left) We show input images, generated captions, and image reconstructions for image-to-text comparison pairs. (Right) shows input prompts, generated images, and text reconstructions for text-to-image comparison pairs. Generally, more accurate, descriptive captions and images that faithfully capture the prompt yield better reconstructions. However, exceptions exist such as the neon sign example (top left).

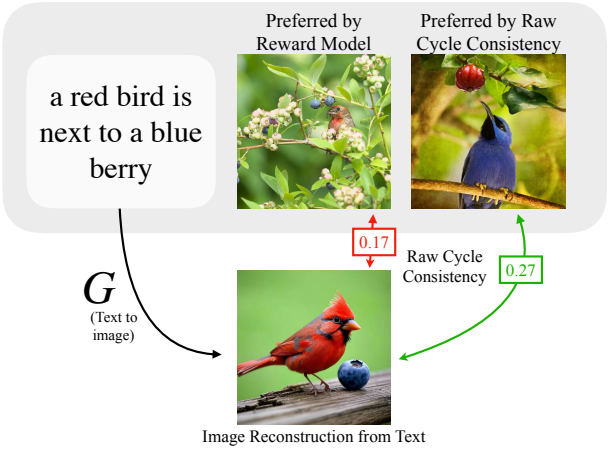

Figure 7. Raw cycle consistency $s(x \to y)$ is computed by comparing the original image (top) with its reconstruction (bottom), with similarity values shown in each box. In this example, although the reconstructed image accurately reflects the prompt, it is visually more similar to the image of the blue bird, leading to an incorrect alignment judgment based on raw cycle consistency. In contrast, our learned reward model, CycleReward, correctly identifies the true alignment.

centered bird image that happens to be structurally similar to the image of the blue bird over the red bird. This finding highlights additional benefits of distilling cycle consistency to a reward model – beyond speed and differentiability.

C. CyclePrefDB Dataset Details

Image and Text Reconstructions We provide examples of reconstructed images and texts used to create comparison pairs in our dataset in Figure 6. Generally, we find that better, more descriptive image captions lead to image reconstructions that are more similar to the input image. Symmetrically, generated images that are faithful to the prompt have text reconstructions that reflect this. However, there exists failure cases often due to poor reconstructions as in Figure 15.

Dataset Filtering Common strategies for filtering human preferences include: (1) removing duplicate entries, (2) filtering out cases where both responses are harmful or irrelevant [94], and (3) excluding low-margin examples where one response is only marginally better than the other [12]. Following these principles, we adopt a similar filtering strat-

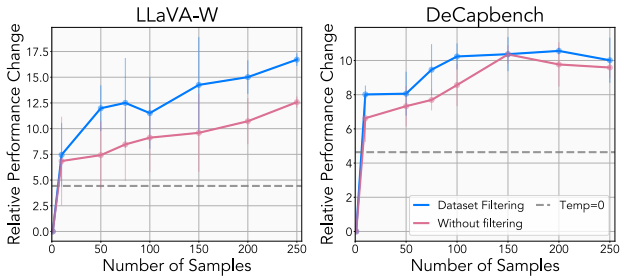

Figure 8. **Best-of-N results with and without dataset filtering**. Filtering the dataset improves inference-time optimization by enabling better candidate selection during best-of-N sampling.

egy by removing duplicate captions, excluding examples where the reward difference is within a certain threshold, i.e., $|r_i - r_j| < \tau_{\text{sim}}$, and discarding comparison pairs where the preferred reward is below a threshold, i.e., $r_i < \tau_{\text{neg}}$. In practice, we use $\tau_{\text{sim}} = 0.005$, $\tau_{\text{neg}} = 0.7$ for DreamSim, and $\tau_{\text{neg}} = 0.4$ for SBERT. In practice, training with this dataset filtering leads to a small performance gain on alignment benchmarks and a bigger performance gap in best-of-N experiments seen in Figure 8.

Prompt Choice To ensure that all image-to-text models can produce image descriptions to the best of their ability, we use the prompt recommended by the model distributor, as shown in Table 8.

Model	Prompt
BLIP2	"this is a picture of"
LLaVA1.5	"Write a detailed description of the given image."
LLaVA1.6	"Write a detailed description of the given image."
LLaVA-OV	"Write a detailed description of the given image."
InternVL2	"Please describe the image in detail."

Table 8. Prompts used for image-to-text models.

D. Model training details

D.1. Reward Modeling

We use the AdamW optimizer [57] with a batch size of 2048 for 2 epochs. The learning rate is set to 3e-5 with a weight decay of 1e-4 for optimizing $\mathcal{L}_{\text{text}}$, while $\mathcal{L}_{\text{img}}$ and joint training use a learning rate of 2e-5 with no weight decay. We set $\lambda = 1$ for joint training. Following the setup in [92], we fix 70% of the transformer layers during training, which we found to outperform full fine-tuning. All models are trained using 8 H100 GPUs.

D.2. DPO

We perform DPO to align Qwen-VL-Chat using our dataset CyclePrefDB-I2T. The model is trained for 5 epochs with the AdamW optimizer [56] and a weight decay of 0.05. We

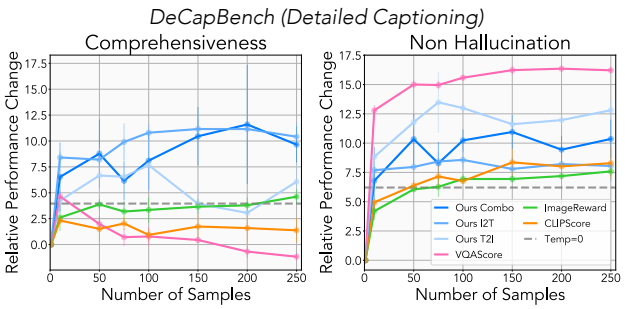

Figure 9. **DeCapBench Best-of-N DCScore breakdown**. DeCapBench evaluation is performed with DCScore which combines scores for Comprehensiveness and Non-Hallucination in the left and right plots respectively.

apply a cosine learning rate schedule with a warmup ratio of 0.1 and a peak learning rate of 1×10^{-5}. Training is performed with a global batch size of 256. To enable more efficient training, we adopt LoRA tuning. The model is trained using 4 H100 GPUs.

D.3. Diffusion-DPO

We use the Diffusion-DPO objective to align Stable Diffusion 1.5 [73] with preferences in our CyclePrefDB-T2I dataset. We use the AdamW optimizer [57] and train with an effective batch size of 512 (batch size 1 with 128 gradient accumulation steps on 4 H100 GPUs). We use learning rate 5×10^{-8} and set $\beta = 1000$ and train for 1500 steps. Similarly to the Diffusion-DPO Pick-A-Pic model, we validate checkpoints with 380 prompts from CyclePrefDB-T2I validation set and select the best checkpoint according to the mean alignment using the CycleReward-T2I reward model.

E. Additional Results

E.1. Alignment Metrics

Figure 10 shows qualitative examples of CycleReward versus other alignment metrics with ground truth preferences in purple. Overall, our CycleReward (CR) models are more successful at assessing detailed captions while performing competitively on evaluating text-to-image generation.

E.2. Best-of-N

Figures 11 and 12 show qualitative examples of how different metrics affect Best-of-N selection for detailed captioning and text-to-image generation, respectively. We show the initial (Best-of-1) output and compare it to the final output selected from the full candidate pool.

Figure 9 shows DeCapBench Best-of-N results separated into the Non-Hallucination and Comprehensiveness categories used by DCScore [93] during evaluation. All CycleReward models lead to improvement in both categories, but CycleReward-Combo and CycleReward-I2T select the best comprehensive captions, while VQAScore and

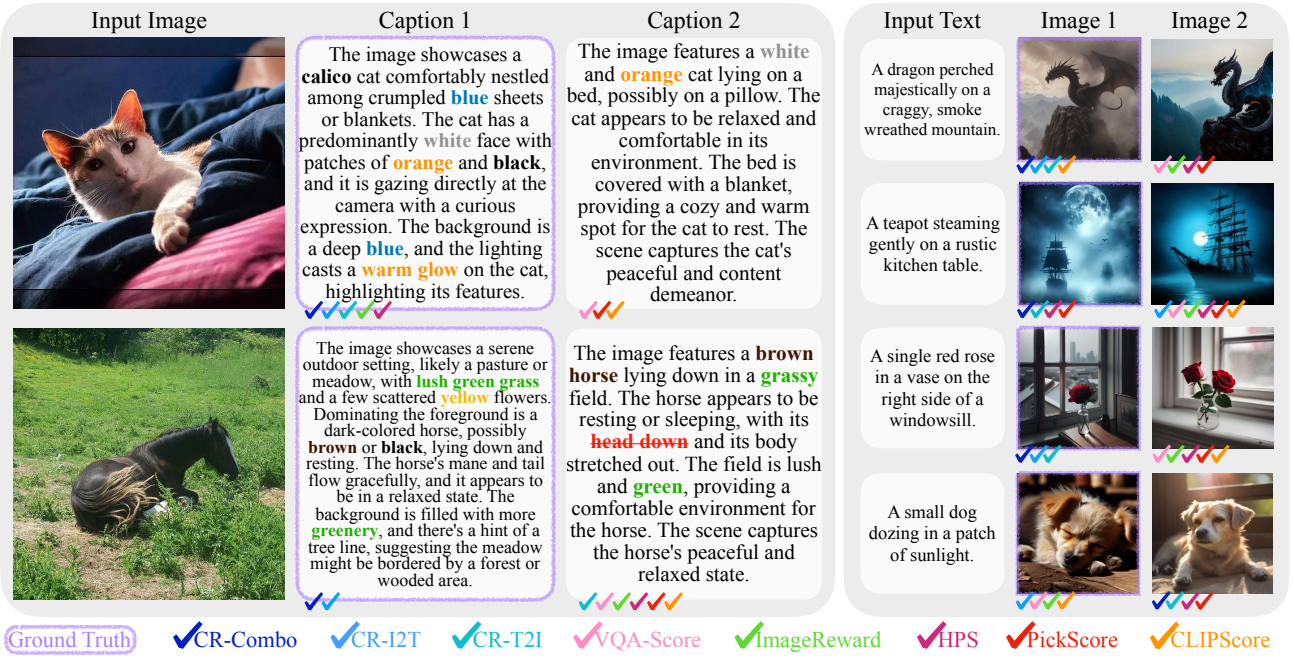

Figure 10. **Alignment metrics on DetailCaps-4870 and GenAI-Bench.** Our reward model excels at identifying detailed captions while performing competitively on GenAI-Bench. We also provide the ground truth label in purple.

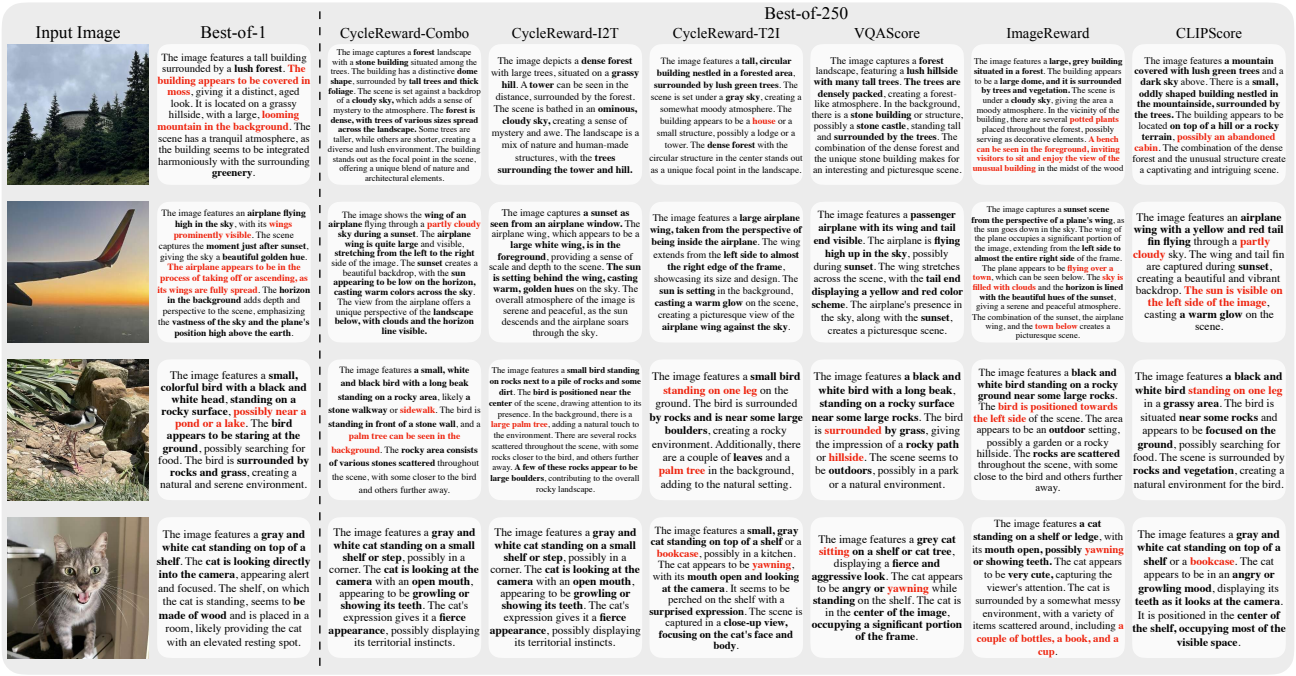

Figure 11. **Best-of-N results on DeCapBench for different metrics.** Overall, our model increases the level of detail in captions while avoiding severe hallucinations.

CycleReward-T2I yield the best non-hallucination scores. Note other metrics such as VQAScore and CLIP have trade-offs which sacrifice description for accuracy.

Sampling Settings To obtain candidate captions for Best-of-N sampling, we used a combination of temperature, nucleus, and prompt sampling with model LLaVA1.5-13B [44, 53]. We set temperature to 1.0, top p to 0.7 re-

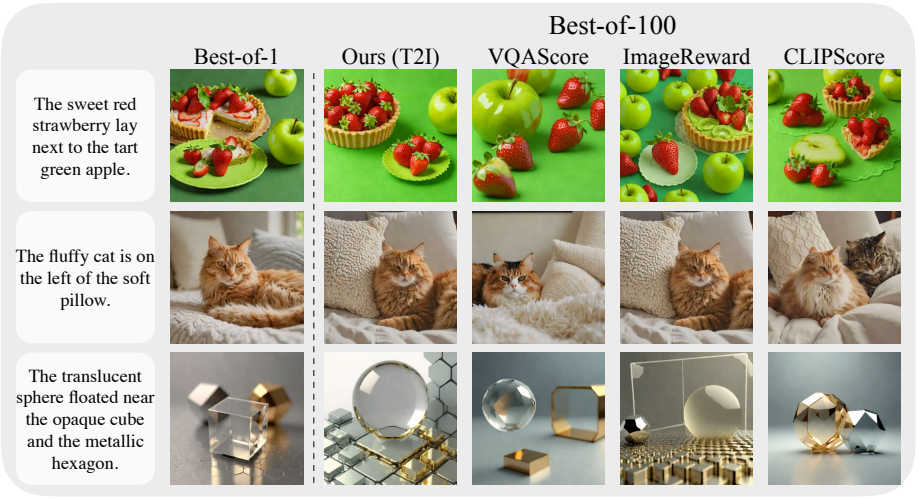

Figure 12. **Best-of-N results on T2I-CompBench for different metrics.** Optimizing with our reward model generally improves results, while VQAScore excels at following positional relationships.

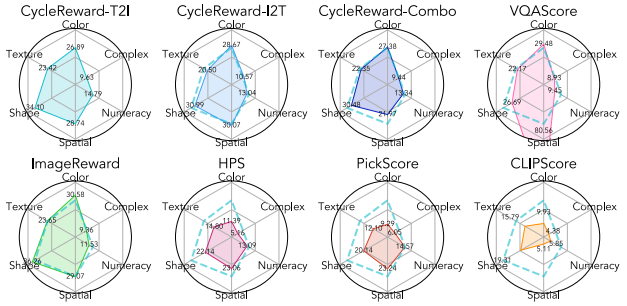

Figure 13. **Relative performance gain on T2I-CompBench from Best-of-1 to Best-of-100 across 6 categories**. We mark CycleReward-T2I's performance with a dashed line in all charts for comparison. While each metric has category-specific strengths, human-supervised ImageReward achieves the most balanced overall performance, followed closely by CycleReward-T2I.

Figure 14. **Generated images from Diffusion DPO training.** We compare images generated by the base Stable Diffusion 1.5 model, a model trained on Pick-A-Pic v2, and a model trained on CyclePrefDB-T2I (ours). Our model captures complex visual details and often outperforms the Pick-A-Pic v2 model trained with human preferences.

spectively, and choose prompts randomly from the original LLaVA dataset prompts [53]. Image candidates are generated using random seed sampling for diffusion models.

T2I-CompBench Categories Figure 13 shows Best-of-N results for individual categories in T2I-CompBench [33]. Our metric is most effective for complex prompts, whereas the VQAScore excels in spatial relationships.

E.3. Winoground

We use the Winoground dataset to benchmark performance on visio-linguistic compositional reasoning in Table 9. Winoground comprises 400 examples, each containing two image-text pairs where the texts use the same words in different orders to convey different meanings. Performance is measured by how often a metric matches the correct image with its corresponding text. Surprisingly, CycleReward variants, trained solely on self-supervised rewards, outperform all metrics trained on expert human annotations. All CycleReward variants are better at selecting text for an image (text score) than selecting images from a given description (image score). While our method outperforms CLIPScore and raw cycle consistency, VQAScore outperforms all other metrics. Note that VQAScore benefits from LLM scale (x6 and x24 larger than other methods). Additionally, our model is trained on visual descriptions instead of reasoning tasks, unlike the CLIP-FlanT5 model used in VQAScore.

Method	Winoground Text Score	Winoground Image Score	Winoground Group Score
Vision-language model			
CLIPScore	28.50	11.20	8.25
VQAScore (3B)	48.75	46.25	35.50
VQAScore (11B)	**58.50**	**56.25**	**44.75**
Human preferences			
HPSv2	26.75	10.50	8.25
PickScore	23.75	12.50	6.75
ImageReward	43.00	15.25	12.75
Cycle consistency			
Raw Cycle Consistency	29.00	17.50	13.50
CycleReward-T2I	40.00	18.50	14.75
CycleReward-I2T	41.50	14.75	11.50
CycleReward-Combo	43.25	16.75	13.25

Table 9. **Winoground results.** Although we do not train on compositional reasoning tasks, CycleReward outperforms models trained on human preferences and raw cycle consistency. VQAScore, based on a large-scale VLM, outperforms all other metrics.

E.4. More Ablations

We study additional ablations on CycleReward-I2T trained on image-to-text comparison pairs. (1) *Objective Function*: We apply MSE loss to directly regress the cycle consistency score. Surprisingly, this results in a severe performance drop. We hypothesize that Bradley-Terry loss [65, 80] better captures relative preferences effectively, while MSE focuses on regressing exact score values. (2) *Dataset Size*: We maintain all configurations but train on a subset of DCI 1K images. The performance gap highlights the efficacy of scaling our dataset. (3) *Dataset Filtering*: We train a model without dataset filtering, which causes a small performance drop on alignment evaluation, with a larger decrease for Best-of-N selection (Appendix C). We believe discarding noisy comparison pairs helps select better candidates as the sample pool expands.

Ablation	DetailCaps-4870	GenAI-Bench
Best variant (CR-I2T)	**58.02**	**53.49**
MSE loss	41.87	40.57
1K images	52.86	44.39
Without filtering	57.28	51.92

Table 10. **Effect of objective function, data size, and filtering.** Choices used by our model are in gray.

E.5. DPO

Figure 14 shows comparisons between the base Stable Diffusion 1.5 model, the Diffusion-DPO model trained with Pick-a-Pic v2, and the Diffusion DPO model trained with our CyclePrefDB-T2I dataset. Training with cycle consistency preferences achieves comparable results as training with Pick-a-Pic v2, despite lacking human labels. Furthermore, our dataset is about half the size of Pick-a-Pic v2.

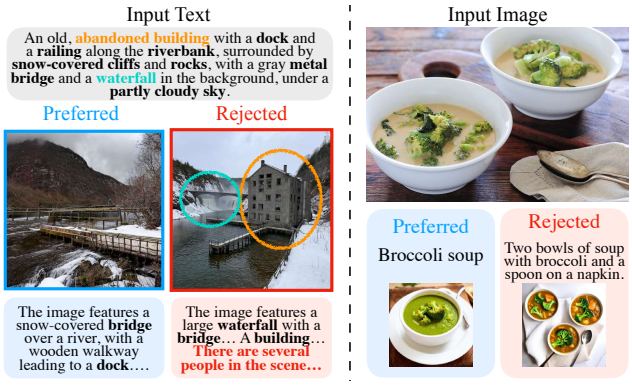

Figure 15. **Failure cases.** (Left): Despite being faithful to the input text, the right image is rejected as the reconstructed text contains hallucination inconsistent with the original prompt. (Right): The short caption is preferred over the descriptive caption due to an error in text-to-image generation. Under each caption we show the corresponding reconstructed images.

E.6. Failure Cases

Although we propose cycle consistency as a self-supervised signal for learning image-text alignment, this method has several limitations. A common source of failure is poor reconstructions which mislead preferences determined by cycle consistency seen in Figure 15. Our method also inherits biases from the underlying models used for reconstructions and and similarity measurements. Stable Diffusion 3 has a 77-token limit which limits consideration of longer texts, and LLaVA-1.5-3B can be prone to hallucinations. DreamSim often favors images with similar foregrounds over backgrounds [22], and SBERT is sensitive to text style. Furthermore, we observe worse text-to-image performance, which may partially stem from dataset differences. HPSv2, PickScore, and ImageReward are trained on prompts from real users often describing artwork, whereas CycleReward is trained on LLM-summarized descriptions for natural images. Moreover, cycle consistency primarily considers preservation of information, while other aspects such as aesthetics or style may also affect human preferences. Future work could address these challenges by improving reconstruction quality, prompt diversity, and applying cycle consistency in different scenarios. More broadly, our framework offers a general approach for learning dense alignment between two modalities, and could be extended to other domains such as audio-text, video-language, or even reasoning tasks.

F. Reward Model Trends

We investigate how text and image properties affect different metrics' alignment preferences for the following factors: caption density, object hallucination, image density, and resolution in Figure 16. For each specific factor, we plot

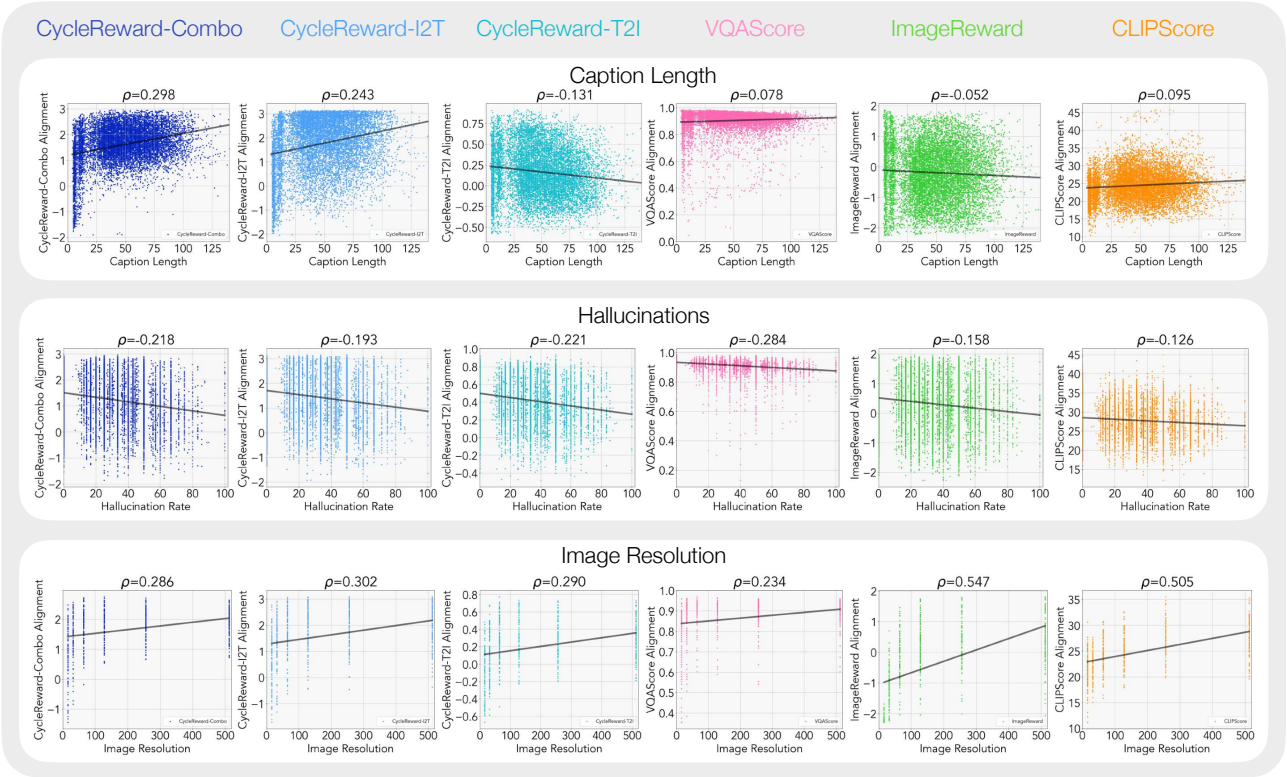

Figure 16. **Text and image data trends for different alignment metrics.** For each metric, we plot how various factors (shown on each row) affect alignment scores. Note that different alignment measurements are not comparable by scale, but their correlation with each specific factor can be measured. CycleReward-I2T and CycleReward-Combo tend to prefer longer captions, while models trained with text-to-image comparison pairs (ImageReward and CycleReward-T2I) generally prefer shorter captions. In terms of number of hallucinations and image operations, we find that all metrics show consistent correlation directions, albeit some metrics such as VQAScore and CycleReward exhibit greater sensitivity to text inaccuracies.

the alignment score for individual image, text pairs based on the relevant image or text characteristic. The title of each plot reports the Pearson correlation coefficient between the alignment score and respective factor. We also display the line of best fit. Note that the scale and range of alignment scores are different and not comparable between metrics. Because of this we instead focus on overall trends and correlations between each factor and alignment.

Caption Length To examine which reward models generally prefer long or short captions, we first create a dataset of images paired with captions of various lengths. We utilize the test and validation sets of the DCI [84] dataset for this task, where each image is paired with a long, descriptive text. For each image, we use an LLM (Meta-Llama-3.1-8B-Instruct [19]) to create captions of different lengths but asking for summaries with different numbers of words, similarly to Huh et al. [35]. We ask for summaries of lengths 5, 10, 20, ..., 100 words, and sample 5 different captions for each length with temperature 0.6 and top p 0.9. This results in 11241 unique image, caption pairs after eliminating duplicates and removing "here is a summary" text.

In Figure 16(top row), we plot the alignment trend for different metrics versus caption length. The Pearson correlation coefficient is reported at the top of each plot. Note that the alignment score scales are different for each metrics and therefore not directly comparable. Instead, we focus on the overall trends. Because captions can be informative or contain mistakes regardless of their lengths, we expect these plots to be noisy. All methods, except for CycleReward-I2T and ImageReward, have positive Pearson Correlation coefficients - meaning they in general longer captions are preferred. However the correlation between caption length and alignment is much weaker for VQAScore and CLIP compared to CycleReward-Combo and CycleReward-I2T.

Hallucination Rate To view how hallucinations affect alignment preferences, we use the M-HalDetect dataset [27]. This dataset contains images paired with captions from InstructBLIP [11]. We use the validation and training sets for this dataset totaling 14143 image caption pairs. Each caption is divided into sections which have been annotated for their accuracy and having hallucinations. We compute the fraction of hallucinated parts in each caption

and plot this value against the alignment in Figure 16(middle row). All metrics tend to prefer captions with less hallucinations (lower hallucination rate), although with different correlation strengths - VQAScore having the strongest correlation followed by CycleReward-T2I and CycleReward-Combo.

Image Resolution For text-to-image, we examine how images of different resolutions affect alignment with the text. To this end, we gather 100 "upsampled" text descriptions created by prompting GPT-4o[61] to add details to short captions from MSCOCO [51]. Text descriptions are encouraged to be visually informative and no longer than 77 tokens. We use SDXL [73] to generate images for each text description at 512×512 resolution. We resize the images to resolutions 256, 128, 64, 32, 16 and compute alignment at each stage in Figure 16(bottom row). For all metrics, alignment is generally not affected when resizing from 512 to 256 and 128 pixels, and then drops off steeply as the resolution goes from 64 to 16. Note that CycleReward and ImageReward preprocess images to be size while CLIP and VQAScore preprocessing resizes image to 336.