

# SUPPLEMENTARY MATERIALS FOR LEVERAGING UN- PAIRED DATA FOR VISION-LANGUAGE GENERATIVE MODELS VIA CYCLE CONSISTENCY

**Anonymous authors**

Paper under double-blind review

## A ADDITIONAL QUALITATIVE RESULTS

**Image-to-Text and Text-to-Image Generation.** In Figure 1, Figure 2 and Figure 3, we show the performance of ITIT on text-to-image generation and image captioning. The model uses ViT-H as the backbone and is trained with CC3M as paired data and WebLI as unpaired data for 1.5M steps. As shown in the results, our model can generate realistic images from text prompts, and can also generate accurate captions for the input image.

**Cycle Generation.** We include more cycle-generation results in Figure 4. With ITIT, the generated results are quite consistent with the original input. Without the cycle consistency loss, the generated text/image can easily miss some key information in the input image/text, causing the cycle-generation results to diverge from the original input. This demonstrates that the proposed cycle consistency loss forces the cycle generation to be more consistent and improve the input-output alignment for both text-to-image and image-to-text generation.

## B IMPLEMENTATION DETAILS

In this section, we include our implementation details, including hyper-parameters, model architecture, and training paradigm. We will also release our code for better reproducibility.

**Image Tokenizer and Detokenizer.** We use a CNN-based VQGAN encoder to encode the 256x256 input images to 16x16 feature maps. The quantizer then quantizes each pixel of the encoder’s output feature map using a codebook with 8192 entries. The detokenizer operates on the 16x16 discrete tokens and reconstructs the 256x256 image. Our VQGAN tokenizer and detokenizer are trained on the WebLI dataset with batch size 256.

**ViT architecture.** After the tokenizer, the image latent sequence length becomes 256. Since we always use a masking ratio larger than 50%, we drop 128 of the masked image tokens from the input. The tokens are then embedded with an embedding layer and concatenated with the text embedding from T5-XXL with a length of 64. We then use an image-text encoder Transformer with standard ViT architecture Dosovitskiy et al. (2021), which consists of a stack of Transformer blocks Vaswani et al. (2017), where each block consists of a multi-head self-attention block and an MLP block.

The text decoder is similar to the one used in GIT (Wang et al., 2022), which consists of 6 Transformer blocks with causal attention mask. The attention mask ensures each text token can only attend to its previous text tokens and the image tokens.

The image decoder is similar to the one used in MAGE (Li et al., 2023), which consists of 8 Transformer blocks with self-attention. The input to the image decoder is padded with the previously dropped masked image tokens. In this way, we save a lot of computation in the image-text encoder.

**Vision-language Training.** Please refer to Table 1 for our default vision-language training setting. With the online synthesis step, ITIT requires  $\sim 2x$  training time as standard I2T and T2I non-cycle training. Our ViT-H training with 1.5M steps takes  $\sim 10.9$  days on 512 TPUv3.

**Gradient scale for I2T loss:** Similar to GIT (Wang et al. (2022)), we found that the image-text encoder should receive a smaller gradient than the text decoder. Therefore, we scale down the gradient backpropagated from the text decoder to the image-text encoder by 0.1. In practice, this is

Table 1: **Pre-training Setting.**

config	value
optimizer	Adafactor (Shazeer & Stern, 2018)
peak learning rate	1e-4
weight decay	0.045
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.96$
T2I batch size	512
I2T/I2I batch size	512
T2I2T batch size	512
I2T2I batch size	512
learning rate schedule	cosine decay (Loshchilov & Hutter, 2016)
warmup steps	5000
training steps	1.5M
gradient clip	3.0
label smoothing (Szegedy et al., 2016)	0.1
dropout	0.1
image masking ratio min	0.5
image masking ratio max	1.0 (T2I), 0.75 (I2T)
image masking ratio mode	0.75
image masking ratio std	0.25

simply implemented by  $z = 0.1z + \text{stopgrad}(0.9z)$  where  $z$  denotes the output of the image-text encoder.

**Cycle Training.** During cycle training, the image-text encoder experiences the forward pass twice. We found that if we back-propagated the gradient back to it twice, the training becomes unstable. Therefore, we stop the gradient after the first forward pass of the image-text encoder, which significantly stabilize the cycle training.

**Gradient Estimation.** We use Gumbel softmax (Jang et al., 2016) with strength 1.0 for the gradient estimation in the T2I2T cycle, and use straight-through softmax for the gradient estimation in the I2T2I cycle. We found that using Gumbel softmax in the T2I2T cycle improves zero-shot CIDEr by 0.3, while using Gumbel softmax in the I2T2I cycle does not bring improvement.

**I2T Inference.** To generate captions for an image, we first extract visual features from the image tokens (and empty text tokens) using the image-text encoder. Then we auto-regressively generate the tokens, conditioned on the visual features and previously generated tokens. Following GIT (Wang et al. (2022)), we use beam search to generate the tokens with a beam size set to 4.

**T2I Inference.** To enable classifier-free guidance (Ho & Salimans, 2022) for generation, when performing I2T training with empty text embedding as input, we also train the image decoder to reconstruct missing tokens. In such cases, the T2I training becomes image-to-image (I2I) training, which is to reconstruct the original tokens from unmasked tokens.

Similar to Muse (Chang et al. (2023)), we use parallel decoding with classifier-free guidance to generate an image from a text prompt. We start with entirely masked image tokens, and concatenate them with the text prompt tokens. At each iteration, the decoder predicts a conditional logit  $l_c$  and an unconditional logit  $l_u$  for each masked token. The final logits  $l_g$  are formed by moving away from  $l_u$  by the guidance scale  $\tau$ :  $l_g = (1 + \tau)l_c - \tau l_u$ . We then sample each token using categorical sampling. After that, the corresponding prediction score of each token plus a noise sampled from a random Gumbel distribution multiplied by temperature  $t$  is used as the "confidence" score indicating the model's belief of each token prediction. We then sample the top-k tokens with the highest predicted probability, and replace the corresponding masked tokens with these sampled predicted tokens. The number of masked tokens to be replaced in each iteration follows a cosine function (Chang et al., 2022). We use a total of 24 steps to generate an image. For each model, we sweep temperature  $t$  and classifier-free guidance scale  $\tau$  for the optimal FID. In practice, we find  $t = 32$  and  $\tau = 2.0$  serve as near-optimal temperature and guidance scale.

## C DATASETS

We use three datasets in our main paper: CC3M, WebLI, and Shutterstock. CC3M contains 3.3M image-text pairs. The raw descriptions are harvested from the alt-text HTML attribute associated

with web images and then filtered to form the final dataset. WebLI (Web Language Image) contains 111 million images from the public web with image-associated alt-text labels where the image-text pairing quality is much lower than CC3M, because no filtering was applied to the alt-text data. Shutterstock contains 398 million images labeled by human annotators, which incurs significant expense and effort.

## D TRAINING SPEED

Table 2: Training speed of different variants of ITIT in steps/s. All experiments use ITIT<sub>B</sub> with total batch size of 2048 and are evaluated using a cluster of 256 TPUv4.

	T2I	I2T	T2I2T	I2T2I	Steps/s
<i>No cycle</i>					
1	✓	✓	✗	✗	3.41
<i>Half cycle</i>					
2	✓	✓	Half	✗	2.41
3	✓	✓	✗	Half	2.19
4	✓	✓	Half	Half	1.64
<i>Full cycle</i>					
5	✓	✓	Full	✗	2.25
6	✓	✓	✗	Full	2.05
7	✓	✓	Full	Full	1.47

In Table 2, we show the training speed of different variants of ITIT. All experiments are evaluated on a cluster of 256 TPUv4, with total batch size equals 2048. As shown in the table, ITIT is  $\sim 2.3\times$  slower than the non-cycle baseline. The full cycle ITIT is slightly slower than the half cycle ITIT as the gradient needs to back-propagate before the argmax. Despite doubling the training time, ITIT significantly reduce the need of collecting large-scale paired datasets. We show in our paper that ITIT trained with only 4M paired data can achieve a similar performance as baseline trained with 400M paired data, reducing the cost of collecting paired data by 100 times.

## E WARM-UP

Table 3: Warm-up steps. Incorporating cycle consistency loss at 25K steps achieves the best performance.

Warm-up steps	0	25K	50K	100K	200K
FID	14.3	<b>14.1</b>	14.3	14.6	14.8
CIDEr	31.1	<b>31.3</b>	31.3	30.9	30.5

At the beginning of the training process, the text-to-image (T2I) and image-to-text (I2T) modules struggle to generate realistic images or texts. Introducing cycle consistency loss during this initial phase would likely introduce unnecessary noise into the training process. To address this, we experimented with a warm-up training scheme, as shown in Table 3. This approach involves delaying the introduction of cycle consistency loss until the T2I and I2T modules have undergone several steps of training. We evaluated all experiments using an ITIT-B model trained for 500K steps. The results indicate that incorporating cycle consistency loss after 25K steps yields optimal performance. Interestingly, introducing the cycle loss at the very beginning (0 step) does not significantly impair

performance. This resilience is likely due to the dominance of paired data in the early training stages. As the generation results improve, the impact of cycle consistency loss becomes more pronounced, achieving greater diversity and generalization ability through the utilization of unpaired data.

## F FAILURE CASES

In Figure 5, we present various instances where the ITIT model does not perform as expected. The first notable type of failure involves the generation of text within images. Additionally, the model occasionally produces images with watermarks or margins, artifacts that can be traced back to the training data. Another challenge arises when the model attempts to generate multiple objects within a single image, often resulting in a compromise on the generation quality of each individual object.

It is important to recognize, however, that these shortcomings are not unique to ITIT; they are, in fact, common across most image-to-text generation models. Some of these issues, such as watermarks, are closely tied to the quality of the data utilized in training. Given this context, we believe that ITIT has the potential to mitigate these prevalent shortcomings by lessening the dependency on paired data in vision-language generative training. A promising direction for future research would involve training ITIT with a combination of a small but exceptionally high-quality, paired image-text dataset and extensive, high-quality unpaired image and text datasets. This approach could significantly enhance the model’s performance by providing a richer and more diverse training environment, potentially overcoming the common failure cases that currently hinder image-to-text generation models.

## REFERENCES

- Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022.
- Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. on Learning Representations (ICLR)*, 2021.
- Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.
- Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.
- Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. Mage: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2142–2152, 2023.
- Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. In *International Conference on Machine Learning*, pp. 4596–4604. PMLR, 2018.
- Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language, 2022.



Figure 1: ITIT text-to-image generation results.









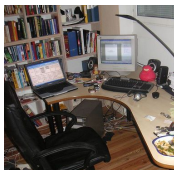
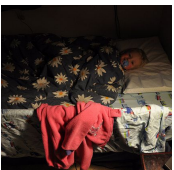
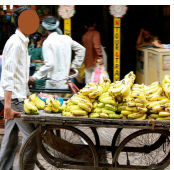
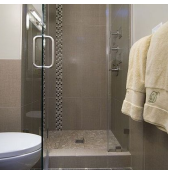
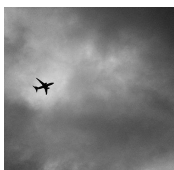

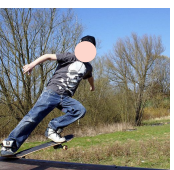

				
Baseline	person and me on the beach	person preparing food in the kitchen	a close up of a bull	tv in the living room
ITIT	person with a surfboard on the beach	preparing food in the kitchen	a bull on the farm	a tv stand in a room
.....				
				
Baseline	a homeless man sleeps on a park bench	black and white photograph of the clock tower	a picture of a baby eating watermelon	a selection of food from the menu
ITIT	a man sitting on a bench	black and white photo of the clock tower	cute baby with a birthday cake	a picture of the meal
.....				
				
Baseline	desk in the living room	a child sleeps on a bed	man selling bananas on the street	bathroom : small bathroom ideas with shower stall
ITIT	my desk at the office	person sleeping on the bed	a vendor sells bananas at a market	example of a trendy bathroom design
.....				
				
Baseline	a plane in the sky	a pizza topped with mozzarella and cherry tomatoes	a skateboarder does a trick on a skateboard	a woman dances in traditional dress
ITIT	a plane in the sky	a photo of a pizza	a young man skateboarding in a park	a girl in traditional costume

Figure 2: Image-to-text generation performance (zero-shot on COCO Captions). The baseline is trained on CC3M only. ITIT is trained on CC3M as paired data and WebLI as unpaired data. We obfuscate the faces of people in the images.

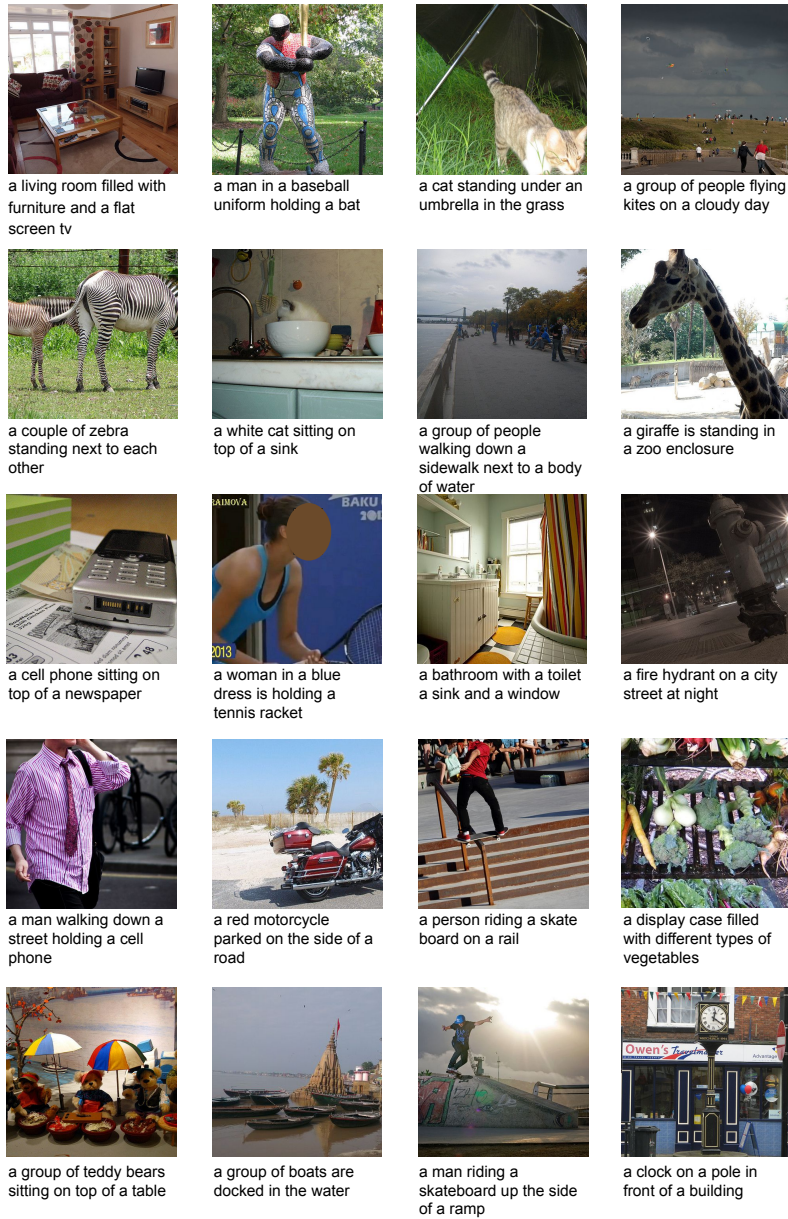


Figure 3: ITIT image-to-text generation results (fine-tuned on COCO Captions).

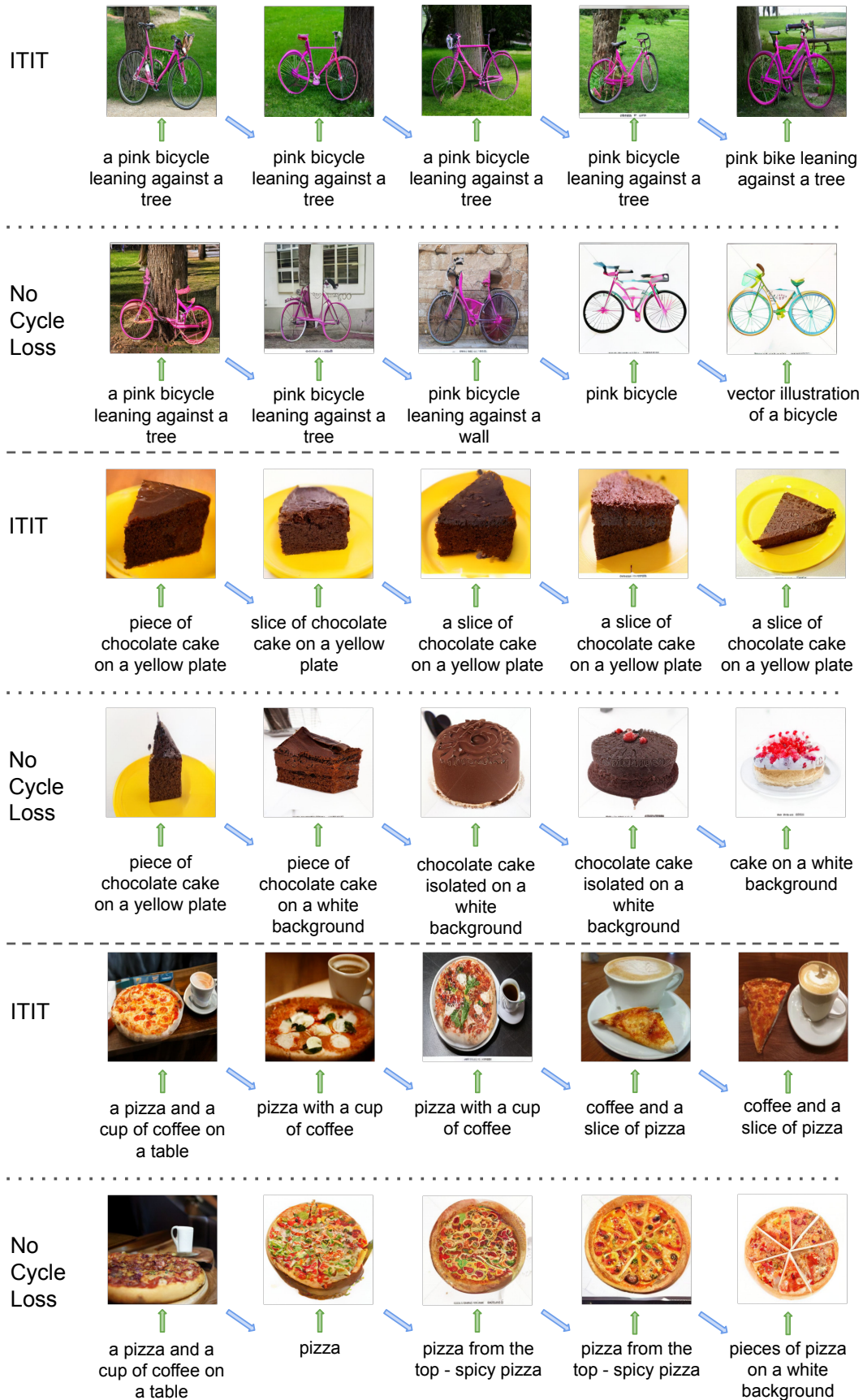


Figure 4: More cycle generation results with or without cycle consistency loss.





A bear holding a board with 'hello' written on it



A colorful slogan



A cake with 'happy birthday' on it



A t-shirt with slogan on it

(a) Generate text in image



A plate of food with noodles, carrots and broccoli



A small white cat on a large bowl



A large red bus on the side of a city street



White cake with chocolate on top

(b) Watermarks / margins in the dataset



Five cats and five dogs are playing together



Many bicycles on the street



Many fishes in a pool



Many apples and pears

(c) Multiple objects in one image

Figure 5: ITIT failure cases.