Reproducibility Study of "ITI-GEN: Inclusive Text-to-Image Generation"

Anonymous authors
Paper under double-blind review

Abstract

Text-to-image generative models often present issues regarding fairness with respect to certain sensitive attributes, such as gender or skin tone. This study aims to reproduce the results presented in "ITI-Gen: Inclusive Text-to-Image Generation" by Zhang et al. (2023a), which introduces a model to improve inclusiveness in these kinds of models. We show that most of the claims made by the authors about ITI-Gen hold: it improves the diversity and quality of generated images, it is scalable to different domains, it has plug-and-play capabilities, and it is efficient from a computational point of view. However, ITI-Gen sometimes uses undesired attributes as proxy features and it is unable to disentangle some pairs of (correlated) attributes such as gender and baldness. In addition, when the number of considered attributes increases, the training time grows exponentially and ITI-Gen struggles to generate inclusive images for all elements in the joint distribution. To solve these issues, we propose using Hard Prompt Search with negative prompting, a method that does not require training and that handles negation better than vanilla Hard Prompt Search. Nonetheless, Hard Prompt Search (with or without negative prompting) cannot be used for continuous attributes that are hard to express in natural language, an area where ITI-Gen excels as it is guided by images during training.

1 Introduction

Generative AI models that solve text-to-image tasks pose a series of societal risks related to fairness. Some of them come from training data biases, where certain categories are unevenly distributed. As a consequence, the model may ignore some of these categories when it generates images, which leads to societal biases on minority groups.

In order to tackle this issue, Zhang et al. (2023a) introduce Inclusive Text-to-Image Generation (ITI-Gen), a method that generates inclusive tokens that can be appended to the text prompts. By concatenating these fair tokens to the text prompts, they are able to generate diverse images with respect to a predefined set of attributes (e.g. gender, race, age). For example, we can add a "woman" token to the text prompt "a headshot of a person" to ensure that the person in the generated image is a woman.

In this work, we aim to focus on:

- **[Reproducibility study] Reproducing the results from the original paper.** We verify the claims illustrated in Section 2 by reproducing some of the experiments of Zhang et al. (2023a).
- **[Extended Work] Proxy features.** Motivated by attributes for which ITI-Gen does not perform well, we carry out experiments to study the influence of diversity and entanglement in the reference image datasets.
- **[Extended Work] Generating images using negative prompts.** Hard Prompt Search (HPS) (Ding et al., 2021) with Stable Diffusion (Rombach et al., 2022) is used as a baseline in the paper. We study the effect of adding negative prompts to Stable Diffusion as an alternative way of handling negations in natural language and compare the results with ITI-Gen.

- **[Extended Work] Modifications to the original code.** We improve the performance at inference by fixing a bug that prevented the use of large batch sizes, integrate ITI-Gen with ControlNet (Zhang et al., 2023b), as well as provide the code to run our proposed method that handles negations. At the same time, we include bash scripts to make it easy to reproduce our experiments.

In the next section, we introduce the main claims made in the original paper. Then, we describe the methodology of our study, highlighting the models and datasets that we used, as well as the experimental setup and computational requirements. An analysis of the results and a discussion about them will follow.

2 Scope of reproducibility

In the original paper, the authors make the following **main claims**:

1) **Inclusive and high-quality generation.** ITI-Gen improves inclusiveness while preserving image quality using a small number of reference images during training. The authors support this claim by using KL divergence and FID score (Heusel et al., 2017) as metrics.

2) **Scalability to different domains.** ITI-Gen can learn inclusive prompts in different scenarios such as human faces and landscapes.

3) **Plug-and-play capabilities.** Trained fair tokens can be used with other similar text prompts in a plug-and-play manner. Also, the tokens can be used in other text-to-image generative models such as ControlNet (Zhang et al., 2023b).

4) **Data and computational efficiency.** Only a few dozen images per category are required, and training and inference last only a couple of minutes.

5) **Scalability to multiple attributes.** ITI-Gen obtains great results when used with different attributes at the same time.

In this work, we run experiments to check the authors' statements above. Additionally, we study some failure cases of ITI-Gen and propose a method that handles negations in natural language.

3 Methodology

The authors provide an open-source implementation on GitHub[1] that we have used as the starting point. To make our experiments completely reproducible, we design a series of bash scripts for launching them. Finally, since the authors did not provide any code for integrating ITI-Gen with ControlNet (Zhang et al., 2023b), we implement it ourselves to check the compatibility of ITI-Gen with other generative models. All of these are better detailed in Section 3.4.

3.1 Model description

ITI-Gen is a method that improves inclusiveness in text-to-image generation. It outputs a set of fair tokens that are appended to a text prompt in order to guide the model to generate images in a certain way. It achieves this by using reference images for each attribute category. For example, if we use the prompt "a headshot of a person" and provide images of men and women, the model will learn two tokens, one for each gender.

The training procedure involves two losses. The first one is the directional alignment loss, which relates the provided reference images to the original prompt and inclusive tokens. In order to compare images and text, CLIP (Radford et al., 2021) is used to map them to a common embedding space. Following the original code, the ViT-L/14 pre-trained model is used. This loss is replaced by a cosine similarity loss when it is

[1] `https://github.com/humansensinglab/ITI-GEN`

undefined (i.e. when the batches are not diverse enough). The second one is the semantic consistency loss, which acts as a regularizer by making the original text prompts similar to the concatenation of the prompt and the tokens. We invite the reader to check the original paper (Zhang et al., 2023a) for a more detailed explanation of the model.

At inference, the output of the generative model will reflect the attribute categories of the fair tokens. In this way, sampling over a uniform distribution over all combinations leads to fair generation with respect to the attributes of interest. In addition, the text prompt can be the same that was used for learning (in-domain generation) or different (train-once-for-all).

The generative model must be compatible with the encoder (CLIP in this case). Following the original paper, we use Stable Diffusion v1.4 (Rombach et al., 2022) for most of the experiments. We also show compatibility with models using additional conditions like ControlNet (Zhang et al., 2023b) in a plug-and-play manner.

An alternative to ITI-Gen is HPS (Ding et al., 2021), which works by specifying the categories directly in the original prompt. For example, we could consider the prompt "a headshot of a woman" to generate a woman. One problem with this approach is that it does not handle negation properly and using a prompt like "a headshot of a person without glasses" will actually increase the chances of getting someone with glasses. However, this can be circumvented by using the *unconditional_conditioning* parameter, which is hard-coded to take in the empty string in the current Stable Diffusion sampler (PLMS). By providing the features that are not wanted we can prevent them from appearing in the generated images. For example, we can pass "eyeglasses" to get a headshot of someone without them.

A denoising step in Stable Diffusion takes an image $\boldsymbol{x}$, a timestep t and a conditioning vector $\boldsymbol{c}$. This outputs another image that we will denote by $f(\boldsymbol{x}, t, \boldsymbol{c})$. The technique involves two conditioning vectors, $\boldsymbol{c}$ and $\bar{\boldsymbol{c}}$ (the negative prompt), and outputs

$$\lambda(f(\boldsymbol{x}, t, \boldsymbol{c}) - f(\boldsymbol{x}, t, \bar{\boldsymbol{c}})) + f(\boldsymbol{x}, t, \bar{\boldsymbol{c}}),$$

where λ is the scale, that we set to the default 7.5 as we explain later in Section 3.3. We call this Hard Prompt Search with negative prompting and we will refer to it as HPSn throughout the paper.

3.2 Datasets

The authors provide four datasets[2] of reference images to train the model:

- **CelebA** (Liu et al., 2015), a manually-labeled face dataset with 40 binary attributes. For each attribute, there are 200 positive and negative samples (400 in total).
- **FAIR** (Feng et al., 2022), a synthetic face dataset classified into six skin tone levels. There are 703 almost equally distributed images among the six categories.
- **FairFace** (Karkkainen & Joo, 2021), a face dataset that contains annotations for age (9 intervals), gender (male or female) and race (7 categories). For every attribute, there are 200 images per category.
- **Landscapes HQ (LHQ)** (Skorokhodov et al., 2021), a dataset of natural scene images annotated using the tool provided in Wang et al. (2023). There are 11 different attributes, each of them divided into five ordinal categories with 400 samples per category.

3.3 Hyperparameters

In order to reproduce the results of the original paper as closely as possible, we use the default training hyperparameters from the code provided. However, the authors do not discuss the hyperparameters used for image generation. We found the scale hyperparameter λ to be the most important one after some experiments. For reproducibility reasons, we use the default value in the image generation script (i.e. $\lambda = 6$).

[2] `https://drive.google.com/drive/folders/1_vwgrcSq6DKm5FegICwQ9MwCA63SkRcr`

Table 1: **Inclusiveness with respect to single attributes.** KL divergences between the obtained and the uniform distributions over the attribute category combinations. Reference images are from CelebA. To classify the images, CLIP and manual labeling are used. The text prompt is "a headshot of a person". An extended version of these results can be found in Appendix A.

Method		Male	Young	Pale Skin	Eyeglasses	Mustache	Smiling
HPS	CLIP	0.000000	0.000000	0.000416	0.387506	0.158797	0.000046
	Reported	0.000010	0.027000	0.002800	0.371000	0.241000	0.004400
ITI-GEN	CLIP	0.000000	0.026869	0.001156	0.015053	0.280577	0.000000
	Manually labeled	0.000000	0.001156	0.000185	0.000000	0.000000	0.000000
	Reported	0.000002	0.000200	0.000000	0.000200	0.000450	0.002500

In the same way, we use $\lambda = 7.5$ for HPS and HPSn with Stable Diffusion (Rombach et al., 2022), as well as $\lambda = 9$ for ControlNet (Zhang et al., 2023b). Moreover, we generate images with a batch size of 8, which is the largest power of two that can fit in an NVIDIA A100 with 40 GB of VRAM.

3.4 Experimental setup and code

The code used to replicate and extend the experiments can be found in our GitHub repository[3]. We made two minor changes to the authors' implementation: adding a seed in the training loop of ITI-GEN to make it reproducible and fix a bug in the script for image generation to handle batch sizes larger than 1. We also provide bash scripts to replicate all our experiments easily.

Moreover, the authors do not provide any code for combining ITI-GEN with ControlNet (Zhang et al., 2023b). Thus, we implement it ourselves on top of the pre-existing ControlNet code[4] in order to validate the plug-and-play capabilities of ITI-GEN with other text-to-image generation methods. We run experiments using depth, canny edge, and human pose as additional conditions for ControlNet.

For reproducing the experiments using HPS (Ding et al., 2021), we use Stable Diffusion's code[5]. We modify the text-to-image generation script to receive an optional negative prompt for HPSn.

We consider two aspects when we evaluate the results: fairness and image quality. To measure fairness, we first generate images for every category combination. Then we classify all images using CLIP (Cho et al., 2023; Chuang et al., 2023). As explained in the original paper, this turned out to be unreliable (even when not using negation), so we manually labeled the images for some attributes. Finally, we compare the obtained distribution with a uniform distribution with respect to all attribute categories of interest by calculating the Kullback–Leibler (KL) divergence. For quantifying the image quality, we computed the Fréchet Inception Distance (FID) score (Heusel et al., 2017; Parmar et al., 2022), which compares the distribution of generated images with the distribution of a set of real images. The score depends on the number of images involved, so, for reproducibility reasons, we used the same amount of generated images mentioned by the authors (i.e. approximately 5,000).

3.5 Computational requirements

We perform all experiments on an NVIDIA A100 GPU. Training a single attribute for 30 epochs takes around a minute for CelebA, 3 minutes for LHQ, 4 minutes for FAIR and less than 5 minutes for FairFace. For the four datasets, we use 200 images per category (or all of them if there are less than 200). Generating a batch of 8 images takes around 21 seconds (less than 3 seconds per image). It is also possible to run inference on an Apple M2 chip, although it takes more than 30 seconds per image. In total, our reproducibility study adds up to at most 20 GPU hours.

[3] `https://anonymous.4open.science/r/iti-gen-reproducibility`
[4] `https://github.com/lllyasviel/ControlNet`
[5] `https://github.com/CompVis/stable-diffusion`

Table 2: **Image generation quality.** FID scores for CelebA dataset (lower is better). As reference dataset, we use the Flickr-Face-HQ Dataset (FFHQ) (Karras et al., 2019). We set different seeds to generate images, which might explain why we get a better score (the authors do not report their generation method).

Method	Number of images	FID score	
		Ours	Reported
Vanilla Stable Diffusion	5,040	78.34	67.40
HPS	5,040	70.55	–
HPS (with negative prompting)	5,040	65.08	–
ITI-Gen (30 epochs)	5,040	54.86	60.38

All our experiments were run on a GPU cluster that has Power Usage Effectiveness (PUE) of 1.19. Thus, using the Machine Learning Emissions Calculator (Lacoste et al., 2019), we obtain that our experiments emitted around 6 kg of CO_2.

4 Results

Our reproducibility study reveals that the first four claims mentioned in Section 2 are correct, whereas the last one does not hold when the number of attributes increases. In this section, we first highlight the results reproduced to support or deny the main claims of the authors. Then, we compare HPSn with ITI-Gen for the cases when ITI-Gen struggles to generate accurate images.

4.1 Results reproducing original paper

4.1.1 Inclusive and high-quality generation

To verify the first claim, we train ITI-Gen on every attribute of the CelebA dataset and generate 104 images (13 batches of size 8) per category. Considering there are 40 binary attributes, that makes a total of $40 \times 2 \times 104 = 8,320$ images.

For every attribute we have $104 \times 2 = 208$ images that are classified using CLIP. Since this classification is not very reliable, we also label the images of selected attributes manually. After that, we compute the KL divergence. In Table 1 we see how HPS struggles with attributes that require negation (e.g., "a headshot of a person with no eyeglasses"), while ITI-Gen performs well.

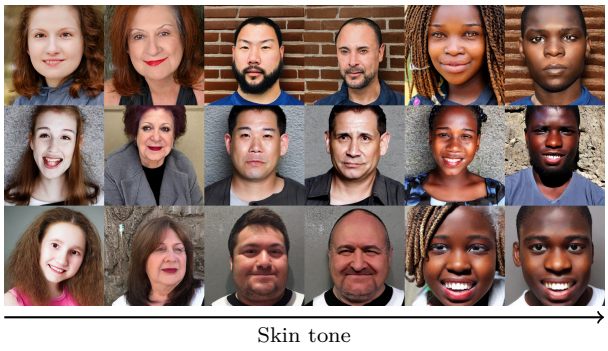
Since the FID score reported in the paper uses around 5,000 images, and we have 8,320, we decide to compute the FID score using the last 63 images of every category, which results in a total of 5,040 images. The results are in Table 2, where we can see that ITI-Gen obtains the best score.

4.1.2 Scalability to different domains

Figure 1 shows that it is possible to apply ITI-Gen to multiple domains, as the second claim holds. For the human faces generated with the "Skin tone" attribute, we are able to compute the FID score in the same way as before (using the FFHQ dataset), and obtain 46.177. For the natural scenes generated with the "Colorful" attribute, we compute the FID score by comparing the generated images with the training ones, obtaining 62.987. Note that both figures are within a similar range as the ones in Table 2.

4.1.3 Plug-and-play capabilities

We perform a number of experiments to verify the third claim, demonstrating in the end that it holds. More specifically, we show that inclusive tokens learned with one prompt can be applied to other (similar) prompts without retraining the model, as it is shown in Figure 2. We train the binary "Age" and "Gender" tokens with the prompt "a headshot of a person" using CelebA dataset and apply them to "a headshot of a firefighter" and "a headshot of a doctor" prompts. However, while the generated images are still diverse,



(a) "Skin tone" applied to "a headshot of a person".

(b) "Colorful" applied to "a natural scene".

Figure 1: **Scalability to different domains.** Images generated with ITI-GEN in two different domains: human faces and natural scenes. Each column corresponds to a different category of the attribute.

$D_{\mathrm{KL}} = 0.000800$

(a) "Age" token applied to "a headshot of a firefighter".

$D_{\mathrm{KL}} = 0.000000$

(b) "Gender" token applied to "a headshot of a doctor".

Figure 2: **Plug-and-play capabilities.** ITI-GEN is used to learn inclusive tokens for "Age" and "Gender" using the text prompt "a headshot of a person". These tokens are then applied to other similar text prompts.

their quality diminished. Thus, we obtain slightly higher FID scores, i.e., 152.94 and 157.6 for the firefighter and doctor examples, respectively.

In addition, we run experiments to illustrate the compatibility of ITI-GEN with ControlNet (Zhang et al., 2023b) in a plug-and-play manner. Figure 3 depicts the desired behaviour: ITI-GEN improves diversity of ControlNet by using the inclusive tokens. We train the "Age" token with "a headshot of a person" prompt using FairFace dataset, which has 9 categorires for this attribute, and apply it to "a headshot of a famous woman" prompt. For additional results, we invite the reader to check Appendix B.

4.1.4 Data and computational efficiency

In order to verify the claim about data efficiency, we generate images for the attributes "Colorful", "Age", and "Skin tone" using different number of reference images per category. As in the original paper, we find ITI-GEN to be robust in the low-data regime. The reader can refer to Appendix D for more information. Regarding computational efficiency, training takes less than 5 minutes and generation takes around 2 seconds per image on our hardware, as we show in Section 3.5.

4.1.5 Scalability to multiple attributes

Table 3 illustrates how ITI-GEN performs for some combinations of binary attributes. We can observe that while the KL divergence is low when we use two attributes, it increases significantly when we keep adding attributes. Thus, this claim is not entirely correct, as we further investigate this observation qualitatively and quantitatively in the next section, where we also highlight more of ITI-GEN's pitfalls.

Figure 3: **Compatibility with ControlNet.** We generate images using the prompt "photo of a famous woman" and human pose (left) as additional condition. The attribute of interest is "Age", which is trained using the text prompt "a headshot of a person".

Table 3: **Inclusiveness with respect to multiple attributes.** Comparison of the KL divergence between the obtained and the uniform distributions. All reference images are from CelebA dataset. The text prompt is "a headshot of a person". We use CLIP to classify images. We also include the results for HPS with negative prompting, which we further discuss in Section 4.2.

Method		Male × Young	Male × Young × Eyeglass	Male × Young × Eyeglass × Smile
HPS	CLIP	0.000990	0.125403	0.032695
	Authors	0.003500	0.399000	0.476000
HPSn	CLIP	0.000990	0.000475	0.000273
ITI-GEN	CLIP	0.051310	0.378709	0.356023
	Authors	0.000130	0.061000	0.094000

4.2 Results beyond original paper

4.2.1 Proxy features

ITI-GEN might use some undesired features as a proxy while learning the inclusive prompts for certain attributes. This is the case for attributes like "Bald", "Mustache" and "Beard", which seem to use "Gender" as proxy. One could argue that this is not important as long as the model is able to generate people with and without the desired attribute. However, it is actually a problem because as we show in Figure 4, some pairs of attributes are strongly coupled and ITI-GEN is not able to generate accurate images for all elements in the joint distribution. HPSn, on the other hand, is able to disentangle the attributes. More examples are shown in Appendix C.

Our main hypothesis is that the reason for this lies within the reference images. If we inspect them, we see that most of the bald people are men, which may be the reason why the model is using gender as a proxy. To delve into this, we perform an experiment using two variants of the "Eyeglasses" dataset (see Table 4), which is diverse and works fine when combined with the "Gender" attribute. Then, we test the ability of the model to learn the joint distribution using those reference images.

Table 4: **Non-diverse "Eyeglasses" reference datasets. (a)** is the original dataset form CelebA. In **(b)**, the negative samples are only of women, and, the positive ones, of men. In **(c)**, only men are included.

	Negative	Positive
(a) Original	Men without eyeglasses Women without eyeglasses	Men with eyeglasses Women with eyeglasses
(b) Gender-biased	Women without eyeglasses	Men with eyeglasses
(c) Male-only	Men without eyeglasses	Men with eyeglasses

$D_{\mathrm{KL}} = 0.000178$ (a) ITI-GEN on "Bald" attribute.

$D_{\mathrm{KL}} = 0.401088$ (b) ITI-GEN on "Gender" × "Bald".

$D_{\mathrm{KL}} = 0.009371$ (c) HPSn on "Gender" × "Bald".

Figure 4: **Proxy features are used by the model for certain attributes. (a)** Generated samples using the fair tokens for "Bald". All positive samples are men, whereas all negative samples are women, which indicates that "Gender" might be used as a proxy feature. **(b)** When combining the "Gender" and "Bald" attributes, ITI-GEN fails to generate samples of bald women. **(c)** HPS with negative prompting is able to accurately generate bald women. In (b) and (c), the KL divergence is computed using 104 manually labeled samples.

$D_{\mathrm{KL}} = 0.034039$ (a) Original "Eyeglasses" dataset.

$D_{\mathrm{KL}} = 0.237189$ (b) Gender-biased "Eyeglasses" dataset.

$D_{\mathrm{KL}} = 0.001636$ (c) Male-only "Eyeglasses" dataset.

Figure 5: **Ablation study on the diversity of reference datasets.** Generated images for all category combinations of the "Gender" and "Eyeglasses" attributes for all variations of the "Eyeglasses" reference datasets introduced in Table 4 .The complete reference dataset for the "Gender" attribute is used in the three cases. In the three cases, the KL divergence (D_{KL}) is computed using 104 manually labeled samples 4.

Figure 5a illustrates with quantitative and qualitative results how ITI-GEN correctly handles the combination of the "Gender" and "Eyeglasses" attributes using the original reference dataset. If we alter the diversity of the dataset by entangling both attributes, the model fails to learn the "Eyeglasses" attribute and uses "Gender" as a proxy feature, as shown in Figure 5b (the generated samples of women with eyeglasses are just men). However, if only men are included in the reference images, the method does learn the feature correctly (Figure 5c), because the only difference between the negative and positive samples are the glasses, and not the gender. This comparison demonstrates that we must be careful choosing the reference images, as the model might not learn what we expect, but the most salient difference between the datasets.

4.2.2 Handling multiple attributes

Computational complexity. The training loop iterates through all reference images, which implies a linear complexity with respect to the size of the training set. At the same time, it considers all category pairs within an attribute, which means it is quadratic with respect to that. It also iterates through all elements in the joint distribution, so it is exponential with respect to the number of attributes. This can be problematic when we train ITI-GEN on many attributes. For example, in the case of binary attributes with the same number of reference images, the time complexity is $O(Ne^N)$, as Figure 7 depicts.

Diversity issues in generated images. Figure 6 illustrates a comparison between ITI-GEN and HPSn. We can observe that ITI-GEN struggles significantly to generate diverse images (the difference between old and young people is almost non-existent, there are many people without eyeglasses, etc.). On the other hand, images generated by HPSn reliably reflect all category combinations.

This behaviour can also be observed in our quantitative results. More specifically, we compute the KL divergence using 104 images in every category combination. The numerical results are displayed in Table 3,

on which we can observe that ITI-Gen obtains a KL divergence of approximately 0.3560, while HPSn's is almost equal to 0.

(a) ITI-Gen.

(b) HPS with negative prompting.

Figure 6: **Qualitative results for multiple attributes.** ITI-Gen and HPSn are compared on generating diverse images with respect to four binary attributes.

5 Discussion

In this work, we conduct several experiments to check the validity of the main claims from the original paper. We find that most of them are correct with some small exceptions.

First, we show ITI-Gen (Zhang et al., 2023a) is able to generate diverse high-quality images, according to the low KL divergence and FID scores we obtained (see Tables 1 and 2). We also highlight that it works on multiple domains, such as human faces and landscapes, as well as with different text-to-image generators (i.e. Stable Diffusion (Rombach et al., 2022), ControlNet (Zhang et al., 2023b)), as displayed in Figures 1, 2, 3. In addition, we verify that only a few dozen reference images are required, and that training is computationally efficient (Figure 10).

On the other hand, our analysis reveals that ITI-Gen can struggle to disentangle certain attributes (e.g. "Gender" and "Bald" in Figure 4) and that training time grows exponentially with respect to the number of attributes (Figure 7). Thus, we propose HPSn which does not require training and produces more accurate images. At the same time, it seems to handle negations better than ITI-Gen , especially when we use a large number of attributes (Figure 6, Table 1). However, a limitation of HPSn is that it cannot handle attributes that are hard to specify with text (e.g. skin tone, colorfulness), whereas ITI-Gen excels on these.

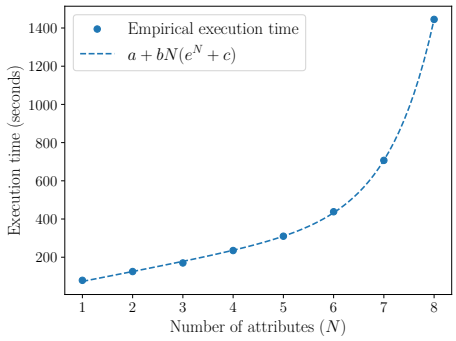


Figure 7: **Computational complexity.** Average training time across 3 executions of ITI-GEN with respect to the number of binary attributes. 400 reference images are used for every attribute, equally distributed between the positive and negative categories.

5.1 What was easy

The paper was well-written and included many examples to make it easier to understand. Additionally, the original code was published on GitHub, with instructions on how to run the training, generation and evaluation scripts.

5.2 What was difficult

The main difficulty consisted in understanding the code, since there were some differences with the paper. More specifically, when the directional loss is undefined, it is replaced by a cosine similarity loss between the image and text embeddings.

Moreover, the evaluation script requires a list of classes that is not specified. The authors mention that they had to change the text prompts to tackle the negative prompt issue. They also had to use pre-trained classifiers combined with human evaluations. Thus, it was not possible for us to easily reproduce the KL divergence results.

5.3 Communication with original authors

We reached out to the authors via email correspondence regarding the quality of the generated images and the missing code for combining ITI-GEN with ControlNet. They replied quickly and cleared up the discrepancies in our results. However, we did not receive from the authors any additional code, since they were still working on uploading it to their GitHub repository.

References

Jaemin Cho, Abhay Zala, and Mohit Bansal. Dall-eval: Probing the reasoning skills and social biases of text-to-image generation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Ching-Yao Chuang, Varun Jampani, Yuanzhen Li, Antonio Torralba, and Stefanie Jegelka. Debiasing vision-language models via biased prompts. *arXiv preprint arXiv:2302.00070*, 2023.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 2021.

Haiwen Feng, Timo Bolkart, Joachim Tesch, Michael J Black, and Victoria Abrevaya. Towards racially unbiased skin tone estimation via scene disambiguation. In *ECCV*, 2022.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017.

Kimmo Karkkainen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age. In *WACV*, 2021.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2019.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *CoRR*, 2019.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Ivan Skorokhodov, Grigorii Sotnikov, and Mohamed Elhoseiny. Aligning latent and image spaces to connect the unconnectable. In *ICCV*, 2021.

Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023.

Cheng Zhang, Xuanbai Chen, Siqi Chai, Henry Chen Wu, Dmitry Lagun, Thabo Beeler, and Fernando De la Torre. ITI-GEN: Inclusive text-to-image generation. In *ICCV*, 2023a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023b.

A KL Divergences on the CelebA dataset

Table 5 shows the performance on diversity of ITI-Gen and the baseline methods on binary attributes.

Table 5: **Inclusiveness with respect to single attributes.** KL Divergence for all attributes from CelebA dataset. We include results for ITI-Gen trained for 10, 20 and 30 epochs. The generated images (208 per attribute) were classified using CLIP, and, for some attributes, also manually (last column).

Attribute	SD	HPS	HPSn	ITI-Gen			
				10 epochs	20 epochs	30 epochs	30 (man.)
5 o'Clock Shadow	0.007833	0.076791	0.016782	**0.000185**	0.007833	0.008748	-
Arched Eyebrows	0.355570	0.120493	**0.011881**	0.101287	0.072061	0.083041	-
Attractive	0.072061	**0.001156**	0.036694	0.158797	0.130812	0.203595	-
Bags Under Eyes	0.325839	0.125592	**0.001156**	0.255716	0.120493	0.034089	-
Bald	0.530124	0.048114	**0.000000**	0.042202	0.068324	0.101287	0.000178
Bangs	0.263824	0.263824	**0.000000**	0.000416	0.002962	0.003749	-
Big Lips	0.158797	0.064693	0.064693	0.001156	**0.000416**	0.002267	0.008915
Big Nose	0.355570	**0.013420**	**0.013420**	0.280577	0.316377	0.272110	0.201355
Black Hair	0.316377	0.110655	**0.002962**	0.196770	0.263824	0.232416	0.020136
Blond Hair	0.638921	0.020527	**0.001665**	0.007833	0.006672	0.007833	-
Blurry	**0.016782**	0.217696	0.022544	0.036694	0.020527	0.031584	-
Brown Hair	0.421958	0.158797	**0.003749**	0.272110	0.224977	0.217696	-
Bushy Eyebrows	0.105913	0.376598	**0.000046**	0.002962	0.016782	0.011881	-
Chubby	0.298080	0.514949	**0.000416**	0.115515	0.164783	0.183554	-
Double Chin	0.272110	0.446529	0.002267	0.031584	0.026869	**0.001665**	-
Eyeglasses	0.398692	0.387506	**0.000000**	0.016782	0.020527	0.015053	0.000000
Goatee	0.345427	0.064693	**0.010438**	0.170903	0.158797	0.170903	-
Gray Hair	0.693147	0.365957	**0.000046**	0.079859	0.088094	0.092379	0.035989
Heavy Makeup	0.434070	0.136156	**0.000185**	0.096776	0.064693	0.045107	0.113232
High Cheekbones	0.472577	**0.003749**	0.096776	0.434070	0.335520	0.355570	-
Male	0.000740	**0.000000**	**0.000000**	**0.000000**	**0.000000**	**0.000000**	0.000000
Mouth Slightly Open	0.083921	0.110655	0.110655	0.007833	**0.000046**	0.000185	-
Mustache	0.545925	0.158797	**0.000000**	0.280577	0.280577	0.280577	0.000000
Narrow Eyes	0.136156	0.051223	0.026869	0.026869	**0.000000**	0.006672	0.000000
No Beard	0.232416	0.500334	0.693147	**0.013420**	0.031584	0.042202	0.000000
Oval Face	0.376598	**0.010438**	0.170903	0.240016	0.101287	0.170903	-
Pale Skin	0.196770	**0.000416**	**0.000416**	0.004630	0.006672	0.001156	0.000185
Pointy Nose	0.500334	**0.061169**	0.164783	0.545925	0.562439	0.500334	-
Receding Hairline	0.410171	0.514949	**0.000740**	0.136156	0.141624	0.141624	
Rosy Cheeks	0.486224	0.051223	**0.024658**	0.345427	0.345427	0.421958	-
Sideburns	0.514949	**0.026869**	0.034089	0.387506	0.410171	0.562439	-
Smiling	0.280577	0.000046	0.000046	**0.000000**	**0.000000**	**0.000000**	
Straight Hair	0.057750	0.662690	**0.000000**	0.002267	0.000185	0.002267	-
Wavy Hair	0.446529	0.240016	**0.000185**	0.115515	0.072061	0.064693	-
Wearing Earrings	0.355570	0.472577	**0.000416**	0.398692	0.446529	0.421958	-
Wearing Hat	0.662690	0.545925	**0.000046**	0.042202	0.136156	0.141624	-
Wearing Lipstick	0.579782	0.500334	**0.000046**	0.247782	0.307126	0.325839	-
Wearing Necklace	0.662690	0.307126	**0.001156**	0.355570	0.355570	0.325839	-
Wearing Necktie	0.514949	0.545925	**0.001156**	0.009088	0.001665	0.002267	-
Young	0.693147	**0.000000**	**0.000000**	0.000046	0.001665	0.026869	0.001156

B Compatibility with ControlNet

We present additional results on the compatibility of ITI-Gen with ControlNet (Zhang et al., 2023b) in Figure 8.

(a) Condition: depth map. Prompt: "a headshot of a female".

(b) Condition: canny edge map. Prompt: "photograph of mount katahdin".

Figure 8: **Compatibility with ControlNet.** ITI-GEN is able to improve diversity in images generated with ControlNet given different conditions. Human images (a) are generated using inclusive tokens trained on "a headshot of a person", while scene images (b) are generated using inclusive tokens trained on "a natural scene".

C Proxy features

We provide additional examples of proxy features used by ITI-GEN in Figure 9.

$D_{\text{KL}} = 0.000000$

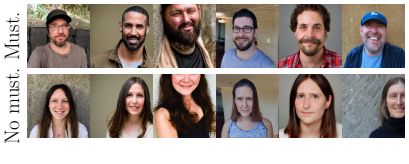
(a1) ITI-GEN on "Beard" attribute.

$D_{\text{KL}} = 0.288772$

(a2) ITI-GEN on "Gender"×"Beard".

$D_{\text{KL}} = 0.113676$

(a3) HPSn on "Gender"×"Beard".



$D_{\text{KL}} = 0.000000$

(b1) ITI-GEN on "Mustache" attribute.

$D_{\text{KL}} = 0.449763$

(b2) ITI-GEN on "Gender"×"Mustache".

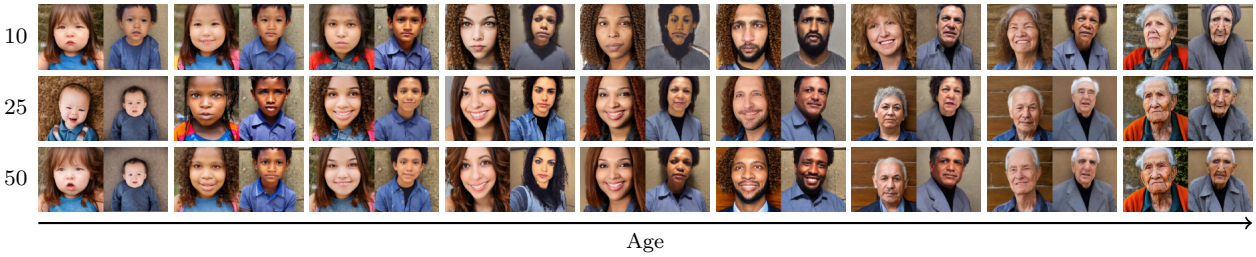
$D_{\text{KL}} = 0.187124$

(b3) HPSn on "Gender"×"Mustache".

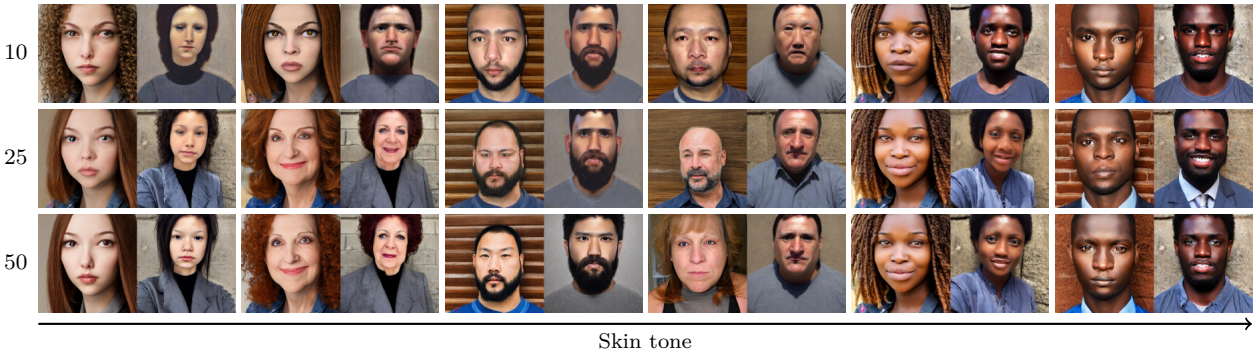
Figure 9: **Proxy features are used by the model for certain attributes.** In (a1) and (b1) we see how ITI-GEN seems to be using "Gender" as a proxy for "Beard" and "Mustache". In (a2) and (b2) we see how ITI-GEN fails to disentangle the attributes. In (a3) and (b3) we see HPSn's results. It struggles sometimes to generate women with beard, but works well with the combination of "Gender" and "Mustache". In (b) and (c), the KL Divergence is computed using 104 manually labeled samples.

D Low data requirements

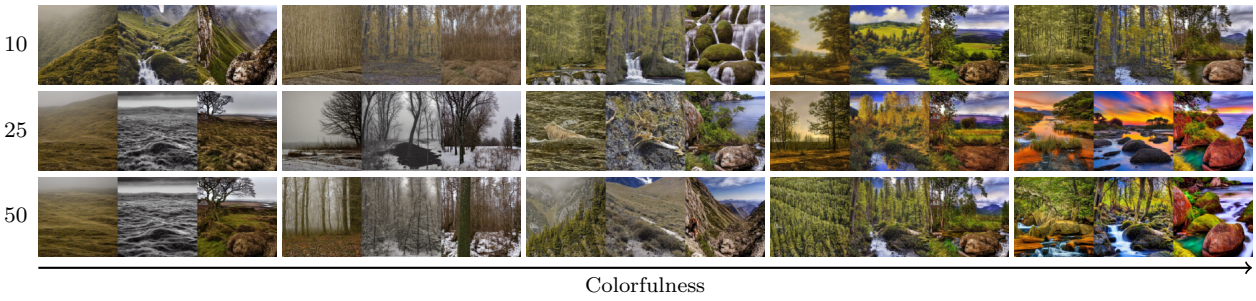
Figure 10 illustrates how ITI-GEN is able to improve diversity using only a few reference images.



(a) Prompt: “a headshot of a person”.



(b) Prompt: “a headshot of a person”.



(c) Prompt: “a natural scene”.

Figure 10: **Effect of of the size of the reference dataset.** ITI-GEN is able to improve diversity with respect to attributes such as “Age”, “Skin tone” and “Colorfulness” using only a few reference images (10, 25 and 50).