Adapting Blackbox Generative Models via Inversion

Sinjini Mitra [1] **Rakshith Subramanyam** [1] **Rushil Anirudh** [2] **Jayaraman J. Thiagarajan** [2] **Ankita Shukla** [1]
Pavan Turaga [1]

Abstract

Adapting large-scale generative AI tools to different end uses continues to be challenging, as many industry grade image generator models are not publicly available. Thus, to finetune an industry grade image generator is not currently feasible in the classical sense of finetuning certain layers of a given deep-network. Instead, we present an alternative perspective for the problem of adapting large-scale generative models that does not require access to the full model. Recognizing the expense of storing and fine-tuning generative models, as well as the restricted access to weights and gradients (often limited to API calls only), we introduce `AdvIN` (Adapting via Inversion). This approach advocates the use of inversion methods, followed by training a latent generative model as being equivalent to adaptation. We evaluate the feasibility of such a framework on StyleGANs with real distribution shifts, and outline some open research questions. Even with simple inversion and latent generation strategies, `AdvIN` is surprisingly competitive to fine-tuning based methods, making it a promising alternative for end-to-end fine-tuning.

1. Introduction

Recent successes in generative modeling (Ramesh et al., 2022; Saharia et al., 2022; Rombach et al., 2022) have brought renewed focus on the challenges and opportunities in adapting large-scale generative models. In particular, adapting generic base models to application-specific domains has emerged as a critical research problem. The *de facto* solution of model fine-tuning is often undesirable due to the exorbitantly large memory footprints of these models, data scarcity in custom domains, and limited compute availability in practical scenarios.

On the other hand, from a deployment standpoint, it is important to consider the potential implications and legal responsibilities associated with making model weights publicly accessible. With risks for misuse and compromising data privacy, it might be beneficial to allow model access only through an API call in a black-box fashion, as currently done with DALL-E (Ramesh et al., 2022) or Midjourney (Midjourney, 2022). In such a setting, it is not clear how model adaptation can be posed, and we need new approaches that do not directly require access to the base model's weights (or gradients), but can synthesize samples from a target distribution. We investigate the feasibility of such an approach, in the context of unconditional StyleGANs (Karras et al., 2019; 2020). To this end, we introduce a generic framework `AdvIN` (Adapting via Inversion) as an alternative for model fine-tuning, particularly for scenarios involving distributed-shifted and limited target data. `AdvIN` first decomposes model adaptation into two independent steps: (i) inverting target data onto the latent space of the generative model, and (ii) designing a target sampler, which is light-weight in comparison to the full generative model, to directly sample from the sub-manifold corresponding to the target data. This approach has two key advantages over model fine-tuning. First, it eliminates the need to access or update the model weights at the client, assuming that the inversion step is carried out on the server. Second, we can leverage the recent advances in GAN inversion that can now enable effective recovery of even out-of-distribution (OOD) images (Subramanyam et al., 2022; Abdal et al., 2020; Alaluf et al., 2022).

In this study, we explore different choices for implementing the inversion module, and design the target sampler as a latent diffusion model. We systematically evaluate the performance of `AdvIN`, particularly at low target sample sizes and with multiple real-world shifts. In addition to making specific contributions, we emphasize the open research questions in this space:

[1]Geometric Media Lab, Arizona State University, Tempe, USA [2]Lawrence Livermore National Laboratory, Livermore, USA. Correspondence to: Sinjini Mitra <smitra16@asu.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

This work was supported in part by the U.S. Department of Energy through the Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344; and in part by the Defense Advanced Research Projects Agency (DARPA) under Grant HR00112290073.

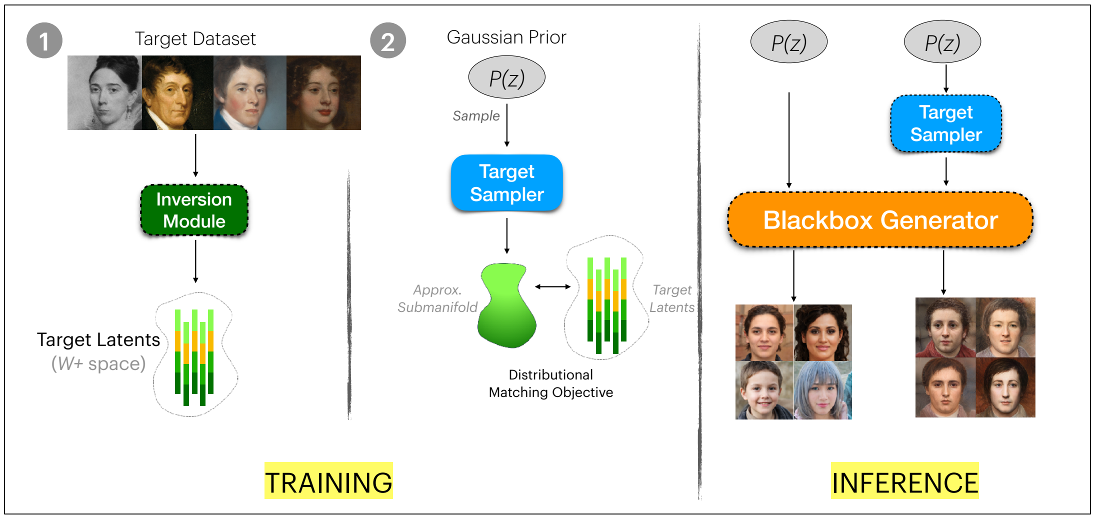

Figure 1. **AdvIN Framework.** We address the problem of adapting pre-trained generators to target distributions, without requiring access to the weights or gradients of the generator. (Left) The two-stage training protocol, where we (1) collect latents for the target samples using an inversion module, and (2) fit a target sampler, which is a latent generative model that learns to sample from the estimated posterior $P(\mathrm{z}|\mathrm{X}_{\text{target}})$; (Right) During inference, we leverage the trained target sampler to synthesize realizations from the target distribution.

(i) When the generator update is restricted, a common approach is to directly learn a latent space transformation using the discriminator loss, e.g., MineGAN++. We find that `AdvIN` is consistently superior and does not have the mode collapse issue faced by MineGAN++. However, advances to GAN inversion and its scalability can improve both quality and efficiency of `AdvIN`.

(ii) There are a number of reasons for end-to-end fine-tuning to be better than `AdvIN`, e.g., the inversion module can work poorly for large shifts or the vanilla latent diffusion can be ineffective with scarce data. Surprisingly, at the low sample sizes we consider, `AdvIN` is indeed competitive to fine-tuning. However, by adopting sophisticated regularizers and optimization techniques, the gap between fine-tuning and `AdvIN` can be further reduced.

(iii) Similar to existing fine-tuning methods, `AdvIN` inherently preserves latent properties from the base model, e.g., attribute directions, semantic interpolation in the target domain. This motivates their utility as priors for downstream tasks in the target domain.

2. Adapting via Inversion

Adaptation is the problem of approximating a target distribution $P_{\text{target}}(X)$ given access to (a small number of) target samples, $\{\mathrm{X}_i \sim P_{\text{target}}\}$, and a pre-trained generator that can sample from a base, training distribution $G(\mathrm{z}) \sim P_{\text{base}}$, where $\mathrm{z} \sim P(\mathrm{z})$ is a known prior.

Though fine-tuning is the most common approach for adaptation (Karras et al., 2020; Sauer et al., 2021), it can be impractical in reality. Through `AdvIN`, we study the effectiveness of adaptation by first inverting the target dataset onto the latent space of the base generative model, and subsequently training a target sampler that fits to the latent (target) distribution. This formulation is agnostic to the choice of inversion technique (e.g., pSp, e4e and hyperstyle) or the target sampler design. Note that, most existing inversion methods require access to gradients from the generator model. However, the inversion step can be carried out directly on the server side, without requiring the client to access the model.

Given latents for the target set, the adaptation is performed by learning to sample from the sub-manifold, $P(\mathrm{z}|\mathrm{X}_{\text{target}})$. Next, we define a target sampler $F(\mathrm{z})$ that learns to approximate $P(\mathrm{z}|\mathrm{X}_{\text{target}})$. In other words, it is a generative model in the latent space, that maps a known prior distribution to match the target latents obtained via inversion. Once trained, we can sample from the target distribution as $G(F(\mathrm{z})) \sim P_{\text{target}}$, as illustrated in Figure 1.

A closely related work to our approach is MineGAN++ (Wang et al., 2021), where an explicit miner network is used to transform the latent prior, so that the generator synthesizes samples only from the target distribution. In practice, this is optimized directly using the discriminator's loss, and one can also optionally update the generator parameters.

Target sampler design. This can be designed as any generative model of choice, that can be trained effectively with limited data. In this study, we experimented with denoising diffusion models (Ho et al., 2020; Song et al., 2020),

Table 1. **Performance Evaluation.** We report the FID↓(blue) and IS↑(black) metrics for the Toonify and MetFaces datasets. AdvIN consistently outperforms MineGAN++ w/ frozen generator across all target sizes, while closely matching the performance of MineGAN++ w/ generator fine-tuning.

	# of target images	AdvIN (pSp)	AdvIN (e4e)	AdvIN (HyperStyle)	MineGAN++ (Finetune Gen.)	MineGAN++ (Freeze Gen.)
MetFaces	100	3.00/5.82	5.10/5.12	4.01/5.33	1.57/8.08	259.73/1.08
	200	3.52/7.26	4.59/5.82	3.72/5.62	1.03/9.13	196.75/1.13
	300	2.72/8.60	4.14/6.28	3.06/6.73	0.76/11.15	151.49/1.45
Toonify	54	10.52/3.11	9.61/3.56	9.69/3.08	23.57/1.523	291.384/1.14
	108	7.13/4.81	8.66/4.79	9.69/4.50	19.88/1.77	220.04/1.66
	217	5.86/6.25	9.61/5.81	9.60/6.23	6.40/2.64	201.69/1.76

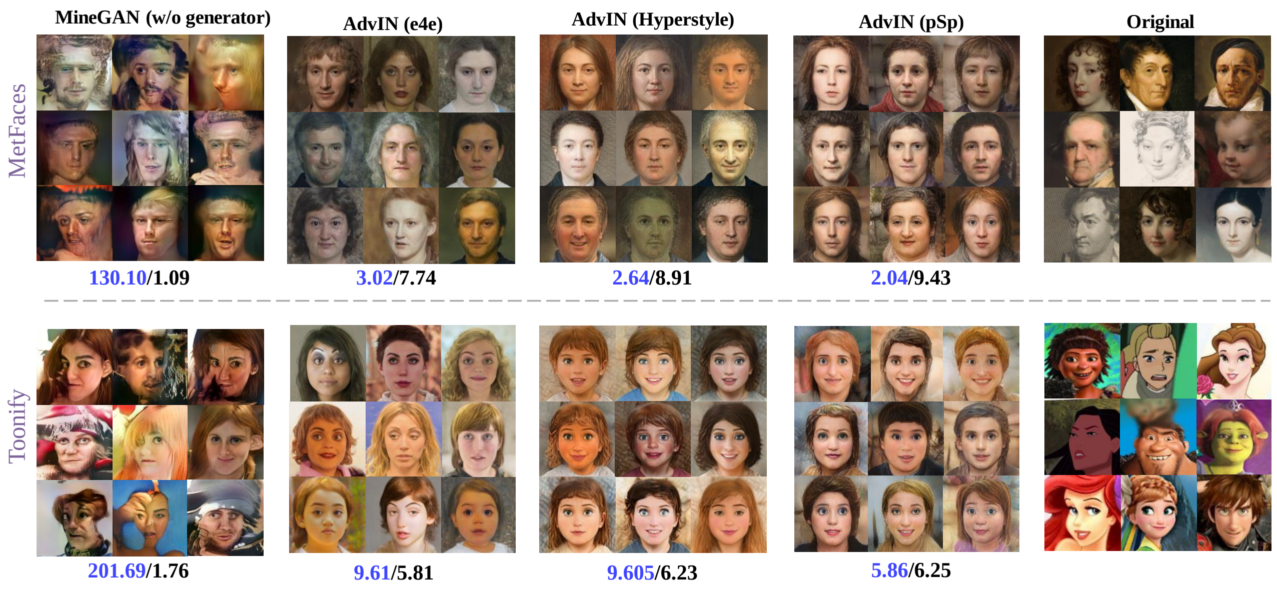

Figure 2. **Results from AdvIN.** Reconstructions and metrics (FID in blue, IS in black) are shown for three implementations of AdvIN with different inversion strategies. We also include the results for MineGAN++ (without generator update). We notice that, across different choices of inversion strategies, AdvIN converges more stably and recovers the target distributions with reasonable fidelity.

since GANs were more challenging to train in the low data regimes. Specifically, we implement a diffusion model with an 1D UNet backbone (Ronneberger et al., 2015), $F : \mathbb{R}^{18\times512} \to \mathbb{R}^{18\times512}$. The U-Net model is lightweight consisting of only 250K parameters, which is $< 1\%$ of the number of parameters in the base generator ($\sim$ 28M parameters).

3. Experiments

To understand the behavior of AdvIN, we select StyleGAN2 trained on FFHQ faces as our base generator. We investigate three widely-used inversion techniques, namely pSp (Richardson et al., 2021), e4e (Tov et al., 2021), and Hyperstyle (Alaluf et al., 2022), for the inversion module. For evaluating the adaptation performance, we utilize two OOD datasets, namely MetFaces and Toonify, which serve as the target distributions.

Baselines and Metrics. As our baseline for comparison, we adopt a variant of MineGAN++ (Wang et al., 2021), wherein we do not update the generator and the discriminator. In this baseline, the miner network (an MLP) is trained using a discriminator-based loss and is utilized as the posterior sampler at inference time. We also report the performance of MineGAN++, when the generator and discriminator are also updated along with miner training. For evaluation, we utilize FID (Fréchet Inception Distance) and IS (Inception Score) metrics to asses the quality, diversity, and similarity between the target and approximated distributions.

Training and inference. After the inversion step, we first normalize the latent codes $(\bar{z}_i - \mu)/\gamma$, where μ and γ are the mean and standard deviation of all latent codes $\{\bar{z}_i\}$. Since diffusion models are sensitive to the variance of the training

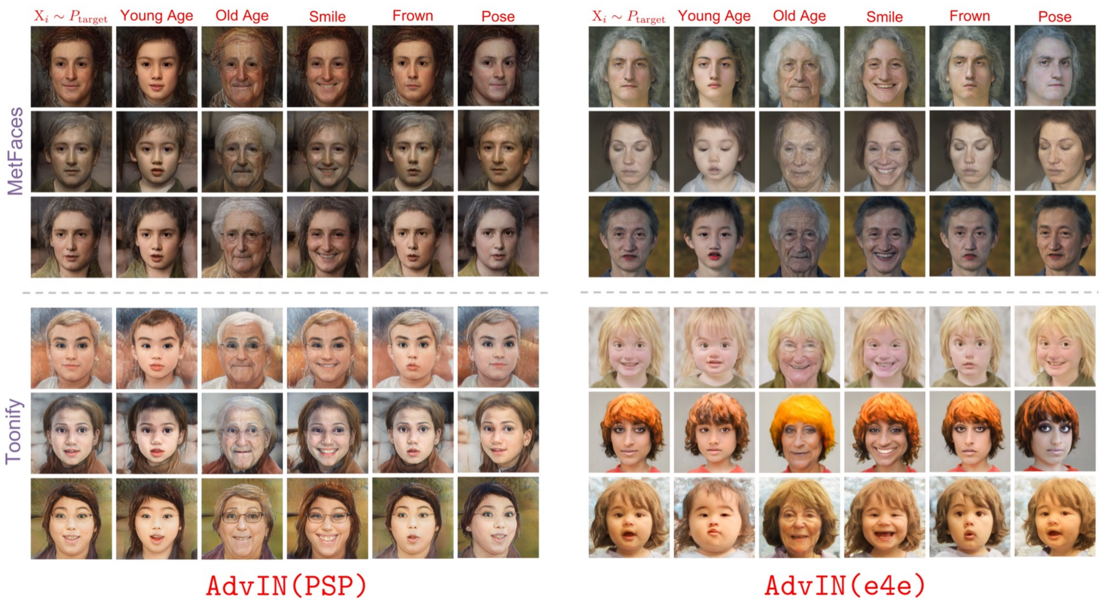

Figure 3. **Attribute preservation results on MetFaces and Toonify datasets.** We illustrate the effectiveness of `AdvIN` in preserving attributes for out-of-distribution target images. We show results on the MetFaces and Toonify datasets, demonstrating attribute manipulation using pre-specified attribute directions.

data, this normalization is crucial to ensure that the target sampler achieves sufficient diversity. For target sampler training, we construct batches of size B (see appendix A for details), such that each input to the diffusion model is of size $B \times 18 \times 512$. During inference, the latent normalization is reversed and the unnormalized latents are passed to the frozen StyleGAN2 generator to generate the images. The training details are included in Appendix A.

Results. We report quantitative metrics and qualitative samples obtained using the different `AdvIN` implementations in Figure 2 and Table 1 respectively. We observe surprisingly competitive performance even when compared to MineGAN++ with fine-tuning, while consistently outperforming MineGAN++ with a frozen generator. We also notice that for each inversion technique, the specific traits of the method are preserved in the final outcome – for instance, e4e is designed to preserve image quality for latent editing and this is reflected in the inference images for that particular method.

Preserving attribute directions. Modern GAN architectures posses capabilities to discover and manipulate semantic attributes. StyleGAN2, for example, enables attribute manipulation by modifying the values of latent vectors ($W+$) along pre-defined directions corresponding to desired attributes. However, achieving similar attribute manipulations in out-of-distribution (OOD) images is known to be challenging. Interestingly, `AdvIN` successfully preserves the attributes within the latent submanifold for OOD images, demonstrating its ability for attribute manipulation. Our results for MetFaces and Toonify datasets obtained using both pSp and e4e shown in Figure 3 clearly demonstrate the effectiveness of `AdvIN`.

4. Discussion

This study presents `AdvIN`, an effective solution to generator fine-tuning in blackbox adaptation to small target datasets. `AdvIN` displays competitive performance to end-to-end fine-tuning approaches, particularly at low sample sizes. One interesting observation we make is `AdvIN`'s ability to preserve attributes within the latent submanifold, even for out-of-distribution (OOD) images, similar to conventional fine-tuning methods. This ability for attribute manipulation presents promising opportunities for leveraging `AdvIN` as a prior to perform downstream tasks (e.g., inverse problems, data augmentation) in the target domain.

In addition to these findings, we want to emphasize some open research problems: a) Advanced inversion methods, that can effectively invert even *far* OOD data into StyleGAN, are essential to further expand the utility of `AdvIN` to arbitary target distributions, and improve the training stability in very low data regimes; b) While the presented implementation of `AdvIN` produces low FID scores, visual inspection reveals that aspects of the source domain (real faces) are still present in the synthesized target images. Potential improvements include regularizers for latent generation, and style manipulation-based augmentations to enhance the sample quality, in terms of avoiding source-domain style characteristics; c) Designing alternatives to vanilla latent diffusion can significantly impact the efficiency of `AdvIN`. In summary, AdvIN presents a promising direction for generator fine-tuning and its utility in application-specific, data constrained target domains is particularly significant.

References

Abdal, R., Qin, Y., and Wonka, P. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Alaluf, Y., Tov, O., Mokady, R., Gal, R., and Bermano, A. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF conference on computer Vision and pattern recognition*, pp. 18511–18521, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020.

Midjourney, I. Midjourney. https://www.midjourney.com/, 2022.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., and Cohen-Or, D. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2287–2296, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241. Springer, 2015.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022.

Sauer, A., Chitta, K., Müller, J., and Geiger, A. Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34:17480–17492, 2021.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Subramanyam, R., Narayanaswamy, V., Naufel, M., Spanias, A., and Thiagarajan, J. J. Improved stylegan-v2 based inversion for out-of-distribution images. In *International Conference on Machine Learning*, pp. 20625–20639. PMLR, 2022.

Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., and Cohen-Or, D. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021.

Wang, Y., Gonzalez-Garcia, A., Wu, C., Herranz, L., Khan, F. S., Jui, S., and Van de Weijer, J. Minegan++: Mining generative models for efficient knowledge transfer to limited data domains. *arXiv preprint arXiv:2104.13742*, 2021.

A. Appendix

Sampling from a sub-manifold. With a frozen generator, the task of learning the correct sub-manifold to sample from is not trivial. In (Wang et al., 2021) the authors present a simple multi-layer perceptron(MLP) to help steer the sampler to the correct regions. We use this network as the "target sampler" in Figure 1 and attempt to learn p_{target} without manipulating the generator. We find that the network is insufficient in accurately estimating the target distribution, as observed in Figure 4. Thus, there is a need to rethink the adaptation problem for generative models and this motivates us to present our novel framework.

Training details. For MetFaces we set the batch size to 32, trained for 400 epochs using the Adam optimizer with a learning rate of $1e-4$, and $\beta = 0.9$, $\beta = 0.999$. For Toonify, the batch size was set to 8, trained for 400 epochs using Adam optimizer with a learning rate of $1e-5$, and $\beta_1 = 0.5$ $\beta_2 = 0.999$. The images in Figure 2 are generated for the "full" size of each training set - 217 for Toonify, 500 for MetFaces. The subset metrics presented in Table 1 are calculated for random subsets generated from this full training set. For Toonify, the batch size = 4 for subset size

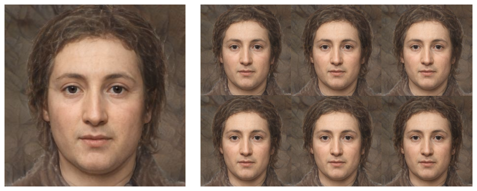

Figure 4. GAN training methods often collapse in settings with a small amount of target data. We observe this phenomenon when deploying the miner from (Wang et al., 2021) where the sampler defaults to the average mode of all input latents (left). This further demonstrates that the problem of estimating q_{target} in the latent space is not trivial and hence motivates us to rethink the problem.

= 54 and 108. For MetFaces, the batch size = 16 for subset sizes = 200 and 300, batch size = 8 for subset size = 100.