Supplementary Materials

A Implementation Details

StyleGAN2 We adopt the standard StyleGAN2 architecture[1] for 256×256 resolution images, with 8 fully connected layers in the mapping network. We keep the hyperparameters such as the learning rate, regularization weights and frequency, untouched, and only add our proposed MixDL.
DiffAug We essentially follow the official configuration[2] for *low-shot* generation, including the two-layer mapping network and three data augmentation methods. We have also tried with a standard 8 FC layer mapping network and observed significant drops in the overall performance as shown in Tab. S1.

FC layers	Obama (100-shot)	Grumpy Cat (100-shot)
2	46.87	26.52
8	71.13	38.42

Table S1: FID for DiffAug with varying number of FC layers

FastGAN We use the official FastGAN implementation[3] for 256×256 images. As FastGAN doesn't have a separate mapping network, we interpolate in $\mathcal{Z}$ space.
Diversity Preservation Methods Baselines such as *Normalized Diversification (N-Div)* [28], *Mode Seeking GAN (MSGAN)* [29] and *DistanceGAN* [4] propose distance preserving objective to combat mode collapse. We train these models with StyleGAN2 architecture for better synthesis quality and fair comparison.
MixDL For MixDL, we alternate between the normal adversarial training step and the interpolation/regularization step. In the former we go through normal image-level discrimination and in the latter, we apply patch-level discrimination on the mixup samples and compute losses for MixDL-G and MixDL-D. For patch discrimination, we largely adopt the implementation of Cross-domain Correspondence (CDC)[4]. Our linear projection layer for the discriminator operates on 512 dimension.
Percpetual Path Length For PPL computation, we mainly follow the implementation in StyleGAN2. The difference is that we subdivide a latent interpolation path into 10 subintervals and compute the perceptual distance for each

[1] https://github.com/rosinality/stylegan2-pytorch
[2] https://github.com/mit-han-lab/data-efficient-gans
[3] https://github.com/odegeasslbc/FastGAN-pytorch
[4] https://github.com/utkarshojha/few-shot-gan-adaptation

line segment. Since the original PPL computation divides the perceptual distance by the squared step size, we divide each subinterval length by 0.1^2. For clear demonstration, we divide the endpoint mean by 0.1^2 as well. Note that the overall procedure is equivalent to calculating LPIPS multiplied by the factor of 100. The standard deviation is computed across the subintervals, and averaged for the interpolation paths.

Number of Modes We generate 500 samples and compute their perceptual distances to the 10 training samples. We record the index for the real sample with the smallest perceptual distance and report the unique count. It is visually apparent from Fig. S1 that our method boosts mode diversity.

B Datasets

We present the datasets used in our work along with their size.

Animal-Face Dog	Oxford Flowers	FFHQ Babies	Sketches	Obama	Grumpy Cat
10	10, 100, 1000, 8192	10, 100, 1000, 2479	5, 10	10, 100	10, 100
Pokemon	Amedeo Modigliani	Anime Face	Landscape	Totoro	
10	10	10	10	5	

Table S2: Number of shots used in each dataset. **Names of datasets** are presented in the first and third rows and their corresponding **number of shots** used in this paper are described in the second and fourth rows.

C Additional Evaluations with CDC [34]

We provide evaluation results for CDC [34] on two popular low shot benchmarks, Obama and Cat (Tab. S3). To simulate few-shot setting, we randomly sample 10 images from each dataset.. Since CDC is pretrained on FFHQ, the domain gap is relatively small, especially for Obama dataset. Nevertheless, we observe superior performances with MixDL.

	Obama (10-shot)				Cat (10-shot)			
Model	FID(↓)	LPIPS(↑)	Prec.(↑)	Rec.(↑)	FID(↓)	LPIPS(↑)	Prec.(↑)	Rec.(↑)
CDC	75.0	0.490	0.47	0.07	45.3	0.451	0.52	0.10
MixDL	**62.7**	**0.601**	**0.53**	**0.09**	**41.1**	**0.590**	**0.78**	**0.11**

Table S3: FID, precision and recall are computed against the full dataset (with 100 images) while LPIPS is computed among the generated samples.

D Additional Baseline Comparisons

We present quantitative evaluation results with concurrent competitive baselines [43, 11] in combination to different data augmentations in Tab. S4. We observe consistent benefits from MixDL.

Dataset	Anime-face		Dog		Flower		Baby	
Metric	FID	LPIPS	FID	LPIPS	FID	LPIPS	FID	LPIPS
LeCam + DA	286.7	0.130	129.7	0.593	189.2	0.688	127.7	0.588
GenCo + DA	222.4	0.082	147.2	0.565	186.1	0.702	119.3	0.605
MixDL + DA	**70.2**	**0.551**	**96.4**	**0.682**	**129.9**	**0.705**	-	-
LeCam + ADA	111.6	0.405	239.0	0.378	191.0	0.659	178.3	0.451
GenCo + ADA	93.7	0.450	112.4	0.652	194.0	0.673	103.8	0.570
MixDL + ADA	**75.0**	**0.571**	**94.1**	**0.684**	**127.7**	**0.763**	-	-
MixDL (no aug.)	73.1	0.548	96.0	0.682	136.6	0.734	**83.4**	**0.643**

Table S4: Comparison with additional baselines. MixDL consistently outperforms others even without advanced augmentations.

E Training Snapshots

We provide training snapshots for FastGAN and StyleGAN2 for visual demonstration of diversity and interpolation smoothness. Fig. S1 clearly shows that as opposed to vanilla FastGAN that rapidly loses diversity and converges to few prototypes, MixDL successfully alleviates this. Fig. S2 displays interpolation snapshots for StyleGAN2. In early training iterations, it does show relatively smooth latent transition, but the sample quality is very unsatisfactory. As the training proceeds, the sample quality improves as the model *overfits*, but consequently the interpolation smoothness is quickly lost. This describes the classic dilemma in few-shot generative modeling. In contrast, Fig. S3 shows that as MixDL is effective at maintaining latent space smoothness, it provides a sweet spot where reasonable sample quality and smooth latent transition coexist. Note that models with MixDL do inevitably overfit in the end, but we can find reasonable stopping point that produces diverse unseen samples with satisfactory visual quality.

F Additional Generated Samples

We present latent interpolation results in Fig. S4 and Fig. S5. Fig. S4 shows that MixDL yields smoother latent interpolation compared to baseline methods that show typical *stairlike latent space*. Fig. S5 reaffirms this observation on various datasets. We note that images of Japanese animation character Totoro were crawled from the web, and 5 real samples were used. Additional synthesis results from face paintings of Amedeo Modigliani and illustrations of Totoro are displayed in Fig. S6 and Fig. S7, respectively.

G Sample Images from Low-shot Benchmarks

In Fig. S8, we present samples from Obama and Grumpy Cat datasets. As they contain images of a single character, the intra-diversity is inherently very limited, which is also demonstrated by the LPIPS measure in Tab. 3 of the main paper.

H Naive Application of GAN adaptation

We display results from naive application of CDC. Since it is very difficult to find a semantically similar source domain for datasets like Pokemon, we naively leverage the source generator trained on FFHQ. As the source and the target are semantically different, the adaptation does not yield satisfactory outcomes as expected. We can observe the dilemma here as well that in the early iterations, the face shape learned in the source domain is clearly visible while in later stages, the face shape is no longer visible but the model collapses altogether. As CDC preserves distances in the target domain through the correspondence to the source domain, it is not applicable to domains that lack an adequate source dataset to transfer from. MixDL, on the other hand, improves upon CDC in that it enables training generative models with minimal overfitting and mode collapse, without leveraging source domain pretraining. Quantitative evaluations further support the claim as in Tab. 1 of the main paper.

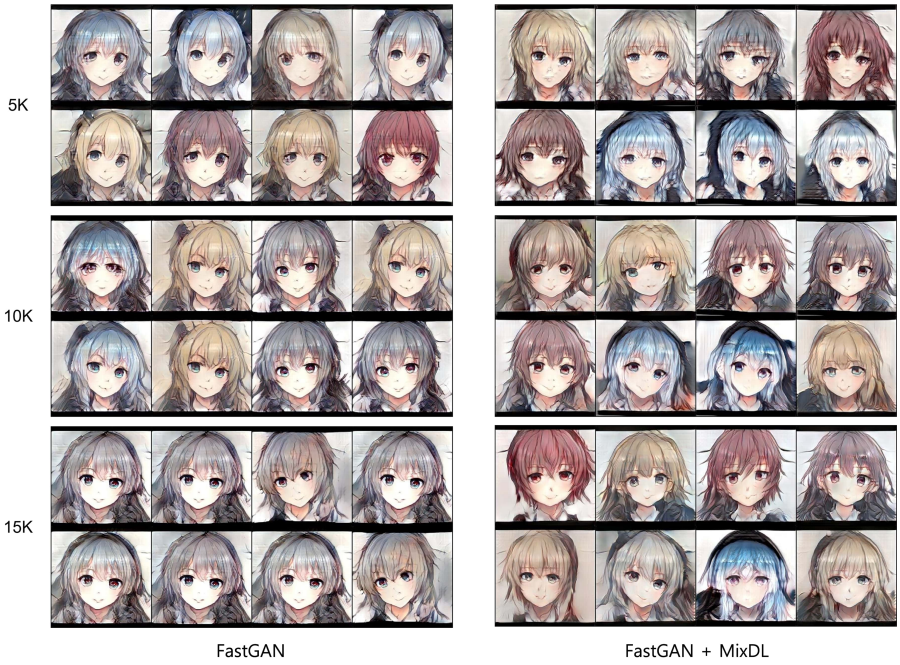

Fig. S1: Training snapshots for FastGAN and FastGAN+MixDL in early iterations. As opposed to the base FastGAN that rapidly loses diversity, our regularizations help preserve the modes throughout the course of training. Numbers in the left indicate training iterations.

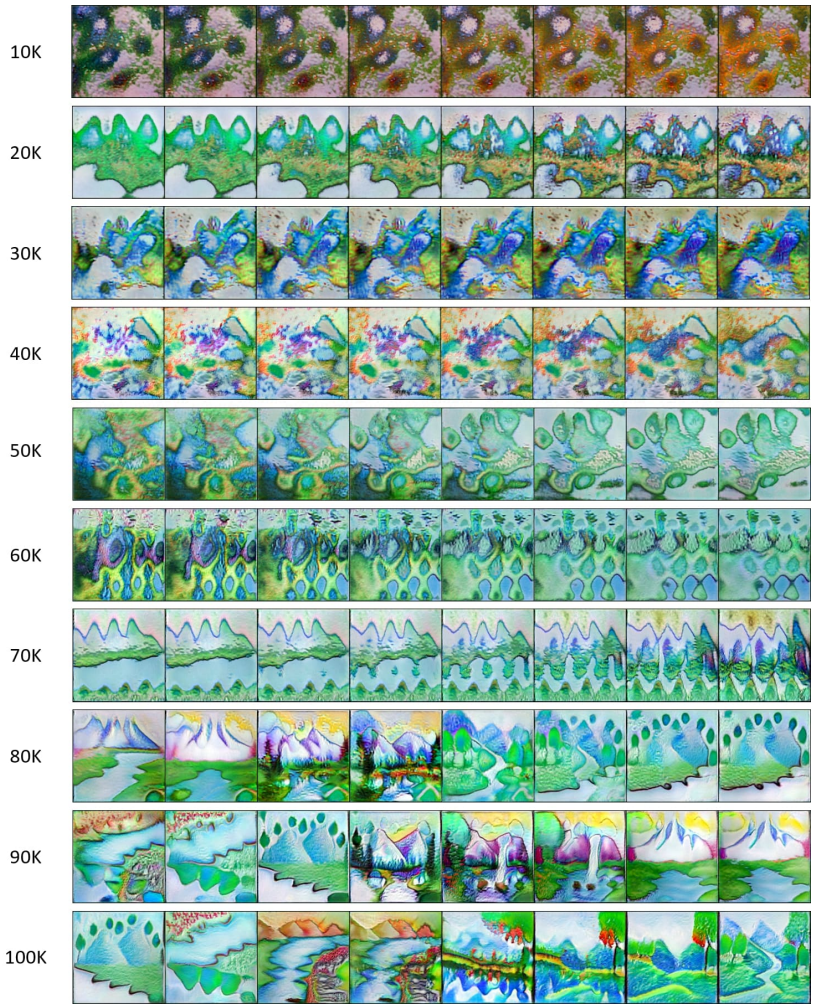

Fig. S2: Interpolation snapshots for StyleGAN2. Numbers in the left indicate training iterations.

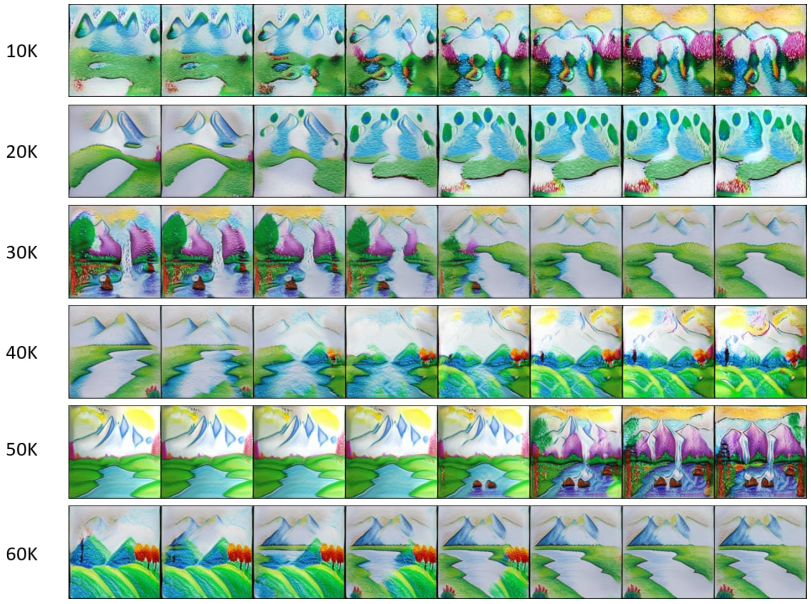

Fig. S3: Interpolation snapshots for StyleGAN2+MixDL.

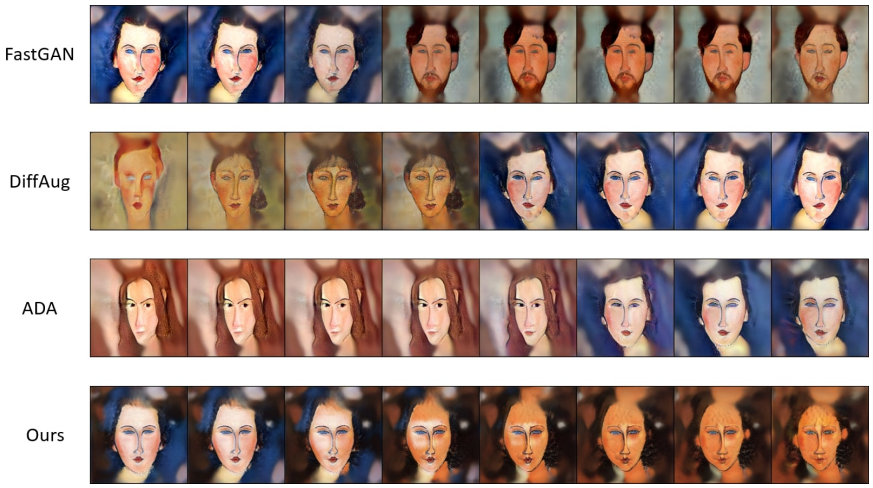

Fig. S4: Interpolation examples. Baselines clearly display *stairlike* latent transition while ours shows smooth interpolation.

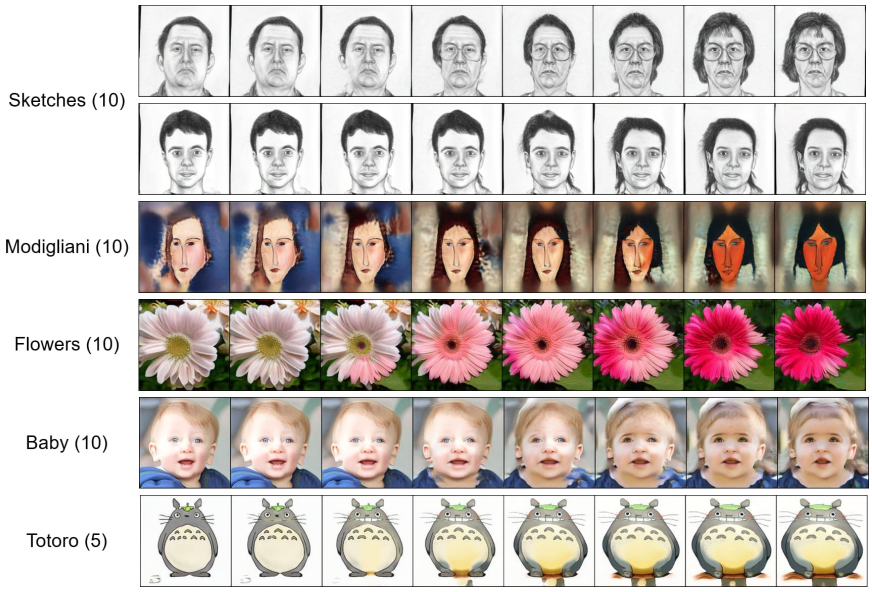

Fig. S5: More interpolation examples from MixDL. Numbers in the parentheses represent the number of training samples used for each dataset.

Fig. S6: Samples from face paintings of Amedeo Modigliani. While the baselines simply replicate the given images, ours produces diverse unseen face images. *Ours* represents samples from StyleGAN2+MixDL.

Fig. S7: MixDL generation result from 5-shot training on Totoro. Although there are only 5 training samples, it combines visual features in a natural way to produce diverse novel samples.

Fig. S8: Random samples from low-shot benchmark datasets, Obama and Grumpy Cat. Since they contain photos of a single character, the intra-diversity is inherently constrained, rendering these benchmarks inappropriate to evaluate generative diversity.

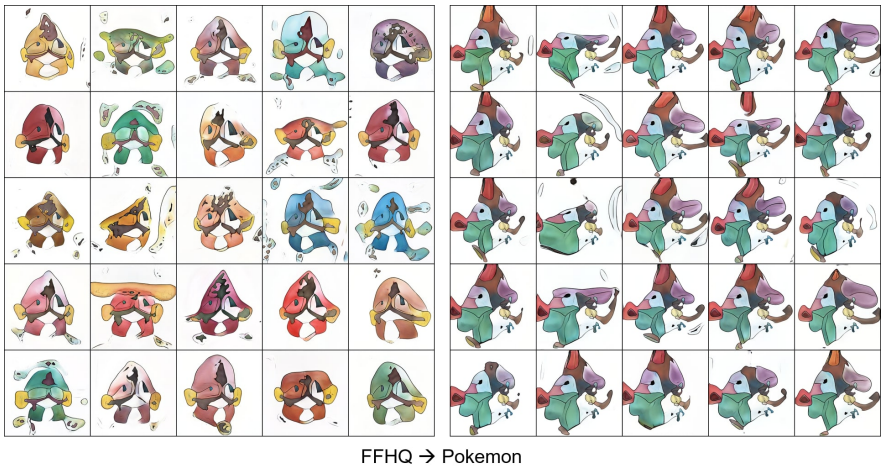

Fig. S9: Naive application of CDC from FFHQ to Pokemon. As the authors have pointed out, the adaptation performance degrades when the two domains are semantically different, but it is not straightforward to find a transferable source domain for datasets like Pokemon. We observe clear human face shapes in the early stages *(left)* and mode collapse in later stages *(right)* where the face shape is no longer visible.