A. Implementation Details

In this section, we will provide the full details for our experiments in Sec. 5 for reproducibility.

A.1. Training and Sampling from Deep Generative Models for Different Datasets

A.1.1 CIFAR-10

We used the checkpoints provided in the official repositories, where LSGM[6] was reported with FID 1.94 and StyleGAN-XL[7] with FID 1.85. LSGM only supports unconditional training on CIFAR-10. Therefore, we additionally applied a classifier trained on CIFAR-10 for labeling and rejection sampling. A Wide Residual Network (WRN) was chosen for this classification task, where the structure *WRN-28-10* was selected. We used the checkpoint provided from an open repository[8], where an accuracy of 96.21% on CIFAR-10 was reported. For rejection sampling, we set the threshold to 0.8, meaning that we exclude all the samples with a prediction probability lower than 80%. The same classifier was applied to StyleGAN-XL as a filtering mechanism on top of the class-conditional sampling to allow for a fair comparison.

A.1.2 CUB-Bird and Oxford-Flower

Compared to CIFAR-10, CUB-Bird and Oxford-Flower are much smaller datasets according to their total size (4,521 and 1,010) and the average number of samples per class (22 and 10). This poses a challenging task for training deep generative models, especially for the diffusion-based kind. We trained four models (details below) on both datasets from scratch, following the instructions from the respective authors' official repositories. All models were trained to generate images at resolution 256×256.

LSGM. We followed the instructions for "CelebA-HQ-256 Quantitative Model" in the original repository and modified the command to train on one Tesla V100 GPU.

Fast-GAN. Following the instructions in the original repository[9], we trained the models on both datasets on one Tesla V100 GPU with batch size 12 for 100,000 iterations.

Projection-GAN. We trained the models on both datasets with one Tesla V100 GPU with batch size 8 for 20,000 kimgs (i.e., the model went through this number of images [27]). Note that for CUB-Bird, we used the configuration of *fastgan* while for Oxford-Flower we applied *fastgan-lite* according to author's suggestions based on dataset size.

[6]https://github.com/NVlabs/LSGM
[7]https://github.com/autonomousvision/stylegan-xl
[8]https://github.com/xinntao/pytorch-classification-1
[9]https://github.com/odegeasslbc/FastGAN-pytorch

Table 5. The training and sampling statistic of various deep generative models on the CUB-Bird and Oxford-Flower dataset. For training, we denote the converged model by "v", otherwise "x". For sampling, we report the number of classes that met the sampling requirements.

Dataset	Stage	LSGM	Fast-GAN	Proj-GAN	SG-XL
CUB	Training	x	v	v	v
	Sampling	0 / 200	173 / 200	199 / 200	194 / 200
Flower	Training	v	v	v	v
	Sampling	69 / 102	67 / 102	102 / 102	101 / 102

StyleGAN-XL. Following the instructions in the original repository, we first trained a model for each dataset at resolution 32 and then directly scaled it up to resolution 256 at the second stage of training. All the models were trained on one Tesla V100 GPU with batch size 8 for 10,000 kimgs.

For sampling, we deployed a transformer-based classifier structure—Big Transfer (BiT) for labeling and rejection sampling. We picked the architecture of *BiT-M-R50x1* as the backbone. The BiT classifiers for both datasets were trained from scratch and reached the accuracy of 82.52% and 98.26% on the validation set for CUB-Bird and Oxford-Flower, respectively. The threshold for rejection sampling for both datasets was set to 0.5.

To construct a synthetic version of both datasets, we aimed to sample 30 images for each class in CUB-Bird and 10 images for each class in the case of Oxford-Flower and let the rejection sampling process run for at most 10 days. We report the training and sampling statistics in Tab. 5. Among all the models, only *LSGM* CUB failed to converge. *Proj-GAN* Flower was the only model to reach the target amount of images we seek to have within the 10-day timeframe. As a result, we only selected the synthetic datasets generated from *Proj-GAN* and *SG-XL* for further experiments in the main paper. Also, we omitted six classes (i.e., *Least Auklet*, *Spotted Carbird*, *Northern Flicker*, *Slaty Backed Gull*, *Whip Poor Will* and *Bohemian Waxwing*) in CUB-Bird and one class (i.e., Tiger Lily) in Oxford-Flower, forming a subset of the original dataset—194 classes and 101 classes for CUB-Bird and Oxford-Flower, respectively.

A.1.3 SDI

As for the SDI dataset, we thank the authors of DT-GAN [56] for providing us the real dataset and the synthetic dataset, where the statistics of the real dataset can be found in the supplementary of their paper and the synthetic dataset contains an 8,000-image subset for each product.

A.1.4 Statistic of the Sampled Synthetic Datasets

We report the statistics of the sampled sythetic datasets in Tab. 6. Interestingly, we note that FID can serve as a rough

Table 6. The quantitative results of the sampled synthetic datasets from commonly used evaluation metrics: FID, Precision/Recall [30], and Diversity/Coverage [35]. Note that the reported FIDs were computed between the original dataset and the sampled synthetic dataset, therefore might differ from the reported FID of the checkpoints used for sampling.

Dataset	Method	FID	Precision	Recall	Diversity	Coverage
CIFAR-10	[54]	16.55	0.66	0.65	0.70	0.74
	[48]	32.97	0.75	0.34	1.01	0.64
CUB	[47]	7.64	0.80	0.59	1.16	0.92
	[48]	10.05	0.88	0.39	1.52	0.92
Flower	[47]	26.91	0.83	0.69	0.97	0.93
	[48]	29.83	0.90	0.48	1.18	0.91
SDI-A	[56]	131.24	0.04	0.01	0.01	0.01
SDI-B		140.39	0.30	0.01	0.14	0.05
SDI-C		116.16	0.01	0.01	0.01	0.01

indicator for the effectiveness of the synthetic datasets, but how to predict the precise effect of these images in downstream tasks is yet to discover. Additionally, the scores of recall, which measures the fraction of the training data manifold being covered by the generator, are the most indicative among the other three metrics. We interpret it as a support sign for our claim—that the *Content Gap* (i.e., mode coverage) exhibited in the synthetic datasets is a main factor for the performance drop in downstream tasks.

A.2. Hyperparameters for Pretrained Guidance and Real Guidance

We report the selected hyperparameters for **Zero-shot** classification in Tab. 7. For Tab. 3 in the main paper, we used $\lambda_3 = 10$ for L1 distance and $\lambda_3 = 1,000$ in the case of KL-divergence. We observed that stronger regularization is in general needed when using random initialization. For **Low-shot** classification, we report the used hyperparameters in Tab. 8. Note that when the number of available real images is much lower than the number of synthetic images, we empirically found that not updating the model with the gradients from real data (e.g., $\lambda_1 = 0$ for CIFAR-10) led to better performance.

B. Extended Investigation Results

In this section, we present additional results of the empirical investigations in Sec. 3.

B.1. Results from ImageNet Initialized Classifiers

We present the achieved accuracy of the classifiers on their respective training sets as well as on the real CIFAR-10 validation and test set in Tab. 9. Additionally, we sorted the *Synthetic* images based on their sample losses in **Observation #3** and divided them into two subsets of equal amounts. Then, we combined each subset with the *Real* images to form two augmented new training sets and trained new classifiers on top. The results shown in Tab. 10 endorse the legality of judging samples by their loss, as it can be seen that the subsets with larger losses boost the performance more than the smaller halves. Note that despite having small losses, the subsets of such images still have a positive impact on the performance. We hypothesize that this is because the synthetic images from deep generative models (DGMs) do add variations in the dense areas of the training distribution despite lacking rare samples.

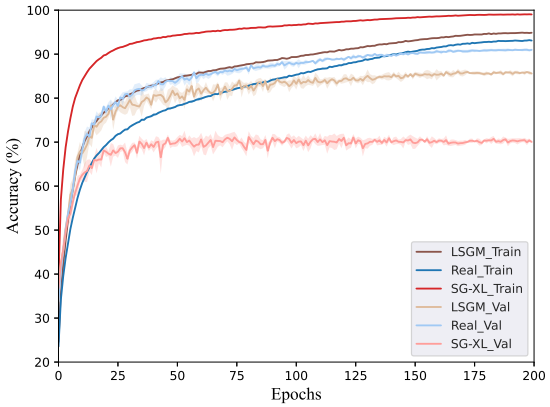


Figure 5. The training and validation curves of CIFAR-10 images from different sources over five runs (Random Initialization). Standard deviations are plotted as shaded area. (Zoom in.)

B.2. Results from Randomly Initialized Classifiers

For the setting where all classifiers were randomly initialized, the learning rate was set to 0.01 while all other hyperparameters remained the same as in the main paper. We report the achieved accuracy of the classifiers on their respected training sets as well as on the real CIFAR-10 validation and test set in Tab. 11. It can be seen in Tab. 12 that the non-mutual performance gap (**Observation #1** in the main paper) is even more pronounced with random initialization. Also, the saturation effect on training and validation curves as mentioned in **Observation #2** can be clearly observed in Fig. 5. Last but not least, the resulting loss distributions plots in this setting (cf. Fig. 6) show the same tend as discovered in **Observation #3** and the classifier performance in Tab. 13 consists with our finding in Tab. 10. We therefore conclude that our insights in Sec. 3 are non-negligible and independent from the initialization method.

B.3. Visualization of High-loss and Low-loss Images

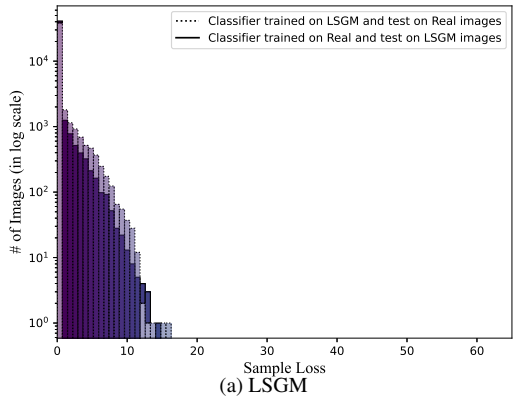
We show the high-loss and low-loss *Synthetic* images in Fig. 9 and Fig. 10, and the *Real* images evaluated by classifiers trained on *Synthetic* images in Fig. 11 and Fig. 12. It can be observed that the quality of the *Synthetic* images from DGMs is on par with the *Real* images from the original

Table 7. The intensity of Pretrained Guidance (λ_3) used in Table 2. (A) The networks were train from scratch. (B) The networks were initialized by pretrained ImageNet weights.

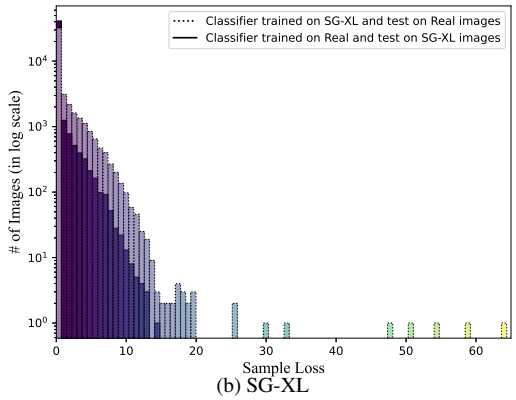
	Dataset	CIFAR-10		CUB		Flower		SDI-A	SDI-B	SDI-C
	Source	LSGM	SG-XL	Proj-GAN	SG-XL	Proj-GAN	SG-XL	DT-GAN		
(A)	**Ours(L1)**	1	1	1	5	1	5	1	5	5
	Ours(KL)	1,000	1,000	600	600	600	600	75	1	1
(B)	**Ours (L1)**	1	1	1	1	1	1	1	5	10
	Ours (KL)	1,000	1,000	45	50	45	1	1	1,000	1

Table 8. The intensity of Real Guidance (λ_1 and λ_2) and Pretrained Guidance (λ_2) used in Table 4 (ImageNet Initialization). We report the selected λ_1, λ_2 and λ_3 in each entry.

Dataset	CIFAR-10 (10-shots)		CUB		Flower		SDI-A	SDI-B	SDI-C
Syn-to-Real Ratio	450:1		1:1		1:1		2:1	2:1	2:1
Source	LSGM	SG-XL	Proj-GAN	SG-XL	Proj-GAN	SG-XL	DT-GAN		
λ_1	0	0	1	1	1	1	1	1	1
λ_2	1	1	2	2	2	2	1	1	1
λ_3	100	100	50	50	50	50	1.75	1.75	0.1



(a) LSGM



(b) SG-XL

Figure 6. The loss distribution of the samples from different sources. We overlapped the evaluation of *Real* $\rightarrow$ *Synthetic* (Solid) and *Synthetic* $\rightarrow$ *Real* (Dotted) in each subgraph. Note that all models used for evaluation were initialized with random weights.

Table 9. The achieved accuracy of classifiers (ImageNet Initialization) trained and tested on different sources of images. Note that the validation and test sets containing only real CIFAR-10 images. The training sets are the one used to train the respective classifier.

Source of Test Images	Train	Val	Test
Classifier Trained on Real	96.34	92.34	91.80
Classifier Trained on LSGM Images	97.71	86.10	85.57
Classifier Trained on SG-XL Images	99.82	79.43	78.94

Table 10. The achieved accuracy of classifiers trained on the datasets augmented by different subsets of synthetic images. We note the size of the training sets in brackets.

	LSGM	SG-XL
Real+Small Losses (67.5k)	91.99 (+0.19)	91.91 (+0.11)
Real+Large Losses (67.5k)	92.50 (+0.70)	91.97 (+0.17)
Real Only (45k)	91.80	

Table 11. The achieved accuracy of classifiers (Random Initialization) trained and tested on different sources of images. Note that the validation and test sets containing only real CIFAR-10 images. The training sets are the one used to train the respective classifier.

Source of Test Images	Train	Val	Test
Classifier Trained on Real	99.85	91.13	90.27
Classifier Trained on LSGM Images	99.83	86.58	85.08
Classifier Trained on SG-XL Images	99.96	72.96	72.22

CIFAR-10 training set. However, compared to the high-loss *Synthetic* samples, the high-loss *Real* samples (i.e., the hard cases in view of classifiers trained on *Synthetic* data) resemble rarer but plausible attributes (e.g., the dog with red hat in Fig. 12). We interpret this as a support sign of our claim in the main paper—that the *Synthetic* dataset is less diverse and simpler than its original training set due to the absence of rare samples.

Table 12. The achieved accuracy of classifiers (Random Initialization) trained and tested on different sources of images. Note that the results were all acquired from the training sets, therefore the high accuracy in the diagonal line indicates that the classifiers have converged on their own training set.

Source of Test Images	Real	LSGM	SG-XL
Classifier Trained on Real	99.85	90.09	97.25
Classifier Trained on LSGM Images	85.82	99.83	98.18
Classifier Trained on SG-XL Images	72.32	79.43	99.86

Table 13. The achieved accuracy of classifier (Random Initialization) trained on the datasets augmented by different subsets of synthetic images. We note the size of the training sets in brackets.

	LSGM	SG-XL
Real+Small Losses (67.5k)	91.54 (+1.27)	90.97 (+0.70)
Real+Large Losses (67.5k)	92.50 (+2.23)	91.38 (+1.11)
Real Only (45k)	90.27	

Table 14. The quantitative results of the sampled synthetic datasets from commonly used evaluation metrics: FID, Precision/Recall [30], and Diversity/Coverage [35]. Note that the reported FIDs were computed between the original dataset and the sampled synthetic dataset, therefore might differ from the reported FID of the checkpoints used for sampling.

Dataset	**Method**	**FID**	**Precision**	**Recall**	**Diversity**	**Coverage**
ImageNet-10%	[11]	4.91	0.84	0.61	1.21	0.94
	[48]	5.28	0.79	0.58	1.06	0.90

B.4. Study on ImageNet

To demonstrate that the effects we observed in Sec. 3 are not confined to CIFAR-10, we also conducted the same investigation on a subset of ImageNet [10]—ImageNet-10%, where 128 samples of each class were randomly selected to form the subset. This resulted in a training set of 128,000 samples and we reported the statistics of the sampled synthetic datasets in Tab. 14. For evaluation, we split the official ImageNet validation set into a validation set of size 12,000 and a test set of size 38,000.

Experiment setup. We selected two popular DGMs—ADM [11] and SG-XL [48]—based on their promising performance on ImageNet and sampled directly from the provided checkpoints by the authors. We used the ImageNet class index as the condition to sample 128 images for each class at resolution 256×256 from both DGM.

Same as in the main paper, we chose ResNet-50 [18] as the backbone for the classifiers and set the image resolution to 224×224, following the common preprocessing procedure for ImageNet. The batch size was set to 256 and four NIVIDA Tesla V100 were used. The initial learning rate was set to 0.1 and a cosine annealing schedule was applied to tune the learning rate during the training. Only random crop and random flip were used as data augmentation. Note that we randomly initialized the network weights to observe the full effect of synthetic data. Initializing with ImageNet pretrained weights as in the main paper would eliminate the need for classifier training since the target dataset is also ImageNet. All the classifiers were trained for 300 epochs and the reported results in the following were acquired from averaging over three random runs.

Table 15. The achieved accuracy of classifiers trained and tested on different sources of images. Note that the validation and test sets are from the official ImageNet validation set, containing only real images. The training sets are the one used to train the respective classifier.

Source of Test Images	Train	Val	Test
Classifier Trained on Real	99.78	57.67	57.67
Classifier Trained on ADM Images	99.69	44.67	43.79
Classifier Trained on SG-XL Images	99.99	31.94	31.55

Table 16. The achieved accuracy of classifiers trained and tested on different sources of images. Note that the results were all acquired from the training sets, therefore the high accuracy in the diagonal line indicates that the classifiers have converged on their own training set.

Source of Test Images	Real	ADM	SG-XL
Classifier Trained on Real	99.78	71.11	71.11
Classifier Trained on ADM Images	47.04	99.69	58.60
Classifier Trained on SG-XL Images	33.28	46.77	99.99

Observations. We present the achieved accuracy of the classifiers on their respective training sets as well as on the real ImageNet validation and test set in Tab. 15. Together with Fig. 7, a sign of underfitting can be observed even on the classifier trained with real data, presumably due to the reduced size of the training set. However, we believe the experiments can still serve as a proxy for the behavior of the full dataset. As shown in Tab. 16, the non-mutual performance gap we claimed in Sec. 3 (**Observation #1**) can be clearly observed. Note that despite the notable drop when applying the classifier trained on *Real* to the *Synthetic* samples, the achieved accuracy is still significantly higher than on the test set of ImageNet ($71.11\% \rightarrow 57.67\%$). Also, it can be seen in Fig. 7 that the training curves show the same trend as in Fig. 2, fitting our claim in **Observation #2** that the training accuracy saturates quickly when training on *Synthetic* sources. Finally, we plotted the loss distribution in Fig. 8 and visualized the low-loss and high-loss samples in Fig. 13-16, where the same conclusion as in **Observation #3** can be drawn—that the datasets formed by *Synthetic* samples contain less information compared to the real one.

We interpret all of these observed effects as support signs to our claim that the synthetic datasets from current DGMs are the simplified version of the original dataset, where we assume the reason to be that the rare samples are either lost

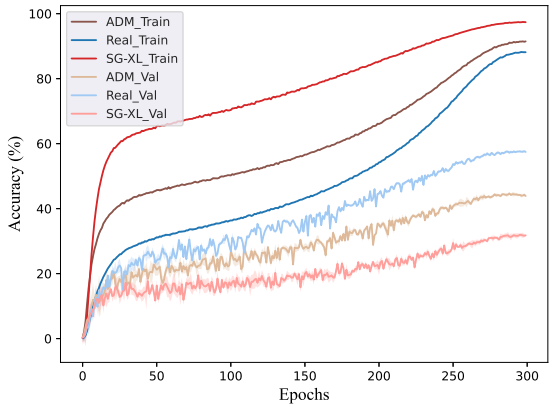


Figure 7. The training and validation curves of ImageNet-10% images from different sources over three runs. Standard deviations are plotted as shaded areas. (Zoom in.)

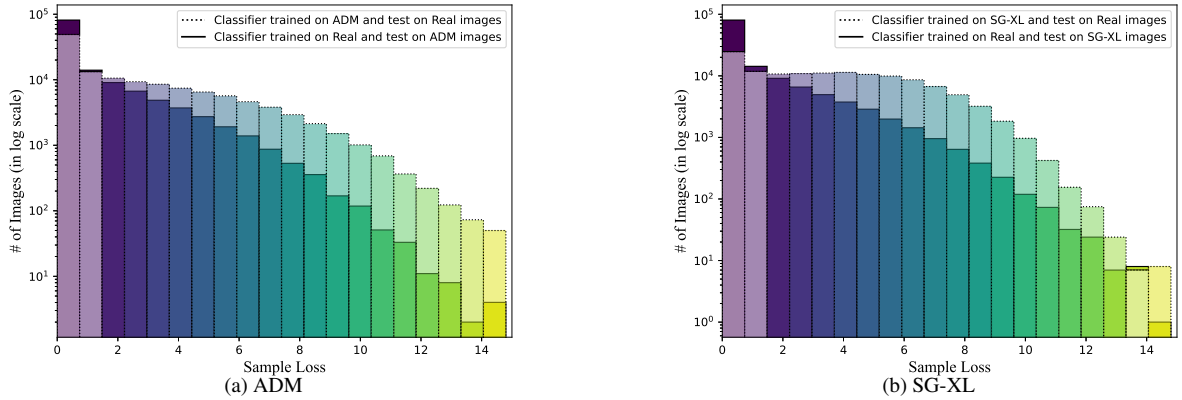
or under-represented in the sampled sets (i.e., the *Content Gap*).

C. Extended Experimental Results

In this section, we present additional experimental results as mentioned in Sec. 5.

C.1. Low-shot Image Classification with Random Initialization

We present additional results of low-shot image classification for random initialization in Tab. 17. From the results, we can draw the same conclusion as in the main paper: applying our proposed Pretrained Guidance and Real Guidance largely improved the classifier performance in most cases, especially in the low-data regimes. However, it is interesting to obverse that the dynamic between Pretrained Guidance and Real Guidance are different when starting the networks from random initialization. We believe this can be a direction of future investigations.

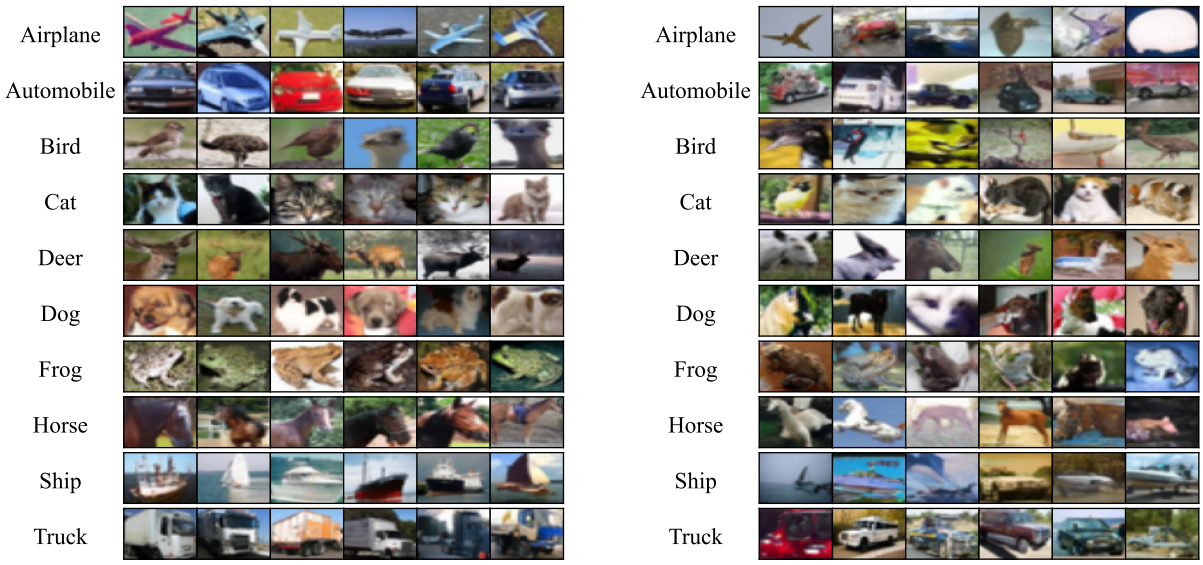


(a) ADM (b) SG-XL

Figure 8. The loss distribution of the samples from different sources. We overlapped the evaluation of *Real* $\rightarrow$ *Synthetic* (Solid) and *Synthetic* $\rightarrow$ *Real* (Dotted) in each subgraph. Note that all models used for evaluation were initialized with random weights.

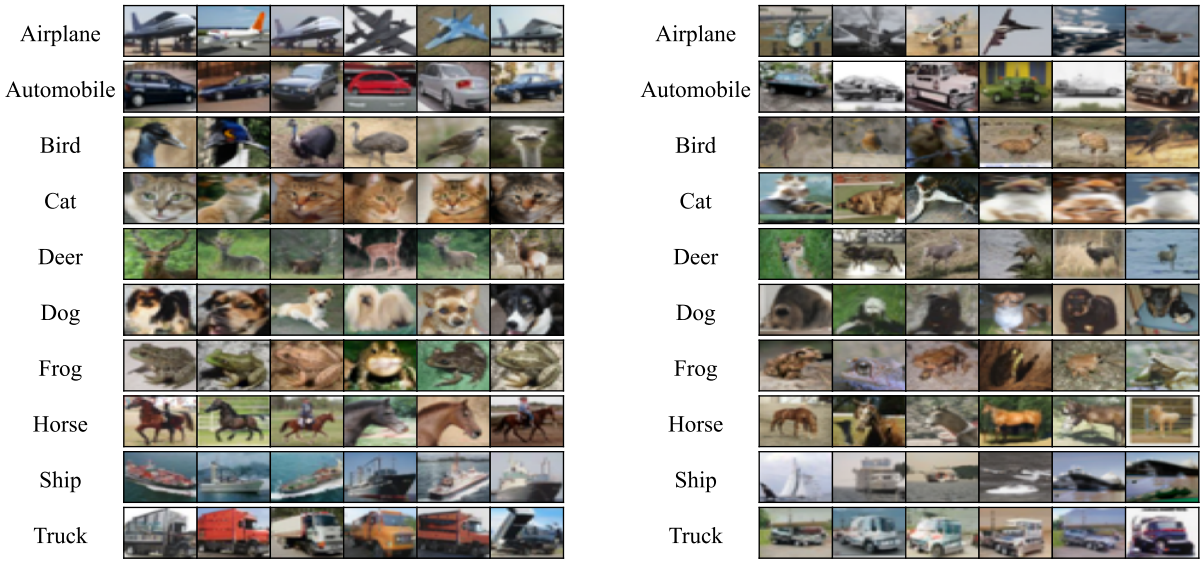
Table 17. The achieved accuracy of classifiers (Random Initialization) trained with different methods under various setting: **RG** indicate Real Guidance, **PG-F** means only apply Pretrained Guidance to the synthetic data, and **PG-R** means only apply Pretrained Guidance to the real data. Note that we denote the baseline two-stage domain adaption as **Adp.** and baseline data augmentation as **Aug.**.

Dataset				CIFAR-10 (10-shots)		CUB		Flower		SDI-A	SDI-B	SDI-C
Syn-to-Real Ratio				450:1		1:1		1:1		2:1	2:1	2:1
Source	**RG**	**PG-R**	**PG-F**	LSGM	SG-XL	Proj-GAN	SG-XL	Proj-GAN	SG-XL		DT-GAN	
Baseline (Real)	-	-	-	90.27		18.82		35.59		83.20	87.80	72.38
Baseline (Fake)	-	-	-	85.08	72.20	12.30	10.80	13.54	13.07	64.72	86.60	61.14
Baseline (Adp.)	-	-	-	84.95	71.93	21.86	21.76	20.11	19.25	73.45	85.60	64.76
Baseline (Aug.)	-	-	-	85.10	72.13	33.78	32.44	36.84	37.93	85.82	89.40	**87.05**
ADDA [53]	-	-	-	83.87	68.83	10.87	11.21	13.84	13.97	54.00	73.20	52.38
DADA [51]	-	-	-	83.01	71.83	12.80	12.09	13.72	14.16	53.45	81.00	51.24
DANN [14]	-	-	-	85.23	73.47	31.62	31.76	35.78	36.60	80.18	91.40	56.95
LTDA [25]	-	-	-	82.13	74.47	21.89	15.77	16.03	13.38	70.91	88.40	71.43
A-GEM [6]	-	-	-	**86.37**	72.79	12.72	12.66	14.76	14.73	74.55	88.40	71.62
	-	v	v	84.92	77.66	**41.59**	**39.49**	**42.52**	**41.27**	**89.27**	93.00	86.10
	v	-	-	**86.37**	72.79	33.04	31.22	36.64	36.47	88.18	**93.20**	77.71
Ours	v	v	-	85.59	78.70	36.27	33.46	35.94	36.29	78.91	91.80	73.90
	v	-	v	86.15	73.19	36.53	35.72	38.59	38.83	88.91	92.60	81.52
	v	v	v	86.00	**78.88**	39.60	37.95	40.14	38.77	86.18	92.60	76.38



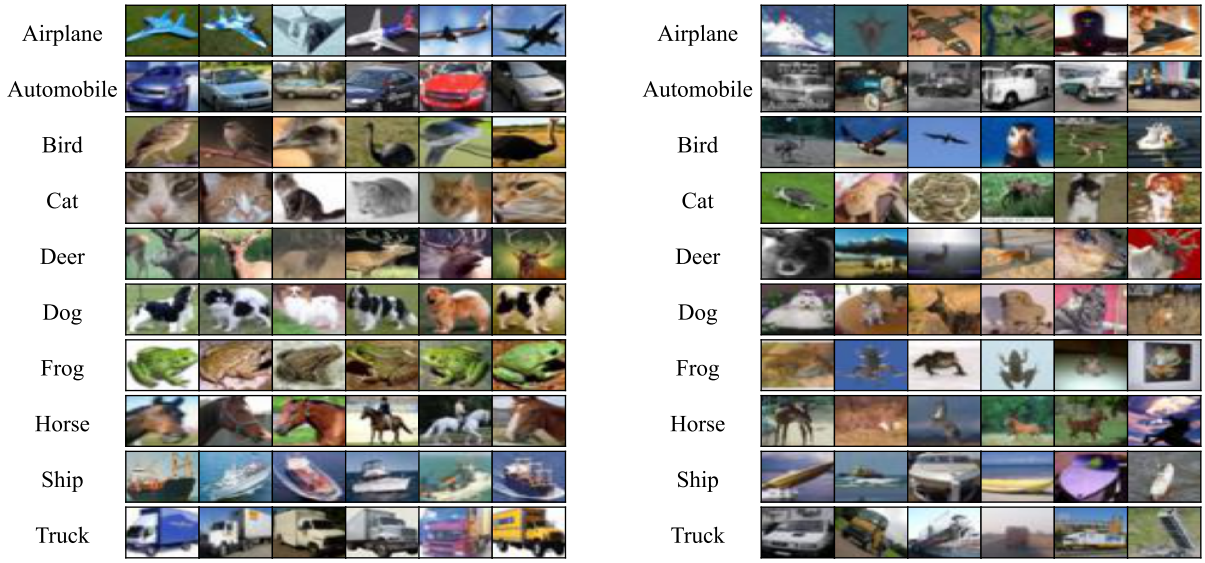
(a) Low-loss Samples (b) Hihg-loss Samples

Figure 9. The low-loss and high-loss LSGM synthetic samples. Note that the images are in ascending order according to their sample loss.



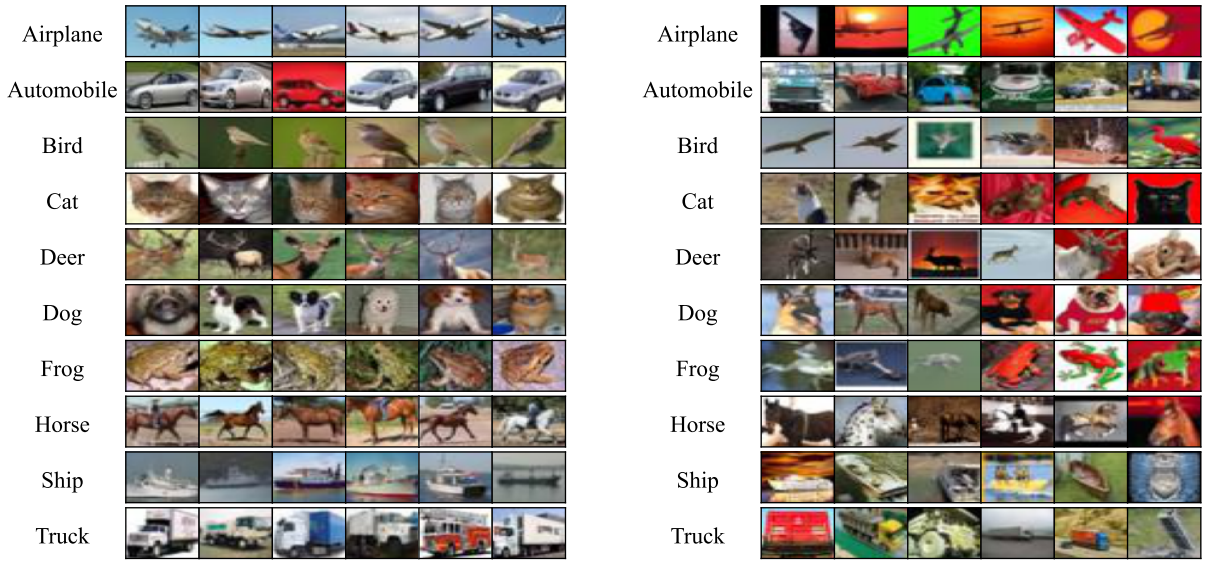
(a) Low-loss Samples (b) Hihg-loss Samples

Figure 10. The low-loss and high-loss StyleGAN-XL synthetic samples. Note that the images are in ascending order according to their sample loss.



(a) Low-loss Samples (b) Hihg-loss Samples

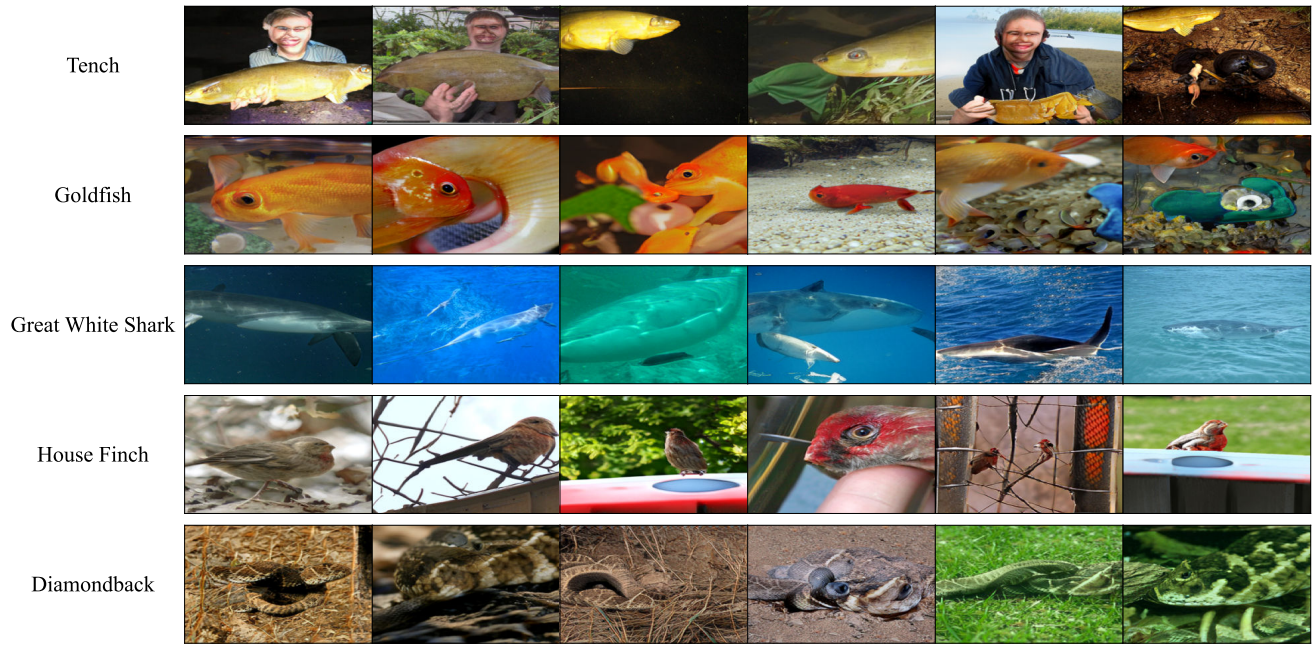
Figure 11. The low-loss and high-loss Real samples based on a classifier trained on LSGM synthetic samples. Note that the images are in ascending order according to their sample loss.



(a) Low-loss Samples (b) Hihg-loss Samples

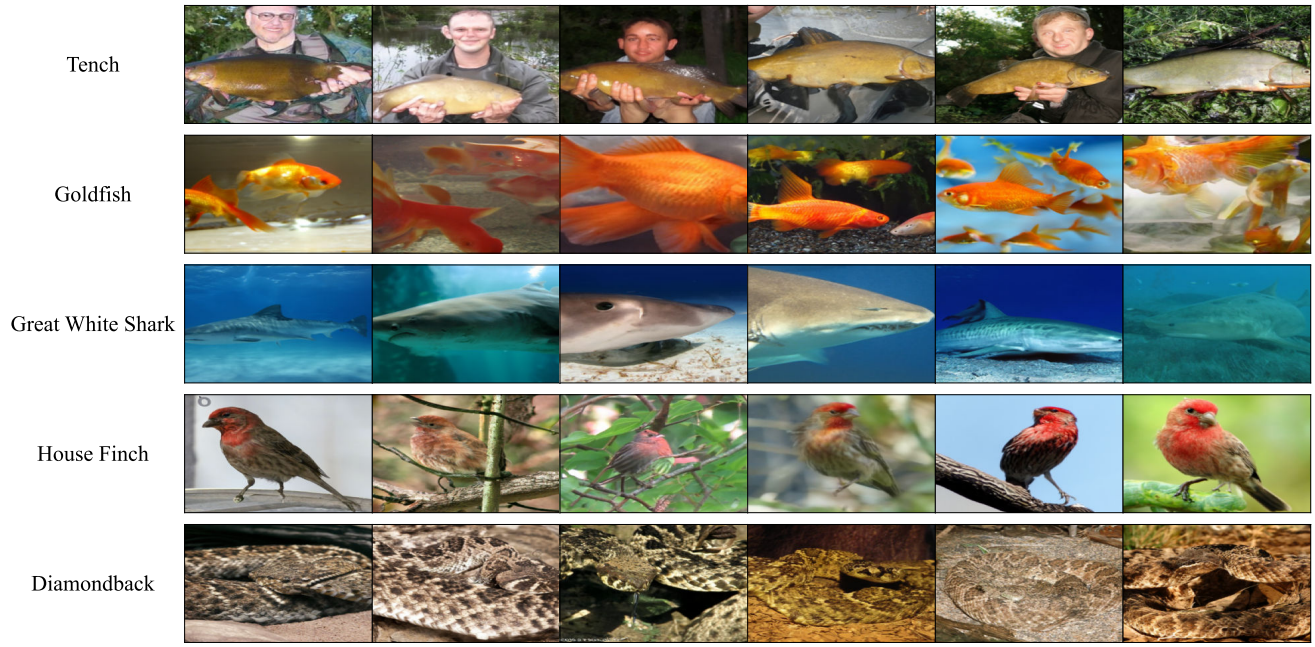
Figure 12. The low-loss and high-loss Real samples based on a classifier trained on StyleGAN-XL synthetic samples. Note that the images are in ascending order according to their sample loss.

(a) Low-loss Samples



(b) Hihg-loss Samples

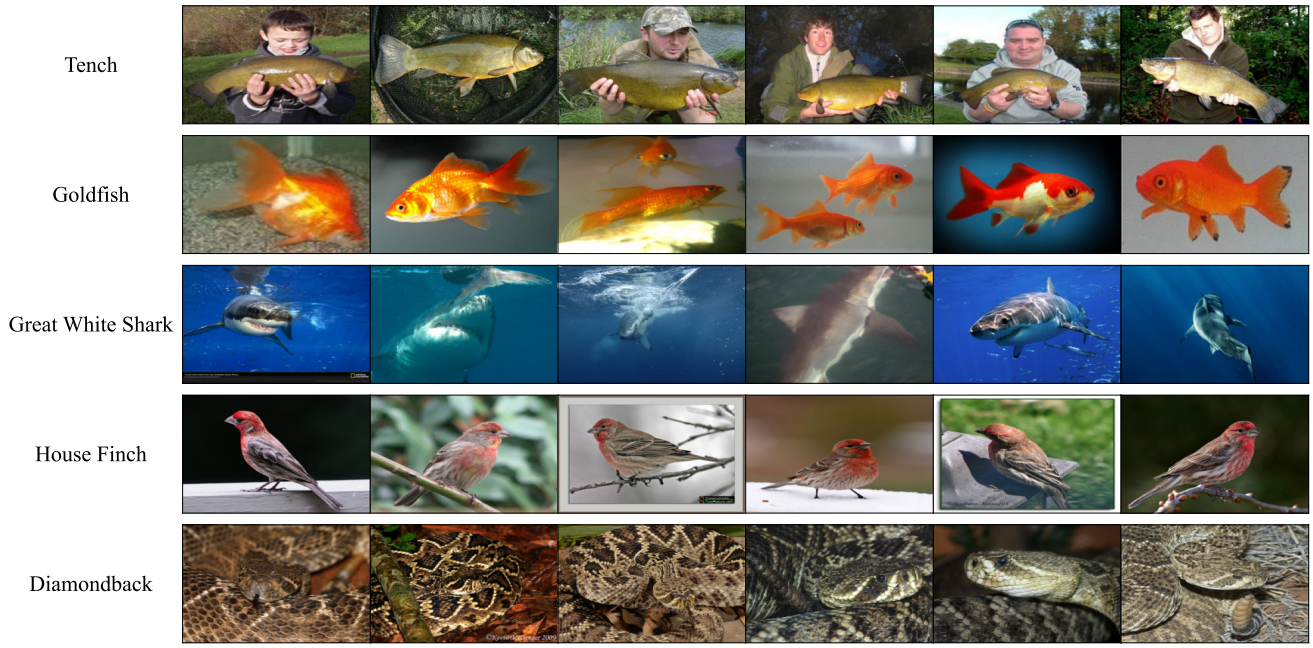
Figure 13. The low-loss and high-loss StyleGAN-XL synthetic samples. Note that the images are in ascending order according to their sample loss.



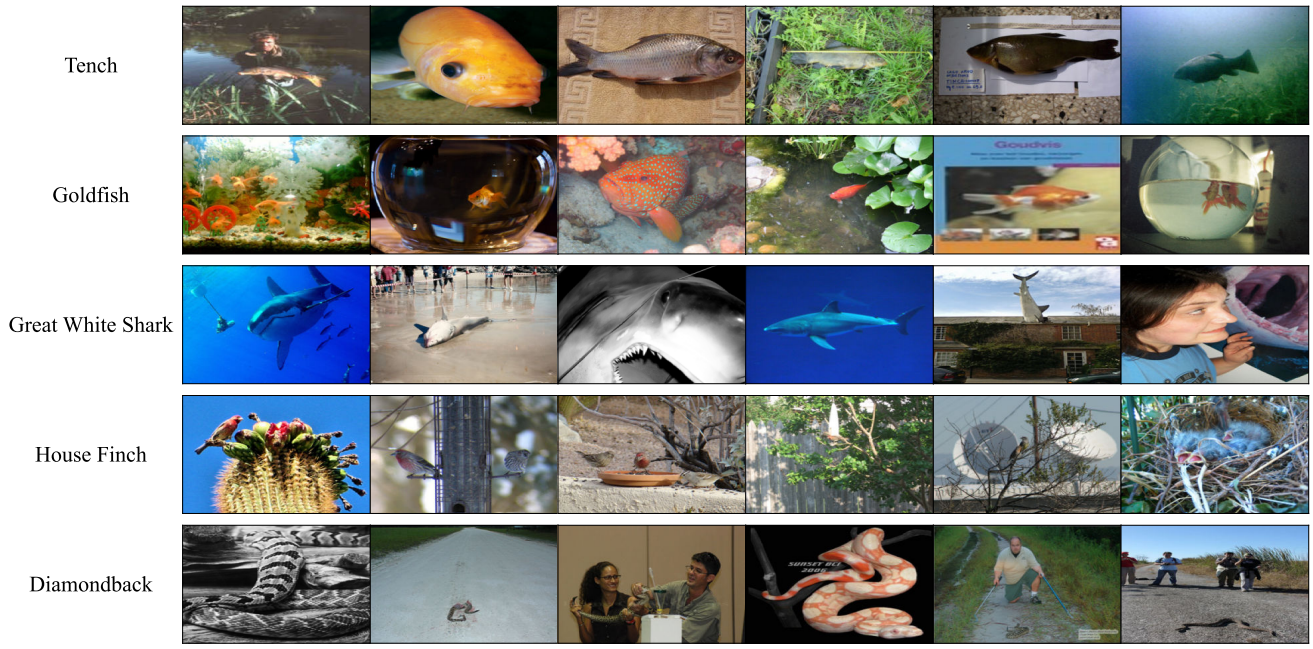
(a) Low-loss Samples

(b) Hihg-loss Samples

Figure 14. The low-loss and high-loss ADM synthetic samples. Note that the images are in ascending order according to their sample loss.

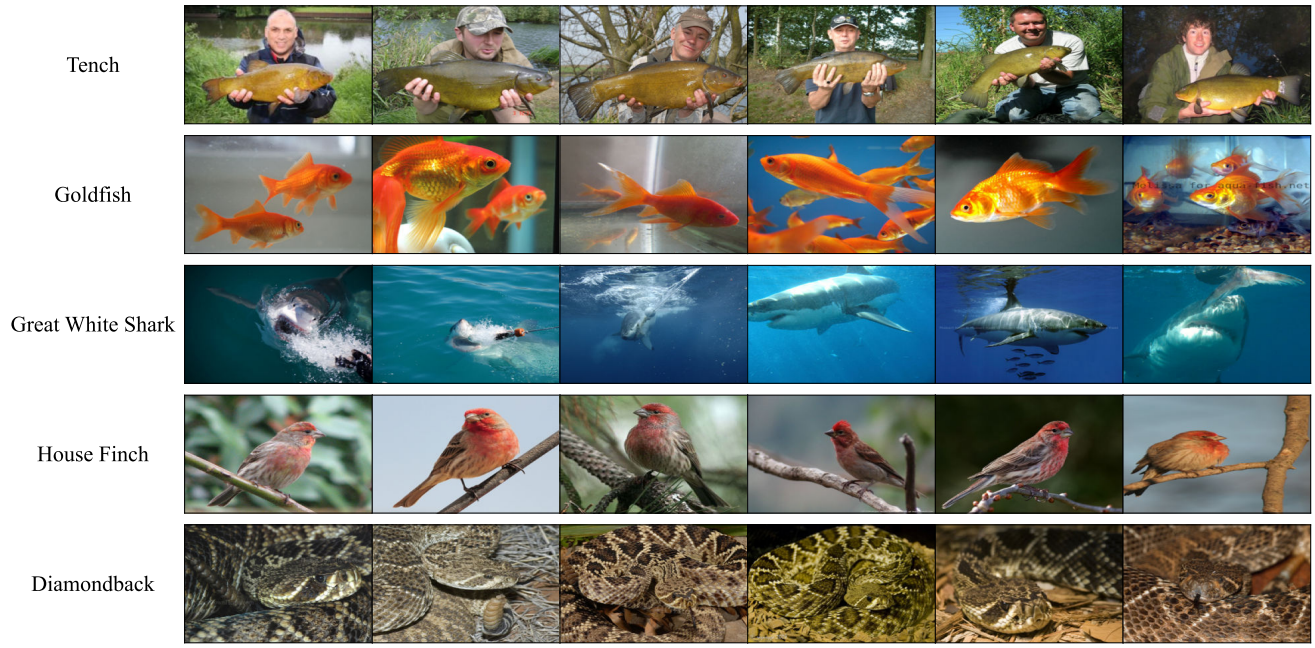


(a) Low-loss Samples

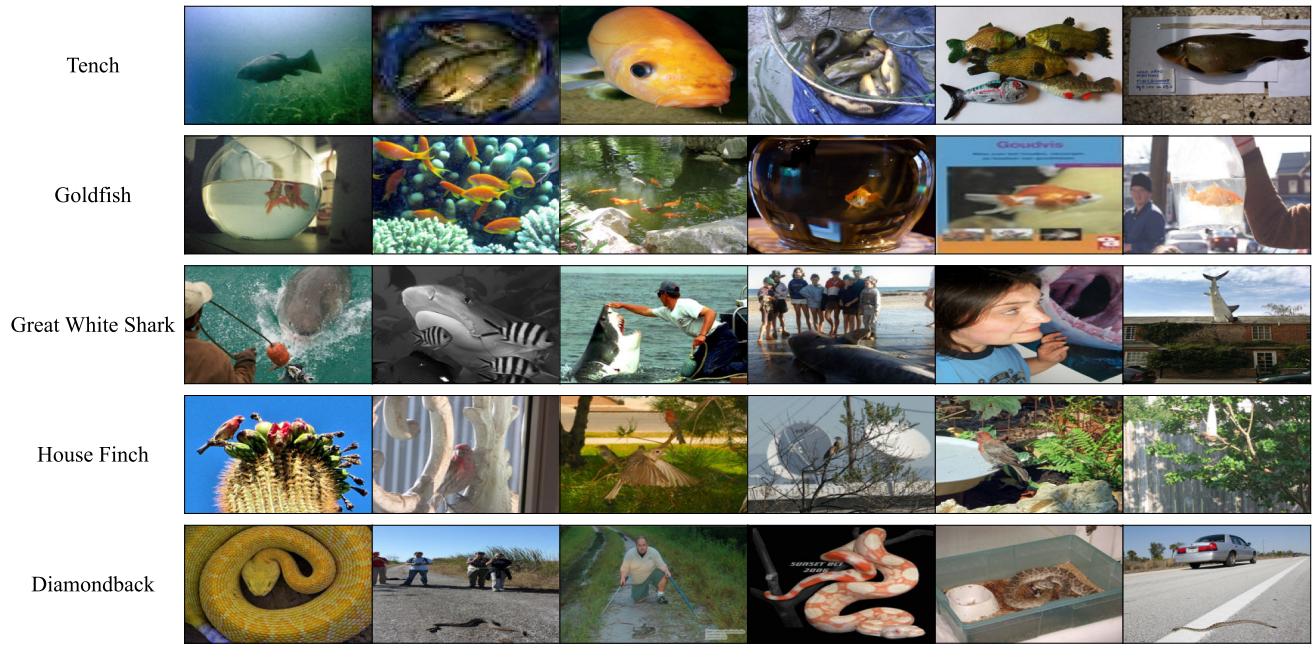


(b) Hihg-loss Samples

Figure 15. The low-loss and high-loss Real samples based on a classifier trained on StyleGAN-XL synthetic samples. Note that the images are in ascending order according to their sample loss.



(a) Low-loss Samples



(b) Hihg-loss Samples

Figure 16. The low-loss and high-loss Real samples based on a classifier trained on ADM synthetic samples. Note that the images are in ascending order according to their sample loss.