Efficient Dataset Distillation via Minimax Diffusion

Supplementary Material

The supplementary material is organized as follows: Section 6 provides more detailed theoretical analysis; Section 7 presents the related work to this paper; Section 8 elaborates upon the method pipeline; Section 9 contains additional implementation details; Section 10 presents some ablation studies; Section 11 discusses the broader impact; and finally, Section 12 presents ethical considerations.

6. Theoretical Analysis

We present the most relevant parts of the referred work [43, Section 2.1-2.2]. Consider that the diffusion takes place over the finite interval $[0, 1]$ and let μ be the desired sample distribution, such that $Z_1 \sim \mu$. Assume μ is absolutely continuous with respect to the standard Gaussian, denoted by γ_d, and define the Radon-Nikodym derivative $f = d\mu/d\gamma_d$. Then the optimal control, defined in the literature as the Föllmer drift and expressed as

$$
\begin{aligned}
&u^*(\mathbf{z}, t) = \nabla \log Q_{1-t}(f) \\
&= \nabla \log \tfrac{1}{(2\pi(1-t))^{d/2}} \int f(y) \exp\left(-\tfrac{1}{2(1-t)} \|\mathbf{z} - y\|^2\right) dy
\end{aligned}
$$

would be such that if $V(Z_t) = u^*(\mathbf{z}, t)$ in Eq. (8), then this drift would minimize the cost-to-go function:

$$
J^u(\mathbf{z}, t) := \mathbb{E}\left[\frac{1}{2}\int_t^1 \|u_s\|^2 ds - \log f(Z_1^u) | Z_t^u = \mathbf{z}\right].
$$

Equivalently, such a control is the one that, among all such transportation that maps from γ_d to μ, minimizes $\int_0^1 \|u_s\|^2 ds$ [14, 24].

The structure of this process presents the opportunity for accurately performing diffusion, enforcing $Z_1^u \sim \mu$, while simultaneously pursuing additional criteria. Specifically:

1. Immediately we recognize that a nontrivial transportation problem implies the existence of a set (*i.e.*, a nonunique solution to the constraint satisfaction problem) of possible drifts such that the final distribution is μ. We can consider maximizing representativeness as an alternative cost criterion to $\int \|u_s\|^2 ds$. To present the criteria in a sensible way, given that the training is conducted on a minimum across mini-batches, we can instead aim to maximize a bottom quantile, by the cost-to-go functional,

$$
J_r(\mathbf{z}, t) = \int_t^1 Q_{\tilde{q}, w\sim\mu}\left[\sigma\left(Z_t, w\right)\right] ds,
$$

   where $\tilde{q}$ is the quantile percentage, *e.g.* 0.02 (for instance, if a mini-batch of fifty samples were given, this would be the minimum).

2. Next, notice that with dataset distillation, the small sample size is significant, which suggests that we can consider the aggregate in a particle framework, where for $i = 1, ..., N_D$, we have,

$$
dZ_t^{u,(i)} = u(Z_t^{u,(i)}, t)dt + dW_t,\ t \in [0, 1];\ Z_0^{u,(i)} = \mathbf{z}_0
$$

   presenting an additional degree of freedom, which we take advantage of by encouraging diversity, *i.e.*, minimizing

$$
J_d(\mathbf{z}, 1) = \max_{i,j=1,..,N_D} \sigma\left(Z_1^{u,(i)}, Z_1^{u,(j)}\right).
$$

   Since generation accuracy and representativeness are criteria for individual particles, maximizing diversity across particles can be considered as optimizing with respect to the additional degree of freedom introduced by having multiple particles.

Thus, we can see that it presents the opportunity to consider generative diffusion as a bi-level stochastic control problem.

A brief note on convergence guarantees for Eq. (7) presented in the main paper. A straightforward extension of [9] to three layers (similar to the extension from bi-level to tri-level convex optimization in [38]) yields convergence guarantees in expectation to a stationary point for all objectives. It is important to note that in the case of nonconvex objectives, the asymptotic point will satisfy a fairly weak condition. Specifically, it may not be stationary for the top objective, as the lower levels are not necessarily at global minimizers. This is, however, the best that can be ensured with stochastic gradient based methods and similar.

7. Related Works

7.1. Dataset Distillation

Dataset distillation (DD) aims to condense the information of large-scale datasets into small amounts of synthetic images with close training performance [5, 21, 47, 57]. The informative images are also useful for tasks like continual learning [16, 21], federated learning [25, 51] and neural architecture search [40]. Previous DD works can be roughly divided into bi-level optimization and training metric matching methods. Bi-level optimization methods incorporate meta learning into the surrogate image update [7, 27, 28, 30, 31, 59]. In comparison, metric matching methods optimize the synthetic images by matching the training gradients [21, 23, 26, 44, 54, 57], feature distribution [35, 45, 56, 58], predicted logits [46] or training trajectories [3, 11, 49] with original images.

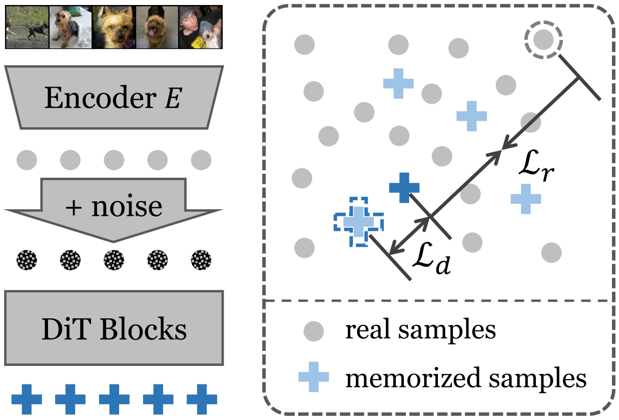


Figure 7. The training pipeline of the proposed minimax diffusion fine-tuning. The DiT blocks predict the added noise and original embeddings (dark-blue crossings). Then the parameters are updated with the simple diffusion objective and the minimax objectives. The minimax objectives (the right part) enforce the predicted embedding to be close to the farthest real sample and be far away from the closest predicted embedding of adjacent iterations.

7.2. Data Generation with Diffusion

The significantly improved image quality and sample diversity by diffusion models opens up new possibilities for data generation [8, 20, 22, 32]. Through prompt engineering [12, 19, 36], latent interpolation [60] and classifier-free guidance [1, 60], the diversity-improved synthetic images are useful to serve as augmentation or expansion for the original samples. The generated images also contribute to zero-shot image classification tasks [39]. However, these works mainly focus on recovering the original distribution with equal or much larger amounts of data. In contrast, we intend to distill the rich data information into small surrogate datasets. Moreover, prompt engineering usually requires special designs according to different data classes, while our proposed method saves extra effort. As far as we have investigated, there are no previous attempts to incorporate generative diffusion techniques into the dataset distillation task. In addition to diffusion models, there are also some previous works considering the diversity issue for Generative Adversarial Networks (GANs) [2, 17, 29, 52]. However, the improvement in diversity is not reflected in downstream tasks. In this work, we seek to enhance both representativeness and diversity for constructing a small surrogate dataset with similar training performance compared with original large-scale ones.

8. Method Pipeline

We demonstrate the pipeline of the proposed minimax fine-tuning method in Fig. 7. The real images are first passed through the encoder E to obtain the original embeddings $\mathbf{z}$. Random noise ϵ is then added to the embeddings by the diffusion process. The DiT blocks then predict the added noise, with which the predicted original embeddings $\hat{\mathbf{z}}$ (dark-blue crossings in Fig. 7) are also able to be calculated. We maintain two auxiliary memories $\mathcal{M}$ (grey dots) and $\mathcal{D}$ (light-blue crossings) to store the encountered real embeddings and predicted embeddings at adjacent iterations, respectively. The denoised embeddings of the current iteration are pushed away from the most similar predicted embedding and are pulled close to the least similar real embedding. The DiT blocks are optimized with the proposed minimax criteria and the simple diffusion training loss $\mathcal{L}_{simple}$ as in Eq. (1).

At the inference stage, given a random noise together with a specified class label, the DiT network predicts the noise that requires to be subtracted. Then the Decoder D recovers the images from the denoised embeddings.

Table 6. The training epoch number on different IPC settings for distilled dataset validation.

IPC	10	20	50	70	100
Epochs	2000	1500	1500	1000	1000

9. More Implementation Details

We conduct experiments on three commonly adopted network architectures in the area of DD, including:

1. **ConvNet-6** is a 6-layer convolutional network. In previous DD works where small-resolution images are distilled, the most popular network is ConvNet-3 [21, 26, 54]. We extend an extra 3 layers for full-sized 256×256 ImageNet data. The network contains 128 feature channels in each layer, and instance normalization is adopted.
2. **ResNetAP-10** is a 10-layer ResNet [18], where the strided convolution is replaced by average pooling for downsampling.
3. **ResNet-18** is a 18-layer ResNet [18] with instance normalization (IN). As the IN version performs better than batch normalization under our protocol, we uniformly adopt IN for the experiments.

For diffusion fine-tuning, an Adam optimizer is adopted with the learning rate set as 0.001, which is consistent with the original Difffit setting [50]. We set the mini-batch size as 8 mainly due to the GPU memory limitation. The employed augmentations during the fine-tuning stage include random resize-crop and random flip.

For the validation training, we adopt the same protocol as in [21]. Specifically, a learning rate of 0.01 for an Adam optimizer is adopted. The training epoch setting is presented in Tab. 6. The reduced training epochs also partly explain the reason why the performance gap between the IPC set-

Table 7. Performance comparison on ImageNet-100. The best results are marked as **bold**.

IPC (Ratio)	Test Model	Random	Herding [48]	IDC-1 [21]	Ours	Full
10 (0.8%)	ConvNet-6	$17.0_{\pm0.3}$	$17.2_{\pm0.3}$	$\mathbf{24.3}_{\pm0.5}$	$22.3_{\pm0.5}$	$79.9_{\pm0.4}$
	ResNetAP-10	$19.1_{\pm0.4}$	$19.8_{\pm0.3}$	$\mathbf{25.7}_{\pm0.1}$	$24.8_{\pm0.2}$	$80.3_{\pm0.2}$
	ResNet-18	$17.5_{\pm0.5}$	$16.1_{\pm0.2}$	$\mathbf{25.1}_{\pm0.2}$	$22.5_{\pm0.3}$	$81.8_{\pm0.7}$
20 (1.6%)	ConvNet-6	$24.8_{\pm0.2}$	$24.3_{\pm0.4}$	$28.8_{\pm0.3}$	$\mathbf{29.3}_{\pm0.4}$	$79.9_{\pm0.4}$
	ResNetAP-10	$26.7_{\pm0.5}$	$27.6_{\pm0.1}$	$29.9_{\pm0.2}$	$\mathbf{32.3}_{\pm0.1}$	$80.3_{\pm0.2}$
	ResNet-18	$25.5_{\pm0.3}$	$24.7_{\pm0.1}$	$30.2_{\pm0.2}$	$\mathbf{31.2}_{\pm0.1}$	$81.8_{\pm0.7}$

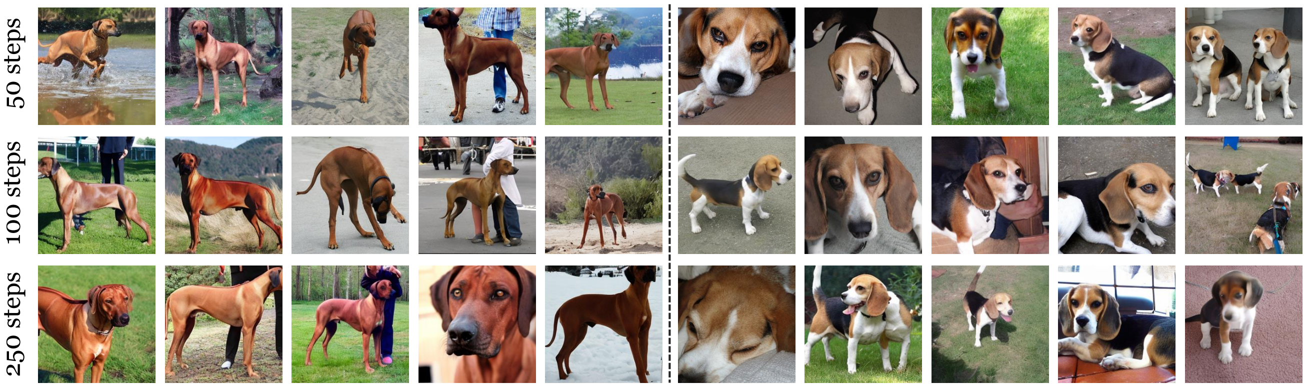


Figure 8. Visualization of images generated by the same model with different denoising steps. For each column, the generated images are based on the same random seed.

Table 8. The influence of diffusion denoising step number on the generation time of each image and the corresponding validation performance. Performance evaluated with ResNet-10 on ImageWoof. The best results are marked as **bold**.

		Denoising Step		
		50	100	250
Time (s)		0.8	1.6	3.2
IPC	10	$39.2_{\pm1.3}$	$35.7_{\pm0.7}$	$\mathbf{39.6}_{\pm0.9}$
	20	$\mathbf{45.8}_{\pm0.5}$	$44.5_{\pm0.6}$	$43.7_{\pm0.7}$
	50	$56.3_{\pm1.0}$	$\mathbf{58.4}_{\pm0.5}$	$55.8_{\pm0.5}$
	70	$58.3_{\pm0.2}$	$\mathbf{59.6}_{\pm1.1}$	$58.9_{\pm1.4}$
	100	$\mathbf{64.5}_{\pm0.2}$	$63.3_{\pm0.7}$	$62.8_{\pm0.6}$

tings of 50 and 70 is relatively small. The adopted data augmentations include random resize-crop and CutMix.

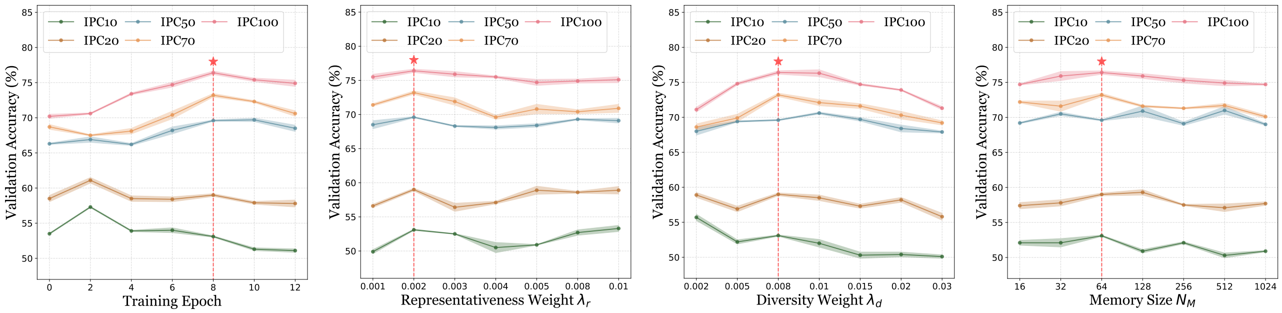
10. More Analysis and Discussion

10.1. Experiments to ImageNet-100

In addition to the 10-class ImageNet subsets and full ImageNet-1K, we also conduct experiments on ImageNet-100, and the results are shown in Tab. 7. The validation protocol follows that in IDC [21]. Due to the limitation of computational resources, here we directly employ the official distilled images of IDC-1 [21] for evaluation. The original resolution is 224×224, and we resize the images to 256×256 for fair comparison. Under the IPC setting of 10, IDC-1 achieves the best performance. Yet when the IPC increases, the performance gap between the distilled images of IDC-1 and randomly selected original images is smaller. Comparatively, our proposed minimax diffusion method consistently provides a stable performance improvement over original images across different IPC settings. *It is worth noting that for IDC-1, the distillation process on ImageNet-100 demands hundreds of hours, while the proposed minimax diffusion only requires* ***10 hours***. The significantly reduced training time offers much more application possibilities for the dataset distillation techniques.

10.2. Diffusion Denoising Step

In our experiments, we set the diffusion denoising step number as 50. We evaluate its influence on the validation performance in Tab. 8. There are no fixed patterns for achieving better performance across all the IPCs. Additionally, we compare the generated images under different step settings in Fig. 8. For DiT [33], the denoising process is conducted in the embedding space. Therefore, it is reasonable that with different steps the generated images are variant in the pixel



(a) (b) (c) (d)

Figure 9. Hyper-parameter analysis on (a) the training epochs; (b) the representativeness weight λ_r; (c) the diversity weight λ_d; (d) the memory size N_M. The results are obtained with ResNetAP-10 on ImageIDC. The dashed line indicates the value adopted in this work.

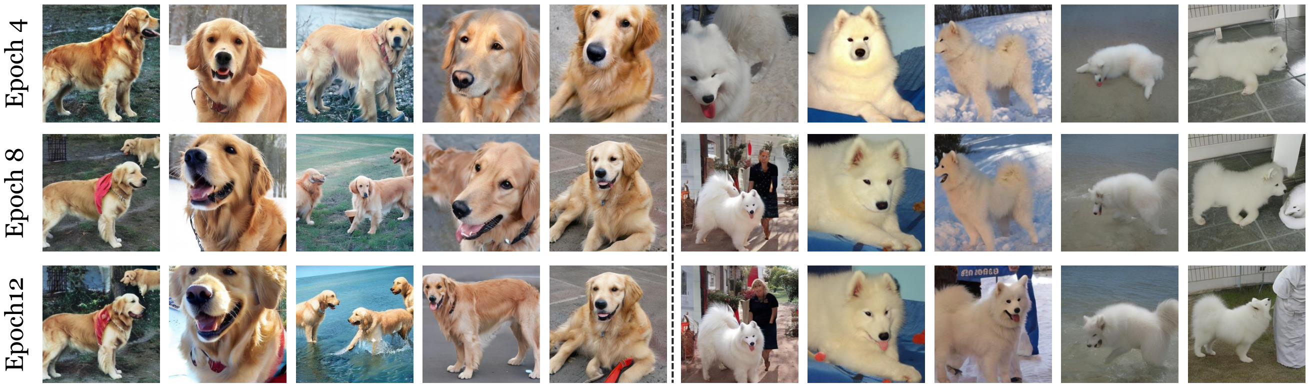


Figure 10. Visualization of images generated by models after different epochs of training. For each column, the images are generated based on the same random noise.

space. It can be observed that under all steps, the model generates high-quality images with sufficient diversity. Taking the calculation time into consideration, we simply select 50 steps in our experiments.

10.3. Parameter Analysis on ImageIDC

We extensively demonstrate the parameter analysis on ImageIDC to illustrate the robustness of the hyper-parameters. Fig. 9a shows the performance curve along the training epochs. As the training process starts, the representativeness constraint quickly improves the accuracy of small IPCs. Further training enhances the diversity, where the performance on large and small IPCs shows different trends. Generally, the generated images achieve the best performance at the 8th epoch, which is consistent with the ImageWoof experiments.

Compared with the results on ImageWoof, further enlarging the representativeness weight λ_r improves the performance on small IPCs, as illustrated in Fig. 9b. In comparison, increasing diversity causes a drastic performance drop in Fig. 9b. Although the default settings remain relatively better choices, the balance point between representativeness and diversity is worthy of further exploration. The memory size N_M merely has a mild influence on the performance, which aligns with that of ImageWoof.

Table 9. The dataset expansion results of the 100-IPC generated images on ImageWoof.

Test Model	Original	Original + 100-IPC
ConvNet-6	$86.4_{\pm 0.2}$	$\mathbf{87.0}_{\pm 0.6}$
ResNetAP-10	$87.5_{\pm 0.5}$	$\mathbf{89.3}_{\pm 0.6}$
ResNet-18	$89.3_{\pm 1.2}$	$\mathbf{90.1}_{\pm 0.3}$

10.4. Extension to Dataset Expansion

In addition to the standard dataset distillation task, where a small surrogate dataset is generated to replace the original one, we also evaluate the capability of the generated images as an expanded dataset. We add the generated 100-IPC surrogate dataset to the original ImageWoof (approximately 1,300 images per class) and conduct the validation in Tab. 9. As can be observed, although the extra images only take up

Table 10. The averaged generation quality evaluation of 10 classes each with 100 images in ImageWoof.

Method	FID	Precision (%)	Recall (%)	Coverage (%)
DM [56]	208.6	22.1	23.8	5.8
DiT [33]	81.4	92.8	38.9	24.1
DiT+$\mathcal{L}_r$	85.4	93.2	38.1	24.6
DiT+$\mathcal{L}_d$	81.1	90.4	46.8	28.3
Ours full	81.5	92.4	45.3	28.6

a small ratio compared with the original data, a considerable performance improvement is still achieved. *The results support that the proposed minimax diffusion can also be explored as a dataset expansion method in future works.*

10.5. Generated Samples of Different Epochs

We visualize the images generated by models after different epochs of training in Fig. 10 to explicitly demonstrate the training effect of the proposed minimax diffusion method. As the training proceeds, the generated images present variation trends from several perspectives. Firstly, the images tend to have more complicated backgrounds and environments, such as more realistic water and objects of other categories (*e.g.* human). Secondly, there are more details filled in the images, like the clothes in the first column and the red spots in the sixth. These new facets significantly enhance the diversity of the generated surrogate dataset. Furthermore, through the fine-tuning process, the class-related features are also enhanced. In the ninth and tenth columns, the model at the fourth epoch fails to generate objects with discriminative features. In comparison, the images generated by subsequent models demonstrate substantial improvement regarding the representativeness property.

10.6. Generation Quality Evaluation.

We further report quantitative evaluations on the generation quality by adding the proposed minimax criteria in Tab. 10. The representativeness and diversity constraints improve the precision and recall of the generated data, respectively. The full method finds a balanced point between these two properties while obtaining the best coverage over the whole distribution. The fine-tuning brings negligible influence on the FID metric. And all the metrics of our proposed method are significantly better than those attained by DM [56].

10.7. Generated Samples of Different Classes

We present the comparison between the samples selected by Herding [48] and those generated by our proposed minimax diffusion method on ImageNet-100 from Fig. 11 to Fig. 20. In most cases, the diffusion model is able to generate realistic images, which cannot easily be told from real samples. Herding also aims to select both representative and diverse samples. However, the lack of supervision on the semantic level led to the inclusion of noisy samples. For instance, the walking stick class contains images of mantis, which can originally be caused by mislabeling. The proposed minimax diffusion, in comparison, accurately generates images of the corresponding classes, which is also validated by the better performance shown in Tab. 7. There are also some failure cases for the diffusion model. The fur texture of hairy animals like Shih-Tzu and langur is unrealistic. The structures of human faces and hands also require further refinement. We treat these defects as exploration directions of future works for both diffusion models and the dataset distillation usage.

11. Broader Impacts

The general purpose of dataset distillation is to reduce the demands of storage and computational resources for training deep neural networks. The requirement of saving resource consumption is even tenser at the age of foundation models. Dataset distillation aims to push forward the process of environmental contributions. From this perspective, the proposed minimax diffusion method significantly reduces the requirement resources for the distillation process itself. We hope that through this work, the computer vision society can put more attention on practical dataset distillation methods, which are able to promote the sustainable development of society.

12. Ethical Considerations

There are no direct ethical issues attached to this work. We employ the publicly available ImageNet dataset for experiments. In future works, we will also be devoted to considering the generation bias and diversity during constructing a small surrogate dataset.

References

[1] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023. 2

[2] Victor Besnier, Himalaya Jain, Andrei Bursuc, Matthieu Cord, and Patrick Pérez. This Dataset Does Not Exist: Training Models from Generated Images. In *ICASSP*, pages 1–5, 2020. 2

[3] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *CVPR*, pages 4750–4759, 2022. 1

[4] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Generalizing dataset distillation via deep generative prior. In *CVPR*, pages 3739–3748, 2023. 1, 5, 6

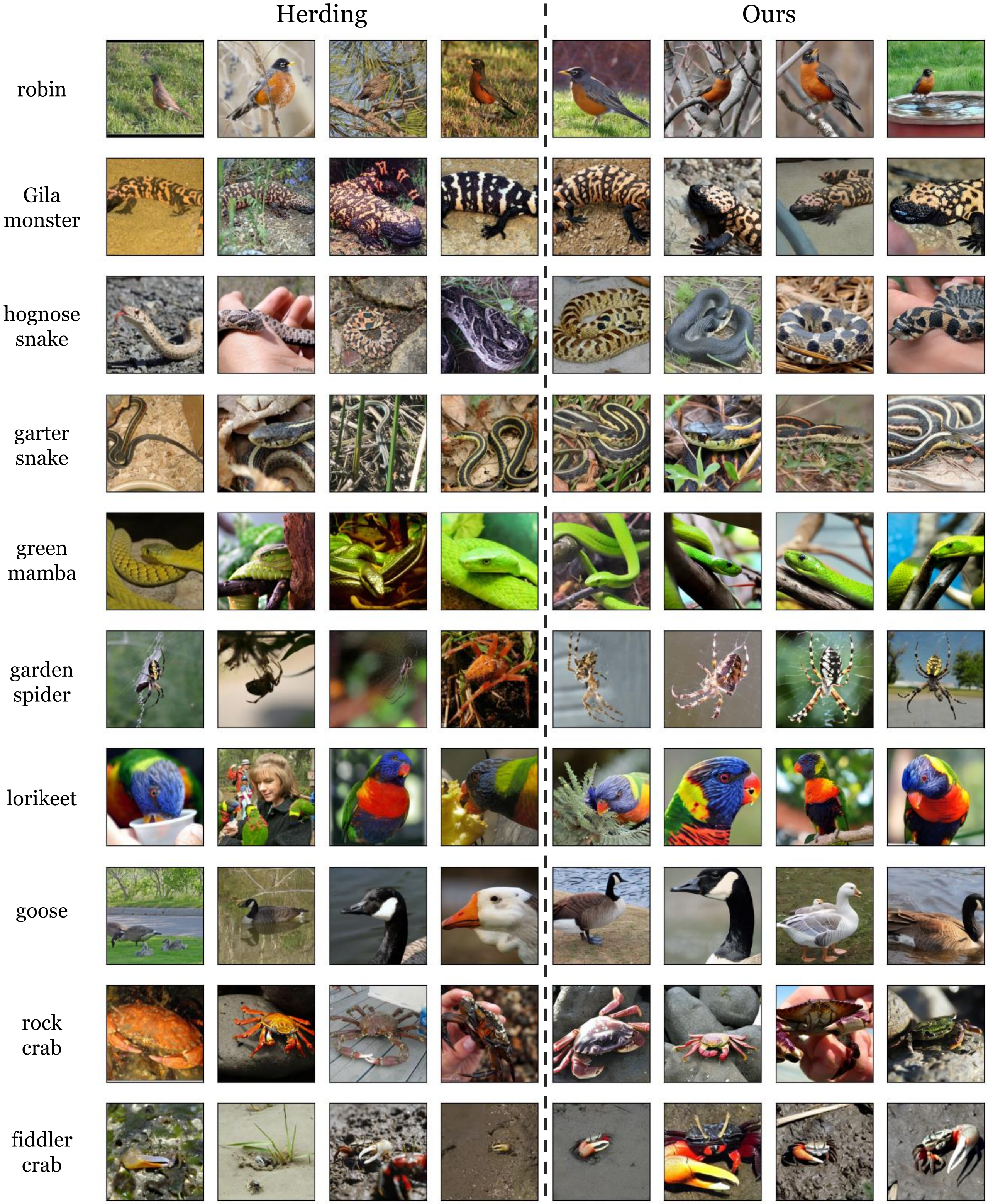


Figure 11. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 0-9. The class names are marked at the left of each row.

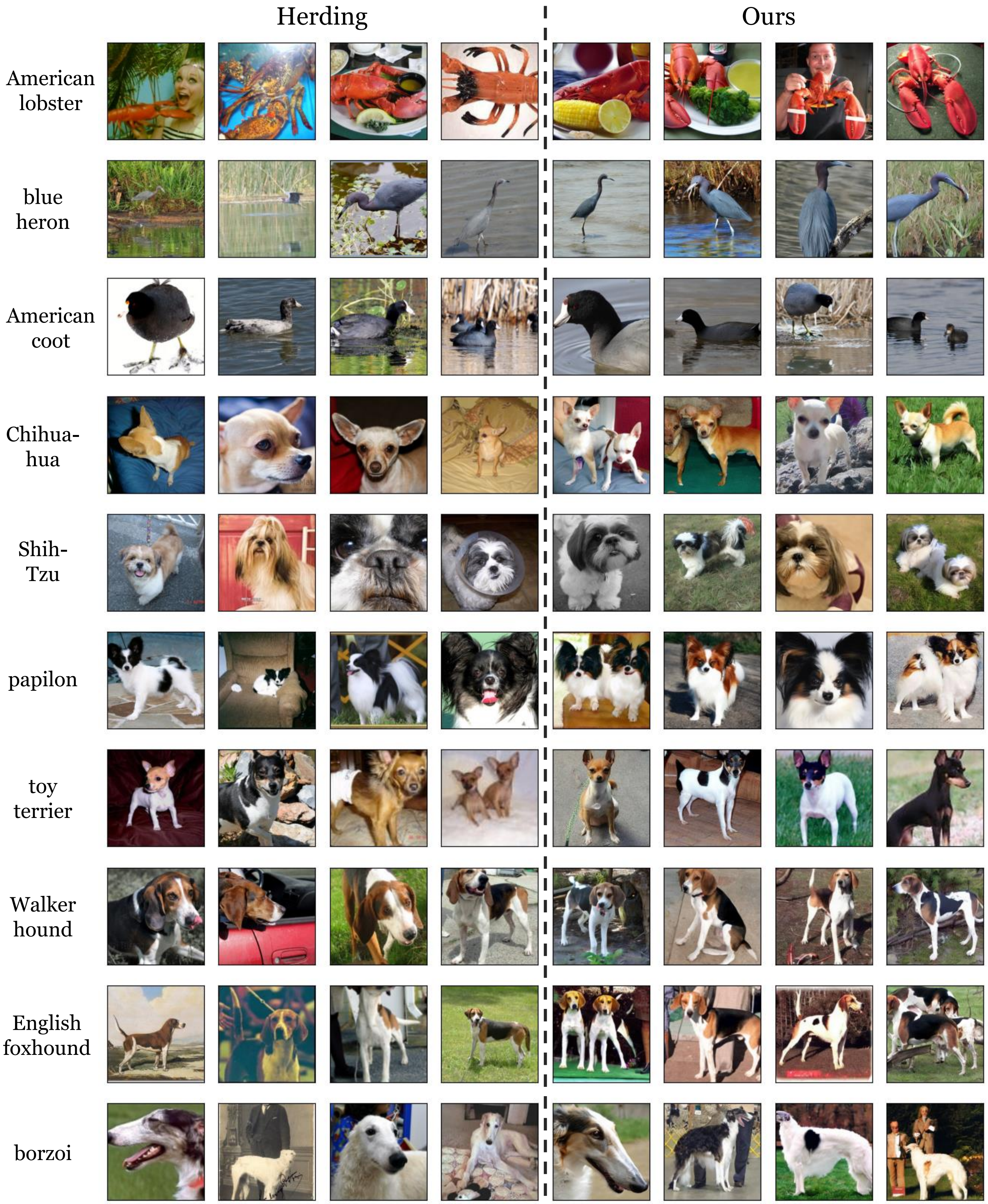


Figure 12. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 10-19. The class names are marked at the left of each row.

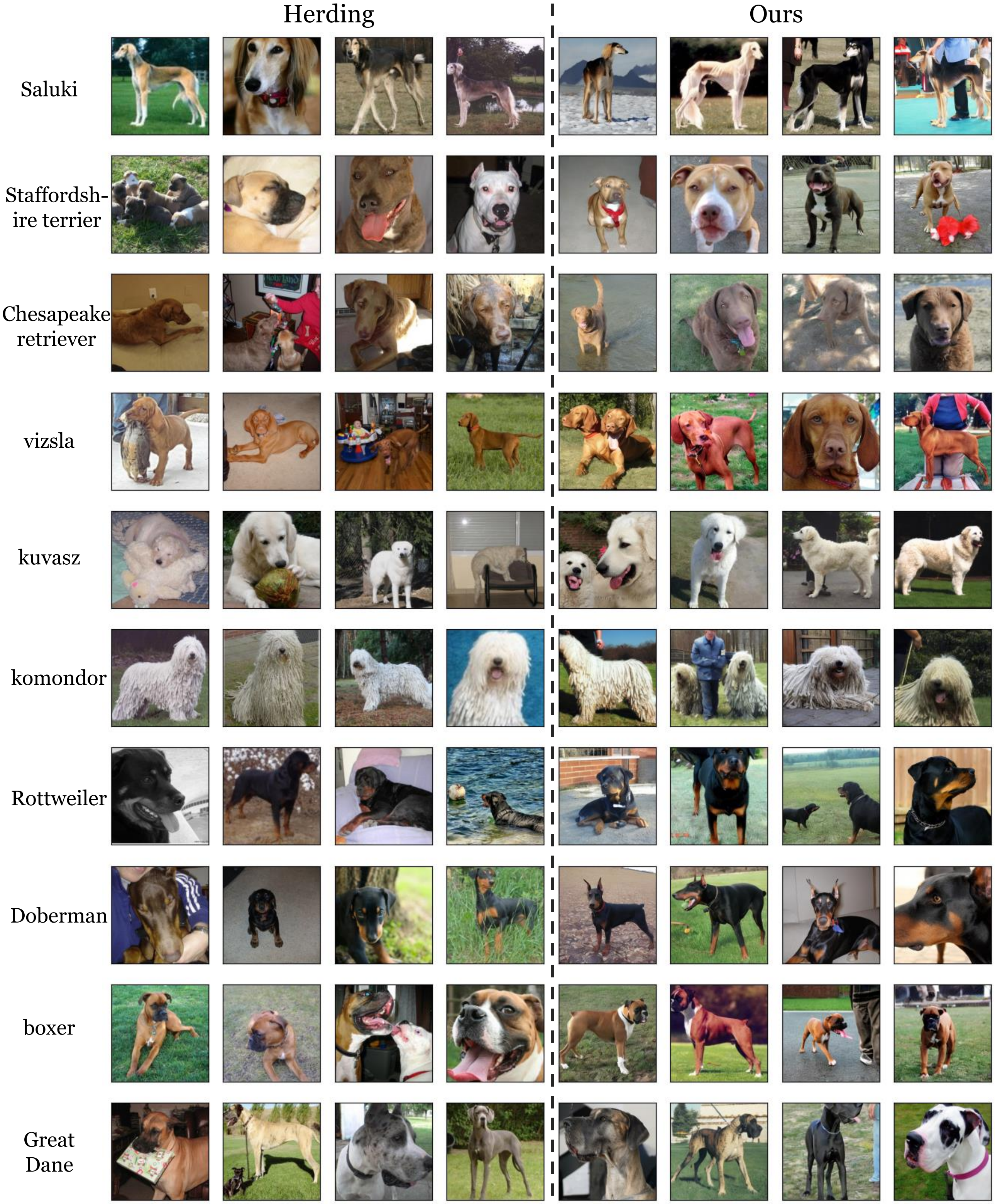


Figure 13. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 20-29. The class names are marked at the left of each row.

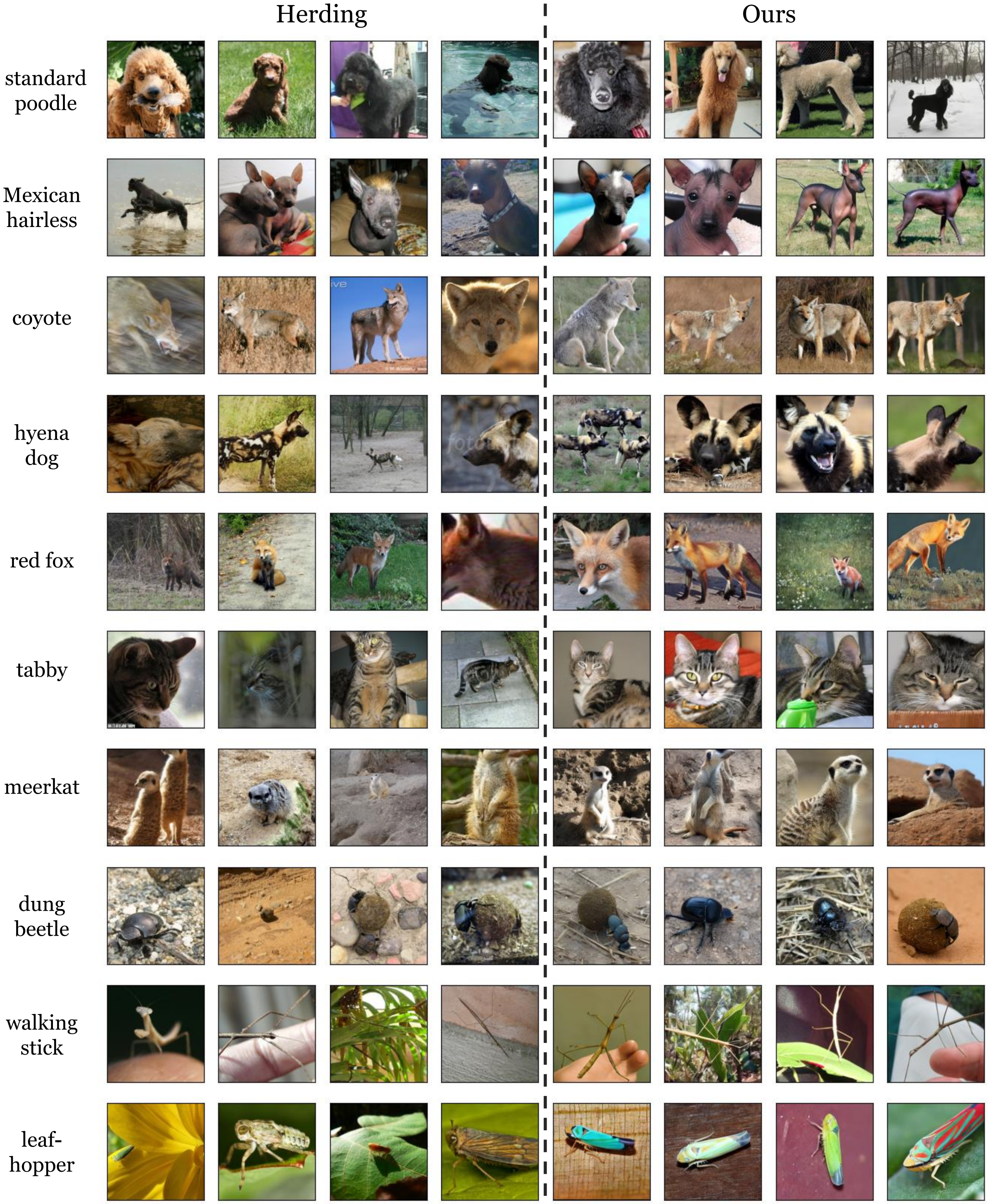


Figure 14. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 30-39. The class names are marked at the left of each row.

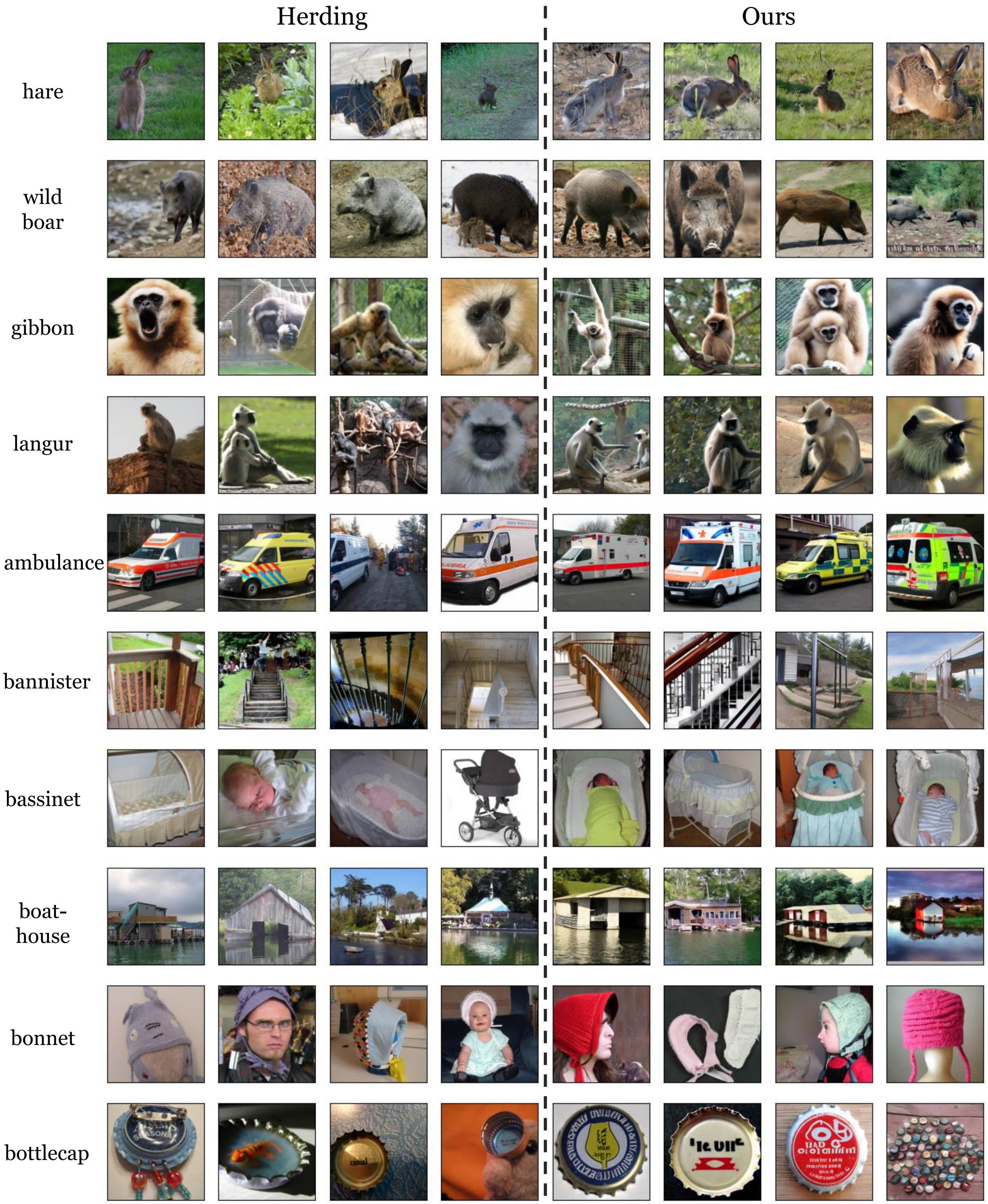


Figure 15. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 40-49. The class names are marked at the left of each row.

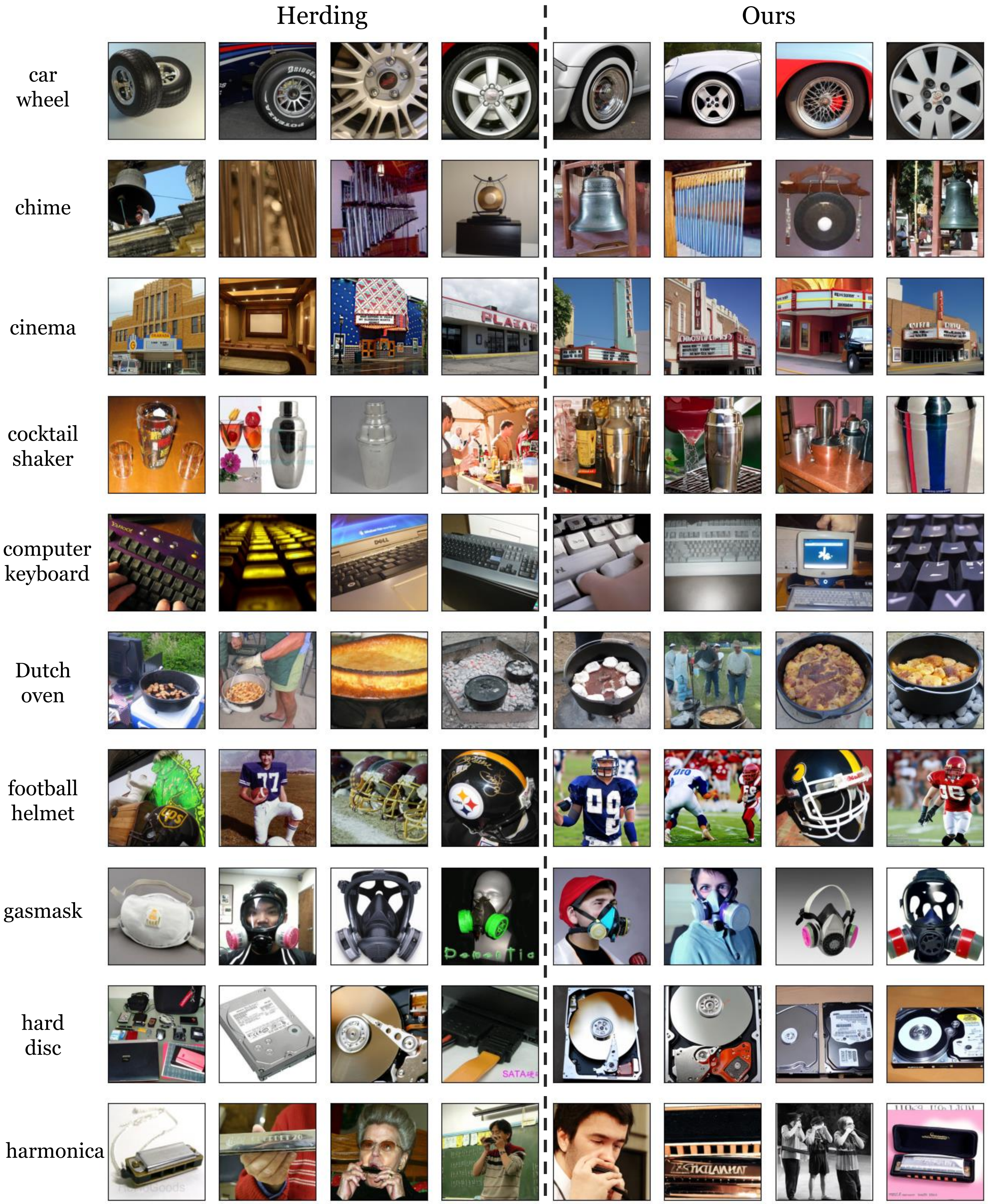


Figure 16. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 50-59. The class names are marked at the left of each row.

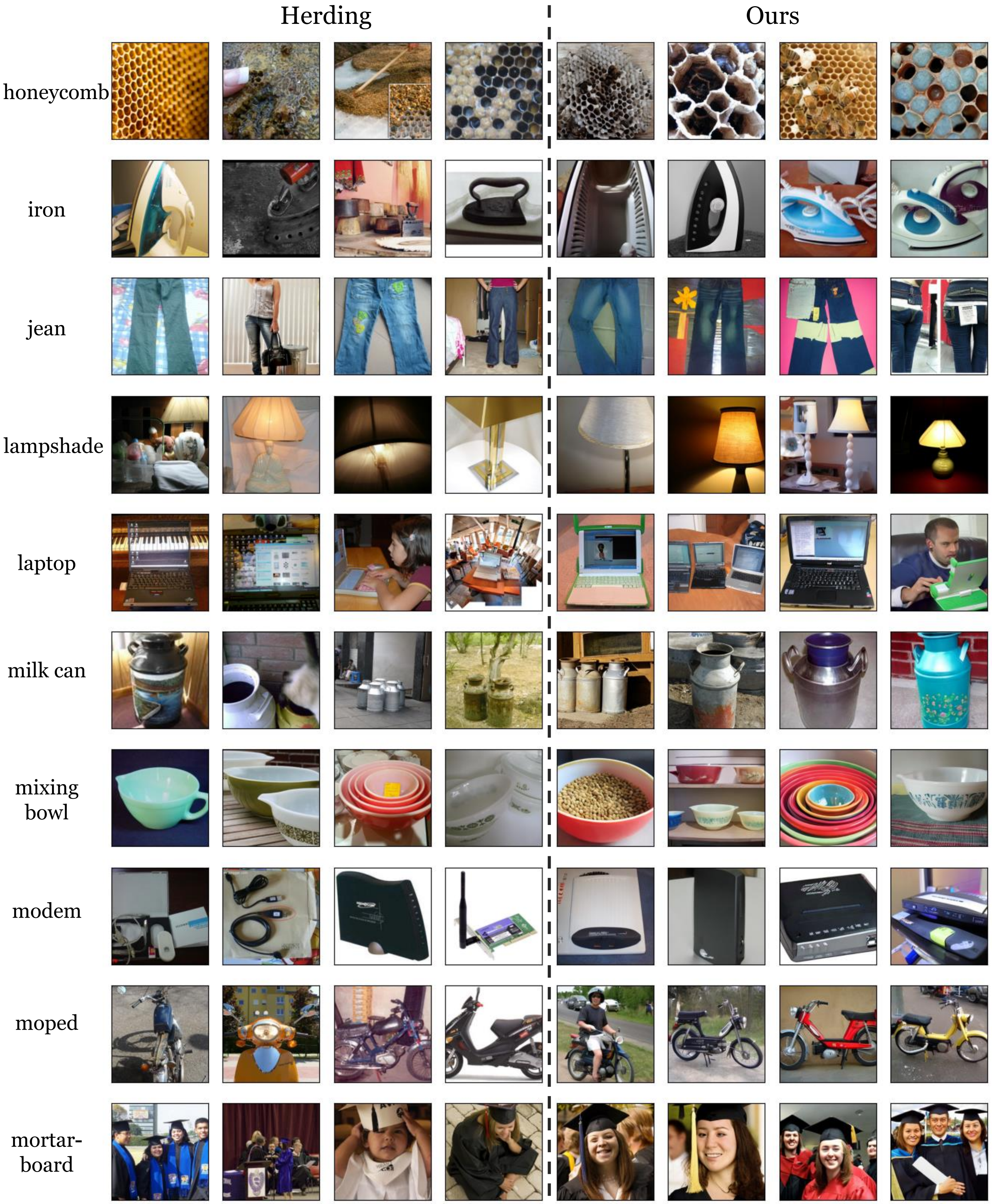


Figure 17. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 60-69. The class names are marked at the left of each row.

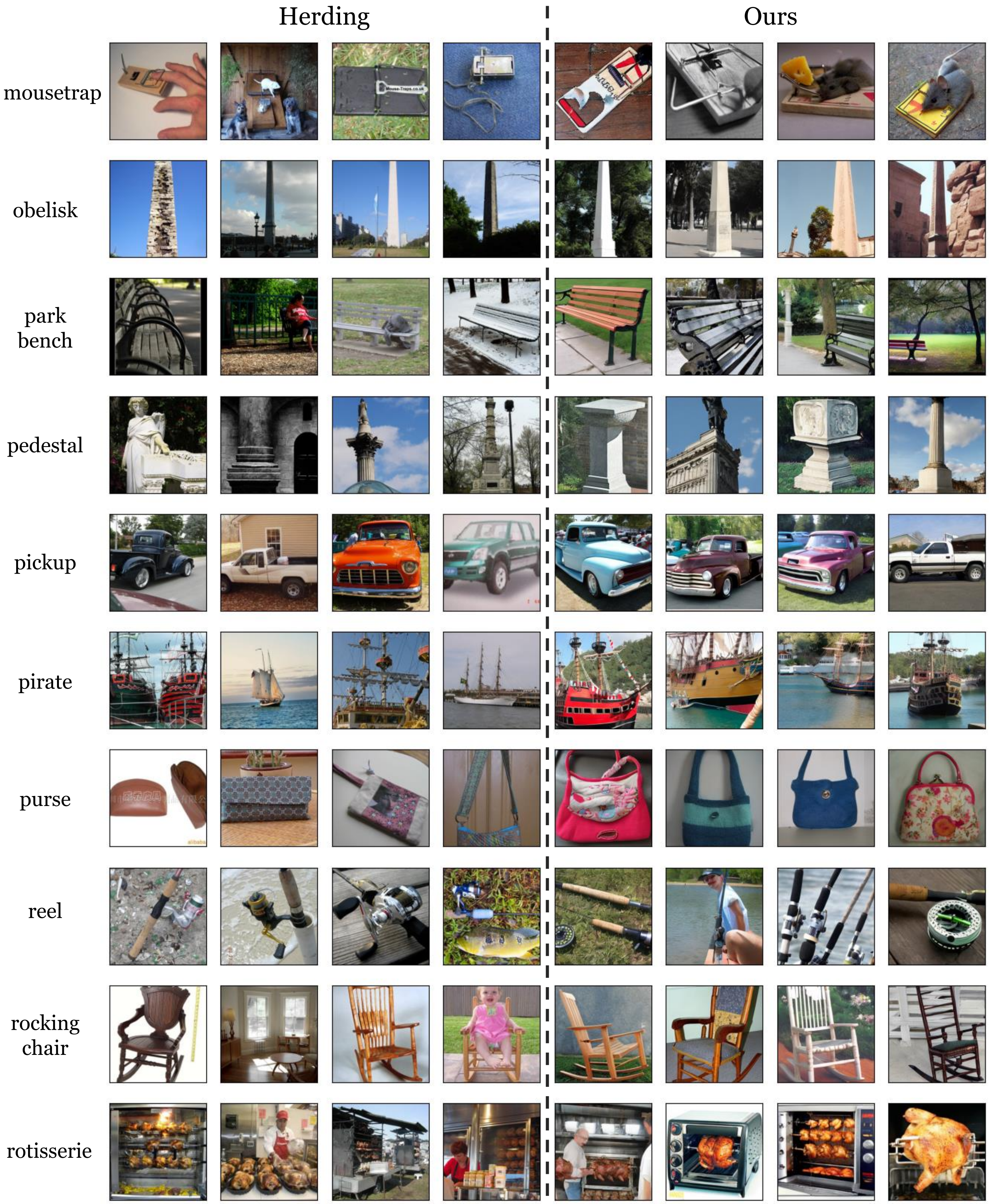


Figure 18. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 70-79. The class names are marked at the left of each row.

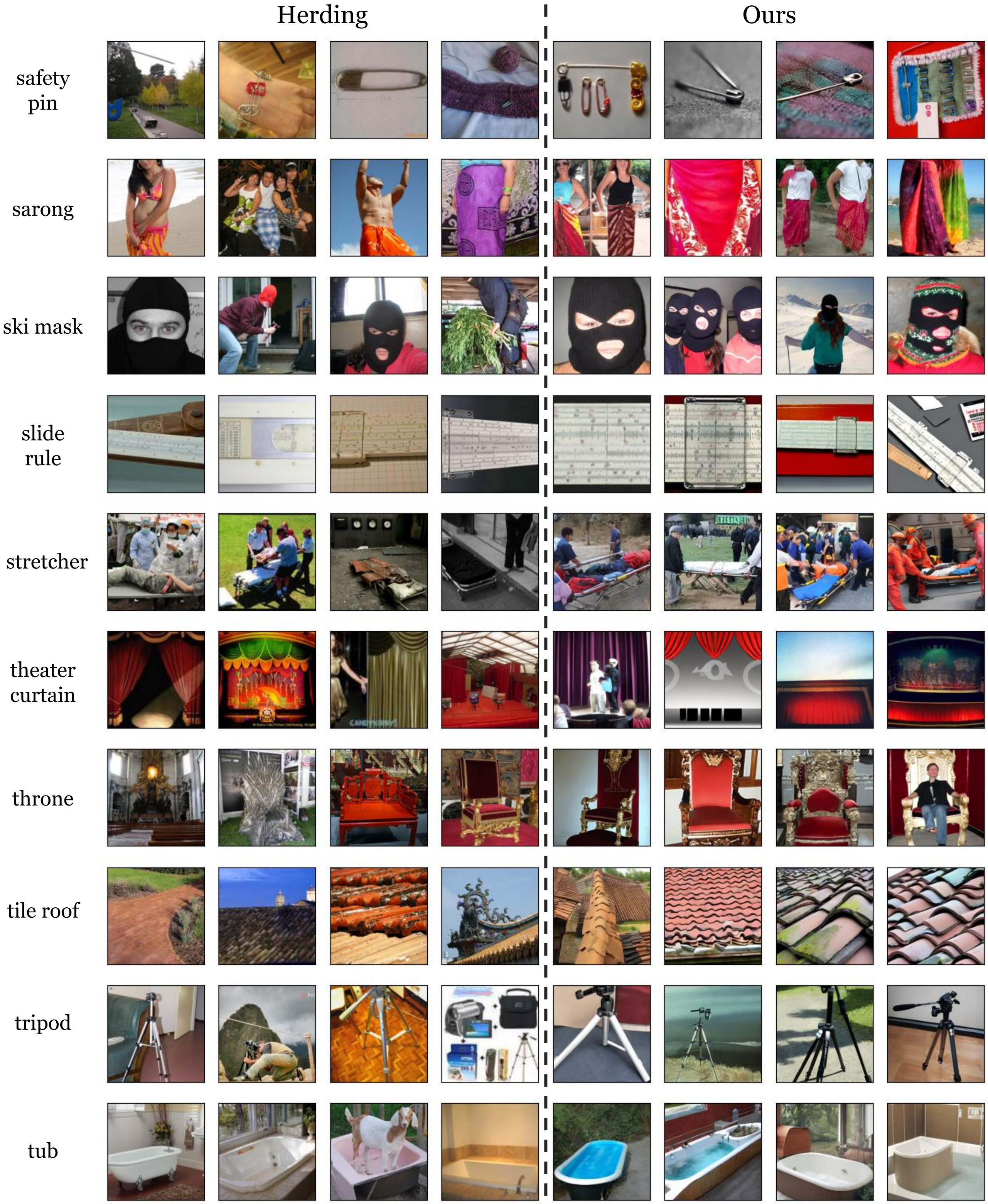


Figure 19. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 80-89. The class names are marked at the left of each row.

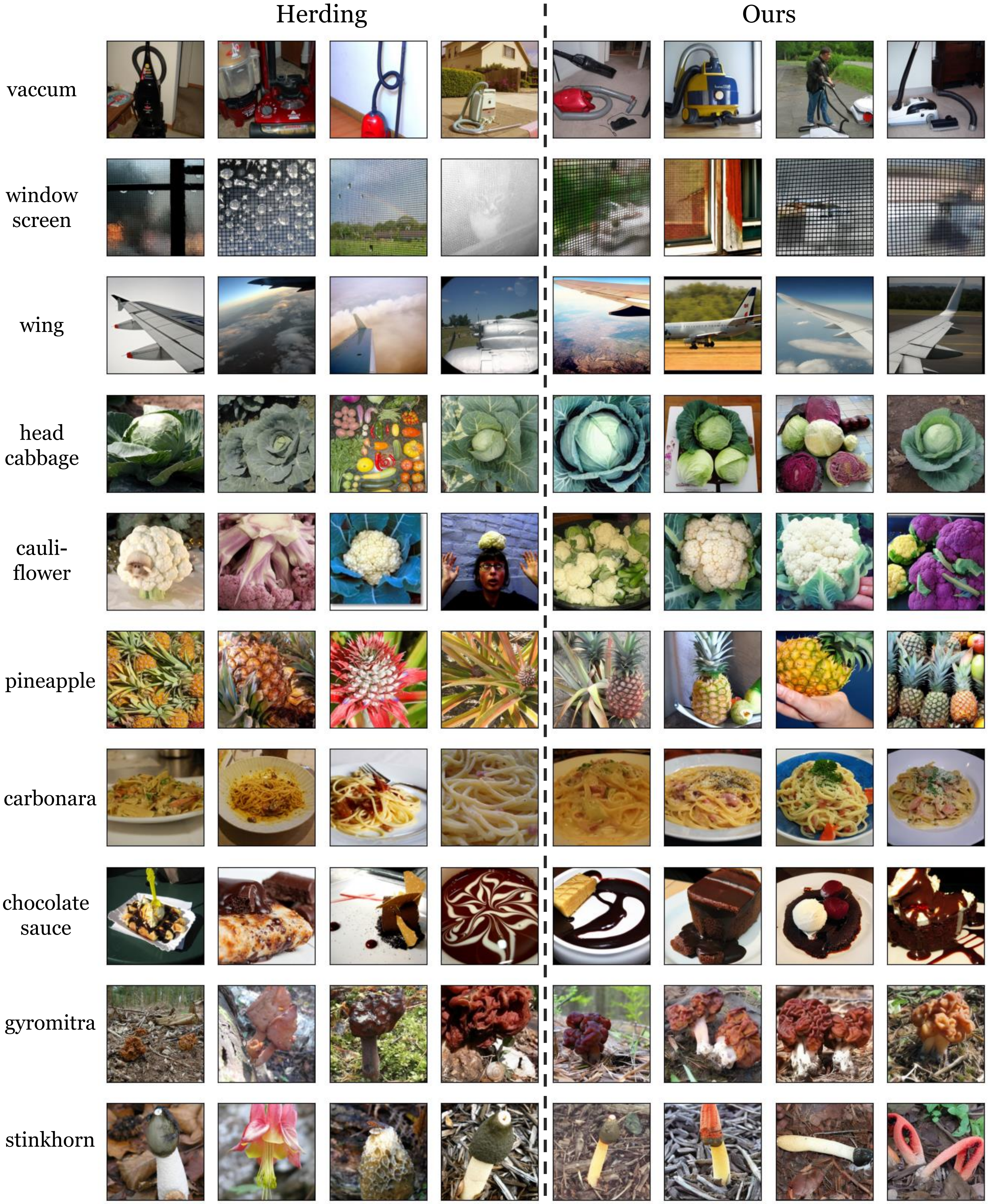


Figure 20. Comparison between samples selected by Herding (left) and generated by the proposed minimax diffusion method (right) for ImageNet-100 classes 90-99. The class names are marked at the left of each row.

[5] Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Dc-bench: Dataset condensation benchmark. *NeurIPS*, 35:810–822, 2022. 1

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 1, 5

[7] Zhiwei Deng and Olga Russakovsky. Remember the Past: Distilling Datasets into Addressable Memories for Neural Networks. In *NeurIPS*, 2022. 1

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. In *NeurIPS*, pages 8780–8794, 2021. 3, 2

[9] Thinh T Doan. Nonlinear two-time-scale stochastic approximation convergence and finite-time performance. *IEEE Transactions on Automatic Control*, 2022. 1

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2022. 1

[11] Jiawei Du, Yidi Jiang, Vincent Y.F. Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the Accumulated Trajectory Error to Improve Dataset Distillation. In *CVPR*, pages 3749–3758, 2023. 1

[12] Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. *arXiv preprint arXiv:2305.16289*, 2023. 2

[13] Gabriele Eichfelder. Scalarizations for adaptively solving multi-objective optimization problems. *Computational Optimization and Applications*, 44:249–273, 2009. 4

[14] Ronen Eldan and James R Lee. Regularization under diffusion and anticoncentration of the information content. *Duke Mathematical Journal*, 167(5):969–993, 2018. 1

[15] Fastai. Fastai/imagenette: A smaller subset of 10 easily classified classes from imagenet, and a little more french. 1, 5, 6

[16] Jianyang Gu, Kai Wang, Wei Jiang, and Yang You. Summarizing stream data for memory-restricted online continual learning. *arXiv preprint arXiv:2305.16645*, 2023. 1

[17] Swaminathan Gurumurthy, Ravi Kiran Sarvadevabhatla, and R. Venkatesh Babu. DeLiGAN: Generative Adversarial Networks for Diverse and Limited Data. In *CVPR*, pages 4941–4949, 2017. 2

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 5, 2

[19] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is Synthetic Data from Generative Models Ready for Image Recognition? In *ICLR*, 2022. 2

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *NeurIPS*, pages 6840–6851, 2020. 2

[21] Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoo Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *ICML*, pages 11102–11118, 2022. 1, 5, 6, 2, 3

[22] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational Diffusion Models. In *NeurIPS*, pages 21696–21707, 2021. 2

[23] Saehyung Lee, Sanghyuk Chun, Sangwon Jung, Sangdoo Yun, and Sungroh Yoon. Dataset condensation with contrastive signals. In *ICML*, pages 12352–12364, 2022. 1

[24] Joseph Lehec. Representation formula for the entropy and functional inequalities. In *Annales de l'IHP Probabilités et statistiques*, pages 885–899, 2013. 1

[25] Ping Liu, Xin Yu, and Joey Tianyi Zhou. Meta Knowledge Condensation for Federated Learning. In *ICLR*, 2022. 1

[26] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *ICCV*, 2023. 1, 2

[27] Noel Loo, Ramin Hasani, Alexander Amini, and Daniela Rus. Efficient dataset distillation using random feature approximation. *NeurIPS*, 35:13877–13891, 2022. 1

[28] Noel Loo, Ramin Hasani, Mathias Lechner, and Daniela Rus. Dataset distillation with convexified implicit gradients. *arXiv preprint arXiv:2302.06755*, 2023. 1

[29] Qi Mao, Hsin-Ying Lee, Hung-Yu Tseng, Siwei Ma, and Ming-Hsuan Yang. Mode Seeking Generative Adversarial Networks for Diverse Image Synthesis. In *CVPR*, pages 1429–1437, 2019. 2

[30] Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. In *ICLR*, 2021. 1

[31] Timothy Nguyen, Roman Novak, Lechao Xiao, and Jaehoon Lee. Dataset distillation with infinitely wide convolutional networks. *NeurIPS*, 34:5186–5198, 2021. 1

[32] Alexander Quinn Nichol and Prafulla Dhariwal. Improved Denoising Diffusion Probabilistic Models. In *ICML*, pages 8162–8171, 2021. 2

[33] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, pages 4195–4205, 2023. 3, 5, 6, 7

[34] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, pages 22500–22510, 2023. 3

[35] Ahmad Sajedi, Samir Khaki, Ehsan Amjadian, Lucy Z Liu, Yuri A Lawryshyn, and Konstantinos N Plataniotis. DataDAM: Efficient Dataset Distillation with Attention Matching. In *ICCV*, 2023. 1

[36] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it Till You Make it: Learning Transferable Representations from Synthetic ImageNet Clones. In *CVPR*, pages 8011–8021, 2023. 2

[37] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. 5

[38] Allahkaram Shafiei, Vyacheslav Kungurtsev, and Jakub Marecek. Trilevel and multilevel optimization using monotone operator theory. *arXiv preprint arXiv:2105.09407*, 2021. 1

[39] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity Is Definitely Needed: Improving Model-Agnostic Zero-Shot Classification via Stable Diffusion. In *CVPR*, pages 769–778, 2023. 2

[40] Felipe Petroski Such, Aditya Rawal, Joel Lehman, Kenneth Stanley, and Jeffrey Clune. Generative Teaching Networks: Accelerating Neural Architecture Search by Learning to Generate Synthetic Training Data. In *ICML*, pages 9206–9216, 2020. 1

[41] Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm. In *CVPR*, 2024. 6

[42] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, pages 776–794, 2020. 5

[43] Belinda Tzen and Maxim Raginsky. Theoretical guarantees for sampling and inference in generative models with latent diffusions. In *COLT*, pages 3084–3114, 2019. 4, 5, 1

[44] Saeed Vahidian, Mingyu Wang, Jianyang Gu, Vyacheslav Kungurtsev, Wei Jiang, and Yiran Chen. Group distributionally robust dataset distillation with risk minimization. *arXiv preprint arXiv:2402.04676*, 2024. 1

[45] Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features. In *CVPR*, pages 12196–12205, 2022. 1

[46] Kai Wang, Jianyang Gu, Daquan Zhou, Zheng Zhu, Wei Jiang, and Yang You. Dim: Distilling dataset into generative model. *arXiv preprint arXiv:2303.04707*, 2023. 1

[47] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. 1, 2

[48] Max Welling. Herding dynamical weights to learn. In *ICML*, pages 1121–1128, 2009. 5, 6, 3

[49] Xindi Wu, Zhiwei Deng, and Olga Russakovsky. Multimodal dataset distillation for image-text retrieval. *arXiv preprint arXiv:2308.07545*, 2023. 1

[50] Enze Xie, Lewei Yao, Han Shi, Zhili Liu, Daquan Zhou, Zhaoqiang Liu, Jiawei Li, and Zhenguo Li. Difffit: Unlocking transferability of large diffusion models via simple parameter-efficient fine-tuning. *arXiv preprint arXiv:2304.06648*, 2023. 3, 5, 7, 2

[51] Yuanhao Xiong, Ruochen Wang, Minhao Cheng, Felix Yu, and Cho-Jui Hsieh. FedDM: Iterative Distribution Matching for Communication-Efficient Federated Learning. In *CVPR*, pages 16323–16332, 2023. 1

[52] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tianchen Zhao, and Honglak Lee. Diversity-Sensitive Conditional Generative Adversarial Networks. In *ICLR*, 2018. 2

[53] Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. In *NeurIPS*, 2023. 6

[54] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *ICML*, pages 12674–12685, 2021. 1, 2

[55] Bo Zhao and Hakan Bilen. Synthesizing informative training samples with gan. *arXiv preprint arXiv:2204.07513*, 2022. 1

[56] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *WACV*, pages 6514–6523, 2023. 2, 5, 6, 1

[57] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *ICLR*, 2021. 1, 2

[58] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved Distribution Matching for Dataset Condensation. In *CVPR*, pages 7856–7865, 2023. 1

[59] Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression. *NeurIPS*, 35:9813–9827, 2022. 1

[60] Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with

generated data. *arXiv preprint arXiv:2305.15316*, 2023. 2