Supplementary Materials for DiffClass

Zichong Meng[1], Jie Zhang[2], Changdi Yang[1], Zheng Zhan[1], Pu Zhao*[1], and Yanzhi Wang*[1]

[1] Northeastern University, Boston MA 02115, USA
[2] ETH Zürich, 8092 Zürich, Switzerland

We further discuss our proposed approach with the following supplementary materials:

- Appendix A: Visualization of Synthetic Images.
- Appendix B: Additional Experimental Results.
- Appendix C: Performace on Synthetic and Combined Test Data.
- Appendix D: Detailed Derivation for Equation (1).
- Appendix E: Potential Improvements for Our Method.

A Visualization of Synthetic Images

We present the synthetic images generated using our method in Fig. A1. We sample those synthetic images from the last incremental task on ImageNet100 with $N = 20$. We also present synthetic images generated from Stable Diffusion V1.5 and real images of the same classes for comparison.

From Fig. A1, we can see synthetic images with full details from our exemplar-free CIL framework are closer to real images of the same classes in terms of style, context, and domain. At the same time, they also preserve real data privacy without remembering absolute details.

B Additional Experimental Results

To further demonstrate the effectiveness of our method, we conduct additional experiments for CIL settings with a larger first task on the ImageNet100 dataset. Following previous works [1, 3, 4, 6–8], we conduct experiments for three CIL settings: (i) $N = 6$ with 50 classes for the first task and 10 classes for each of the other 5 tasks (6 tasks in total); (ii) $N = 11$ with 50 classes for the first task and 5 classes for each of the other 10 tasks (11 tasks in total); (iii) $N = 21$ with 40 classes for the first task and 3 classes for each of the other 20 tasks (21 tasks in total).

We present the experimental results in Tab. A1. We notice that although certain methods (such as PASS, IL2A, SSRE, FeTrIL, SSRE, and DDGR) originally perform poorly in equally split tasks, they achieve significantly better performance in imbalanced spliting settings with large first task. These methods

* Corresponding Author

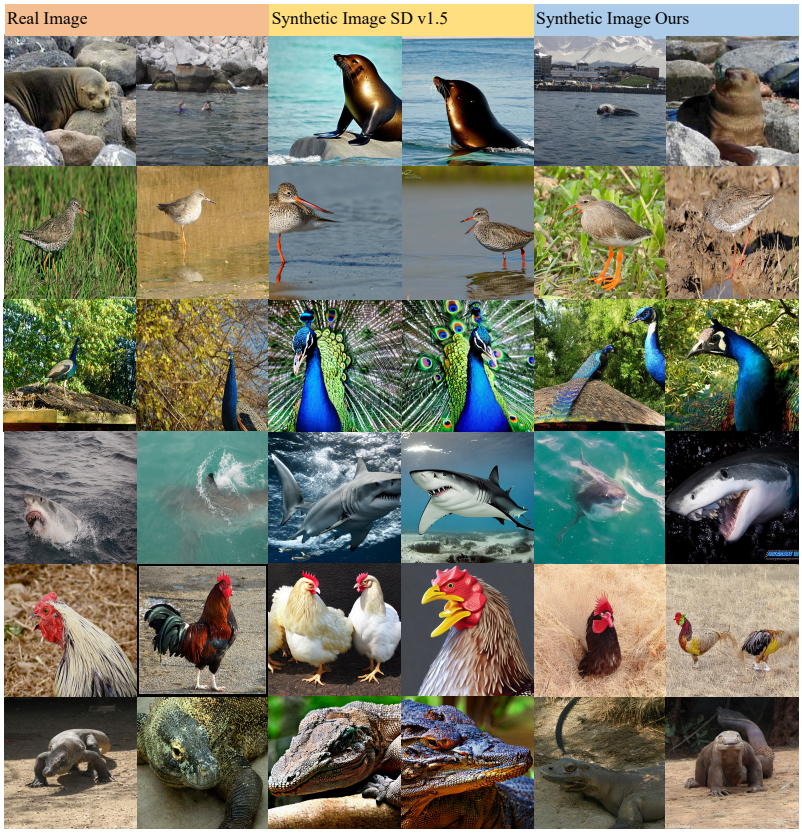

Fig. A1: Visualization of Synthetic Images and Real Images. Synthetic images with full details generated in our method exhibit a closer distribution to real images.

Table A1: Evaluation on ImageNet100 with large first tasks. When $N = 6, 11$, the first task consists of 50 classes, and the remaining 50 classes are split equally into additional 5,10 tasks. When $N = 21$, the first task consists of 40 classes, and the remaining 60 classes are split equally into additional 20 tasks.

Approach	$N = 6$	$N = 11$	$N = 21$
	Acc_{avg}	Acc_{avg}	Acc_{avg}
ABD (ICCV 2021)	65.89	61.76	50.53
PASS(CVPR 2021)	64.37	61.87	51.41
IL2A (NeurIPS 2021)	66.46	61.98	50.81
R-DFCIL (ECCV 2022)	67.13	62.37	53.74
SSRE (CVPR 2022)	67.63	66.57	57.21
DDGR (ICML 2023)	63.54	65.18	66.39
FeTril (WACV 2023)	72.13	71.07	67.09
SEED (ICLR 2024)	**75.31**	70.17	62.90
Model(Ours)	74.39	**72.94**	**72.57**

benefit from a large first task to build strong feature extractors that contribute greatly to their learning in remaining incremental tasks. Our method does not require a robust feature extractor previously trained on large initial tasks and still surpasses the best method (FeTril) by 1.87 and 5.48 percent for $N = 11$ and 21, respectively. SEED presents better average incremental accuracy for $N = 6$, which is just slightly higher (0.92%) than ours. Its performance for $N = 11$ and 21 drops significantly compared with ours.

C Performace on Synthetic and Combined Test Data

Table A2: Evaluation on ImageNet100 test data from different domains with protocol that equally split 100 classes into N tasks.

Test Data Domain	$N = 5$		$N = 10$		$N = 20$	
	Acc_{avg}	Acc_L	Acc_{avg}	Acc_L	Acc_{avg}	Acc_L
Real	74.85	67.26	73.87	67.02	72.51	68.68
Synthetic	72.67	71.83	74.11	72.31	73.54	71.80
Synthetic&Real	73.67	69.55	73.99	69.67	73.03	70.12

The results in the main paper demonstrate the performance on real test data. To further demonstrate the multi-domain adaptation ability of our method, we further present our model's performance on synthetic test data or combined test data in Tab. A2. We finetune a diffusion model with MDM technique using

all testing data from ImageNet100 and generate an equal amount of synthetic test data as real test data. For combined test data, we combine both real and synthetic test data to form this test dataset.

From Tab. A2, our method performs well on not only real test data, but also synthetic or combined test data with similar average accuracy, demonstrating that our method is multi-domain adaptive in exemplar-free CIL settings.

D Detailed Derivation for Equation (1)

Previous training data synthesis work (Real Fake [5]) proposes to minimize an additional Maximum Mean Discrepancy (MMD) to achieve real and generated data distribution matching in fine-tuning diffusion models. The MMD between the distributions of p (real data $\in \mathcal{R}$) and q (proposed generated data $\in \mathcal{S}$) is represented as the supremum of the difference of expectations with constraints from the norm of the function $\psi_\vartheta \in \mathcal{F}$ in the Reproducing Kernel Hilbert Space (RKHS).

$$\mathrm{MMD}[\mathcal{F}, p, q] = \sup_{\|\psi_\vartheta\|_{\mathcal{H}} \leq 1} \left(\mathbb{E}_q\left[\psi_\vartheta(\mathcal{R})\right] - \mathbb{E}_p\left[\psi_\vartheta(\mathcal{S})\right]\right). \tag{1}$$

Different from Real Fake [5], to benefit exemplar-free CIL settings, our objective is to minimize not only the MMD between synthetic $\mathcal{D}_i^{\mathrm{syn}}$ and real data $\mathcal{D}_i^{\mathrm{real}}$ in each task, but also the synthetic images $\mathcal{D}_{0:i+1}^{\mathrm{syn}}$ among all incremental tasks.

A naive approach is to directly minimize MMD between $\mathcal{D}_i^{\mathrm{real}} + \mathcal{D}_{0:i}^{\mathrm{syn}}$ and synthetic data of current task $\mathcal{D}_i^{\mathrm{syn}}$. However, this approach incorporates large amounts of synthetic data from previous tasks, which makes the computations more complicated and inevitably affects the diffusion model's qualitative generation performance on current task classes.

Thus we incorporate random sampling (Z) of a small fixed amount of previous synthetic data and alleviate them only as auxiliary constraints when matching current task real and synthetic data distributions, as shown below,

$$\mathrm{MMD}[\mathcal{F}, p, q] = \sup_{\|\psi_\vartheta\|_{\mathcal{H}} \leq 1} \left(\mathbb{E}_q\left[\psi_\vartheta(\mathcal{D}_i^{\mathrm{real}} + Z(\mathcal{D}_{0:i}^{\mathrm{syn}}))\right] - \mathbb{E}_p\left[\psi_\vartheta(\mathcal{D}_i^{\mathrm{syn}})\right]\right) \tag{2}$$

We can then expand utilizing the inner product in RKHS,

$$\mathrm{MMD}[\mathcal{F}, p, q] = \sup_{\|\psi_\vartheta\|_{\mathcal{H}} \leq 1} \left\langle \mu_{\mathcal{D}_i^{\mathrm{real}} + Z(\mathcal{D}_{0:i}^{\mathrm{syn}})} - \mu_{\mathcal{D}_i^{\mathrm{syn}}}, \psi_\vartheta \right\rangle_{\mathcal{H}} \tag{3}$$

Our method alleviates a Stable Diffusion V1.5 model, therefore the function ψ is actually Variational Autoencoder (VAE) to obtain latent space embeddings. When training a latent diffusion model, p is actually the initial latent noise embedding ϵ, and q is the predicted latent noise embedding by a noise predictor ϵ_θ from noised version x'_t of initial latent embedding of the original image. Therefore

we can reform as:

$$\mathrm{MMD} = \sup_{\|vae\|_{\mathcal{H}} \leq 1} \left\langle \frac{1}{|\mathcal{D}_i^{\mathrm{real}} + Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \sum_{j=1}^{|\mathcal{D}_i^{\mathrm{real}}|+|Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \left(\epsilon - \epsilon_\theta(x_t', t)\right), vae \right\rangle_{\mathcal{H}} \tag{4}$$

Influenced by previous works [2, 5], we believe that squaring the MMD in our exemplare-free CIL settings will obtain a more suitable metric for deep learning optimization, as shown below,

$$\mathrm{MMD}^2 = \left[\sup_{\|vae\|_{\mathcal{H}} \leq 1} \left\langle \frac{1}{|\mathcal{D}_i^{\mathrm{real}} + Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \sum_{j=1}^{|\mathcal{D}_i^{\mathrm{real}}|+|Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \left(\epsilon - \epsilon_\theta(x_t', t)\right), vae \right\rangle_{\mathcal{H}} \right]^2 \tag{5}$$

$$\mathrm{MMD}^2 = || \frac{1}{|\mathcal{D}_i^{\mathrm{real}} + Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \sum_{j=1}^{|\mathcal{D}_i^{\mathrm{real}}|+|Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \left(\epsilon - \epsilon_\theta(x_t', t)\right) ||_{\mathcal{H}}^2 \tag{6}$$

From the above, we form our loss function for MDM. In addition, we incorporate the original diffusion loss on real and synthetic data of current tasks as an upper bound constraint for our loss function as below. And with we only sample a small amount of synthetic data from all previous learning phases instead of using all of them, the qualitative generation performance of the finetuned model can be preserved for the current task classes by these two components.

$$\begin{aligned} \mathcal{L}_{MDM} &= || \frac{1}{|\mathcal{D}_i^{\mathrm{real}}| + |Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \sum_{j=1}^{|\mathcal{D}_i^{\mathrm{real}}|+|Z(\mathcal{D}_{0:i}^{\mathrm{syn}})|} \left(\epsilon - \epsilon_\theta(x_t', t)\right) ||_{\mathcal{H}}^2 \\ &\leq \frac{1}{|\mathcal{D}_i^{\mathrm{real}}|} \sum_{j=1}^{|\mathcal{D}_i^{\mathrm{real}}|} ||\epsilon - \epsilon_\theta(x_t, t)||_{\mathcal{H}}^2 = \mathcal{L}_{diff}. \end{aligned} \tag{7}$$

E Potential Improvements for Our Method

Table A3: Average training & sampling time in hours for generative models in equal split incremental learning task with $N = 5, 10, 20$ on ImageNet100.

Approach	$N = 5$	$N = 10$	$N = 20$
	Average Time	Average Time	Average Time
ABD	1.4	1.5	1.5
R-DFCIL	1.4	1.5	1.5
DDGR	6.0	5.9	5.7
Model (Ours)	5.1	4.2	3.3

Training Time. Our method takes a long training time because of the computations to finetune each MDM diffusion model in each incremental task. We show the average time cost to train and sample from the generative model in each equal split task with $N = 5, 10, 20$ on the ImageNet100 dataset in Tab. A3. Diffusion-based methods (DDGR and ours) generally take longer time than other methods. Exploring ways to finetune diffusion models more efficiently is very important to reduce the overall training time for our exemplar-free CIL method.

Image with Low Resolutions. Even though our method achieves SOTA performance on low-resolution datasets (*e.g.* Cifar100), we believe our method can still be improved. Stable Diffusion V1.5 adopted in our method is designed to produce 512×512 resolutions images with 64×64 latent space input. Cifar100 only has images with the resolution of 32×32, which is even smaller than the latent space output from the VAE encoder in Stable Diffusion V1.5. This inevitably affects the output quality and thus affects the model's learning in each incremental task. We believe that modifying Stable Diffusion V1.5's architecture (*e.g.* remove the VAE module) to obtain better-quality synthetic low-resolution data can further enhance our learning performance in each incremental learning phase.

References

1. Gao, Q., Zhao, C., Ghanem, B., Zhang, J.: R-dfcil: Relation-guided representation learning for data-free class incremental learning. In: European Conference on Computer Vision. pp. 423–439. Springer (2022)
2. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
3. Petit, G., Popescu, A., Schindler, H., Picard, D., Delezoide, B.: Fetril: Feature translation for exemplar-free class-incremental learning. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 3911–3920 (January 2023)
4. Rypeść, G., Cygert, S., Khan, V., Trzcinski, T., Zieliński, B.M., Twardowski, B.: Divide and not forget: Ensemble of selectively trained experts in continual learning. In: The Twelfth International Conference on Learning Representations (2024)
5. Yuan, J., Zhang, J., Sun, S., Torr, P., Zhao, B.: Real-fake: Effective training data synthesis through distribution matching. In: The Twelfth International Conference on Learning Representations (2024)
6. Zhu, F., Cheng, Z., Zhang, X.Y., Liu, C.l.: Class-incremental learning via dual augmentation. Advances in Neural Information Processing Systems **34**, 14306–14318 (2021)
7. Zhu, F., Zhang, X.Y., Wang, C., Yin, F., Liu, C.L.: Prototype augmentation and self-supervision for incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5871–5880 (June 2021)
8. Zhu, K., Zhai, W., Cao, Y., Luo, J., Zha, Z.J.: Self-sustaining representation expansion for non-exemplar class-incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9296–9305 (2022)