Class-Incremental Learning using Diffusion Model for Distillation and Replay

Quentin Jodelet[1,2], Xin Liu[2], Yin Jun Phua[1], Tsuyoshi Murata[1,2]

[1] Department of Computer Science, Tokyo Institute of Technology, Japan

[2] Artificial Intelligence Research Center, AIST, Japan

jodelet@net.c.titech.ac.jp, xin.liu@aist.go.jp, phua@c.titech.ac.jp, murata@c.titech.ac.jp

Abstract

Class-incremental learning aims to learn new classes in an incremental fashion without forgetting the previously learned ones. Several research works have shown how additional data can be used by incremental models to help mitigate catastrophic forgetting. In this work, following the recent breakthrough in text-to-image generative models and their wide distribution, we propose the use of a pretrained Stable Diffusion model as a source of additional data for class-incremental learning. Compared to competitive methods that rely on external, often unlabeled, datasets of real images, our approach can generate synthetic samples belonging to the same classes as the previously encountered images. This allows us to use those additional data samples not only in the distillation loss but also for replay in the classification loss. Experiments on the competitive benchmarks CIFAR100, ImageNet-Subset, and ImageNet demonstrate how this new approach can be used to further improve the performance of state-of-the-art methods for class-incremental learning on large scale datasets.

1. Introduction

In a class-incremental learning (CIL) scenario, deep neural network models are not trained offline on a fixed pre-collected dataset but sequentially on new incoming data. The model has to be incrementally updated using only a limited number of new classes at a time, with the objective of learning a unified classifier among all seen classes. This paradigm, similar to the human learning process, is challenging as models are suffering from catastrophic forgetting [21, 23]: learning new data or classes will result in severe degradation of the performance on the past ones.

Multiple researches have highlighted the importance of rehearsal for continual and incremental learning: by storing samples from each previously encountered class in a memory, models are able to replay past data while learning new ones in order to mitigate catastrophic forgetting. However, the size of this memory is often limited to a few

samples per class due to either storage limitation or privacy concerns. Several authors [16, 17, 19, 37] have proposed to leverage additional external training data as a means to work around this constraint. Those additional training data are sampled from large curated datasets of real images, such as ImageNet, and belong to classes different from the ones encountered by the incremental learner. They are mostly used in an unsupervised manner, for distillating knowledge from the past model to the new one.

In this work, we propose to use the pretrained state-of-the-art generative model Stable Diffusion [25] as a source of complementary data for class-incremental learning. Compared to approaches that rely on additional external real datasets, our method generates synthetic samples belonging to the same classes as the ones previously encountered by the model. This fundamental difference allows us to make use of those samples not only for the knowledge distillation loss but also for replay in the classification loss whereas concurrent works only focus on the distillation loss. Ablation study highlights the improvement resulting from using the additional data for both losses. To the best of our knowledge, this is the first attempt to use large pretrained text-to-image generative models to improve general class-incremental learning methods on large scale datasets.

2. Related work

Class-incremental Learning (CIL): Methods for class-incremental learning [38] usually rely on knowledge distillation and a small replay memory containing past samples combined with a bias mitigation method. Knowledge distillation is used to preserve previous knowledge, it may be logit distillation [18, 24, 34], feature distillation [5, 10] or relational distillation [30]. Using a memory for replaying previously learned exemplars is a simple yet effective method to recall knowledge of old classes. However, the limited size of the memory induces a bias toward the new classes that strongly degrade the performance of the model if not mitigated. To this end, several methods have been proposed such as Nearest-Mean-of-Exemplars (NME) classifier [24], cosine classifier [10], bias correc-

Figure 1. Samples from $\mathcal{S}$ the set of additional synthetic images generated by Stable Diffusion for class-incremental learning on CIFAR100; images are generated with size 512x512 and are being resized to 32x32 before being fed to the model. Images are displayed before resizing to 32x32 for better appreciation. From left to right, top to bottom: Apple, Aquarium fish, Baby, Bear, Beaver, Bed, Bee, Bicycle, Bottle, Bridge, Camel, Clock, Cloud, Crab, Crocodile, Fox, Hamster, Lawn-mower, Maple tree, Orchids.

tion layer [34], finetuning on a balanced subset [3] or specific losses [1, 13, 14]. To overcome the limitation induced by the limited size of the memory, several works have also proposed to use model inversion [29, 36] or to train a generative model in parallel to generate samples from past classes [6, 28, 33]. These methods rely on the quality of the generated data, which poses a challenge for large scale datasets and is impacted by catastrophic forgetting when trained sequentially.

Additional data for CIL: Various approaches have studied the use of additional external data, either as a direct substitute in case of the absence of memory [17, 37] or as a complementary resource during training [16, 19]. Global distillation (GD) [17] proposed a confidence-based sampling strategy to sample additional external data and used them for a specifically-designed distillation loss. Deep Model Consolidation (DMC) [37] consists in training one distinct model for each training step and use external data to combine them into a unified model. Lechat *et al.* [16] proposed a pseudo-labeling approach to better exploit the additional external unlabeled data for representation learning. Liu *et al.* [19] observed that using samples from new classes in the knowledge distillation loss negatively impacts the training of incremental models and proposed to replace them using placebos of old classes sampled from an external dataset using an online learning method. Those different approaches rely solely on real images, most of the time unlabeled, by sampling their additional data from large curated datasets: primarily ImageNet [26] or TinyImages dataset [31].

Learning from synthetic data: In one of the pioneer works on learning from synthetic images, Besnier *et al.* [2] trained a classifier for 10 classes of ImageNet using samples generated by a class-conditional GAN trained on ImageNet. Jahanian *et al.* [12] used a GAN to generate multiple views for contrastive learning methods. Recently, following the recent breakthrough in text-to-image generative models, Sariyildiz *et al.* [27] proposed the use of those general-purpose text-conditioned pretrained generative models for synthesizing datasets as a direct replacement of real image datasets for the training of large scale image-level classification models. While these models can not reach the same accuracy as the models trained using real images when tested on real test datasets, they exhibit other advantages such as strong generalization capability. Concurrently, He *et al.* [8] also proposed to use large synthetic datasets for improving pretrained models on zero-shot and few-shot learning, and transfer learning.

In this work, we propose to study the use of widely available pretrained generative models such as Stable Diffusion [25] as a source of additional labeled samples combinable with general class-incremental learning methods applied on large scale datasets.

3. Proposed method

The class-incremental training procedure is divided into $T + 1$ steps: the base step, often referred to as step 0 or initial step, followed by T incremental steps. Each step consists of a training dataset $\mathcal{D}_t$ containing images belonging to the set of new and previously unseen classes $\mathcal{Y}_t$; $\forall i, j \in \{0, 1, ..., T\}, \mathcal{Y}_i \cap \mathcal{Y}_j = \emptyset$ for $i \neq j$. During each step, the model θ_t is trained on $\mathcal{D}_t \cup \mathcal{M}$ where $\mathcal{M}$ is the replay memory containing few samples from previously encountered classes. After each step, the model θ_t is evaluated on the test set of all the classes learned so far without having access to any step or task descriptor. Additionally, during every step, the model also has access to an external source of data $\mathcal{S}$ that may or may not change during the training. This external source of additional data $\mathcal{S}$ can either be an online, and potentially infinite stream of data that the model can fetch when needed or a fixed set of data accessible in an offline manner.

3.1. Synthetic images

Following the recent in-depth study of Sariyildiz *et al.* [27], we propose to use a pretrained Stable Diffusion [25]
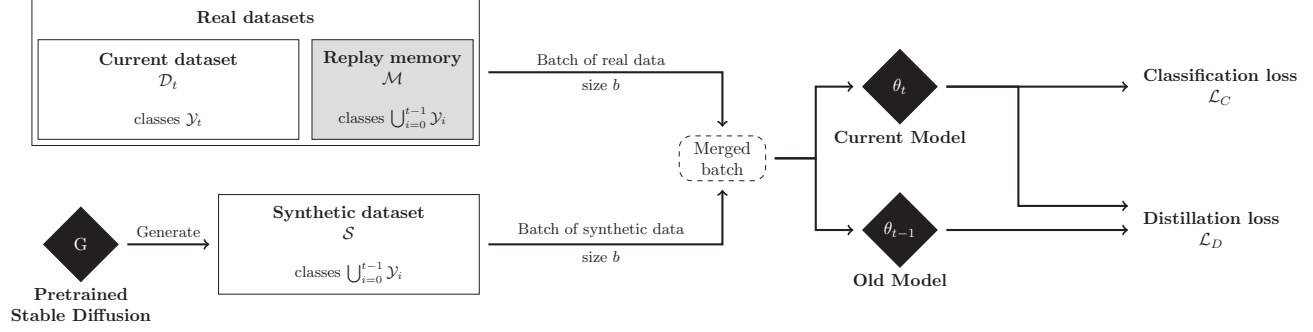
Figure 2. Diagram representing our proposed method SDDR for leveraging additional external synthetic labeled samples for class-incremental learning. More details in Algorithm 1.

as the source of additional data $\mathcal{S}$ for incremental learning.

Prompting: To generate a sample for $\mathcal{S}$, the generative model Stable Diffusion requires a textual prompt. As proposed in [27], to generate a synthetic image for the class c we use the prompt "$c,\ d_c$" where "c" is the textual name of the class and "d_c" its description. Although using only "c" as a prompt may be enough in most cases, this may lead to semantic errors with homographs. For example, in the CIFAR100 dataset using only "apple" as a prompt may generate images of products or shops from the homonym brand instead of the fruit. Adding the definition of the class to the prompt helps solve this issue. In order to produce these prompts automatically without any further engineering, WordNet [22] is used to provide the lemmas of the synset as the class name "c" and the definition of the synset as the description "d_c". For datasets such as ImageNet, the association of each class with a synset of Word-Net is included in the dataset itself. For other datasets such as CIFAR100, the association with WordNet can be done semi-automatically at a negligible cost. Figure 1 shows synthetic samples for CIFAR100 before being resized to 32x32. In the cases where the mapping between the target dataset and WordNet can not be done, it may be necessary to either manually design and tune the prompt for each class or to rely on a different automation approach such as using image-to-text model for example.

Guidance scale: The classifier-free guidance [9] controls the trade-off between the quality of the generated synthetic samples and their diversity in diffusion models. Following Sariyildiz *et al*. [27], we also used a guidance scale of 2.0 compared to the default 7.5 to increase the diversity of our source of additional data at the cost of a slight decrease of the quality of each sample.

3.2. SDDR for Class-incremental learning

Using the previously described generative model, we propose a new method for class-incremental learning:

Algorithm 1: SDDR for class-incremental learning

Input: Data-flow $\{\mathcal{D}_i\}_{i=0}^T$ and pretrained generative model G.
Ouput: Models $\{\theta_i\}_{i=0}^T$, replay memory $\mathcal{M}$, and synthetic dataset $\mathcal{S}$.

for $i \in \{0, 1, ..., T\}$ **do**
    for $e \in \{0, 1, ..., E\}$ **do**
        while $(X, Y) \sim \mathcal{D}_i \cup \mathcal{M}$ **do**
            if $i > 0$ **then**
                `// If not base step`
                $(X_S, Y_S) \sim \mathcal{S}$
                $(X, Y) \leftarrow (X \cup X_S, Y \cup Y_S)$
            end
            Update model θ_i using (X, Y)
        end
    end
    $\mathcal{M} \leftarrow \text{UpdateMemory}(\mathcal{M}, \mathcal{D}_i)$
    $\mathcal{S} \leftarrow \text{UpdateSynthetic}(\mathcal{S}, \text{G}, \mathcal{Y}_i)$
end

Stable **D**iffusion for **D**istillation and **R**eplay (**SDDR**), which jointly uses the additional synthetic data $\mathcal{S}$ with the real images from the replay memory $\mathcal{M}$.

Datasets of real images used by concurrent approaches as an external source of additional data for class-incremental learning do not contain any image belonging to the same classes as the one encountered by the incremental model. This leads these methods to mainly use this external source of data in an unsupervised manner for knowledge distillation only. By leveraging the pretrained text-to-image generative model, our method is able to generate an additional dataset $\mathcal{S}$ containing labeled images belonging to the same classes as the one previously learned by the model without the need for costly manual labeling and curating process.

Methods	CIFAR100			ImageNet-Subset			ImageNet	
	T=5	10	25	5	10	25	5	10
iCaRL [24]	57.17	52.57	-	65.04	59.53	-	51.36	46.72
BiC [34]	59.36	54.20	50.00	70.07	64.96	57.73	62.65	58.72
LUCIR [10]	63.42	60.18	-	70.47	68.09	-	64.34	61.28
LUCIR w/ Mnemonics [20]	63.34	62.28	60.96	72.58	71.37	69.74	64.54	63.01
LUCIR w/ DDE [11]	65.27	62.36	-	72.34	70.20	-	67.51	65.77
LUCIR w/ BalancedS-CE [14]	64.83	62.36	58.38	71.18	70.66	65.10	**67.81**	**66.47**
TPCIL [30]	65.34	63.58	-	<u>76.27</u>	74.81	-	64.89	62.88
DER (w/o Pruning) [35]	68.52	67.09	-	-	**78.20**	-	-	-
iCaRL* [24]	57.68	52.44	48.05	65.34	60.74	54.31	53.58	48.65
iCaRL w/ SDDR (ours)	62.01	59.24	56.47	68.64	65.55	62.75	57.15	54.07
LUCIR* [10]	63.37	60.88	57.07	70.75	68.48	63.14	66.69	64.06
LUCIR w/ SDDR (ours)	65.77	63.84	61.73	72.92	71.48	69.48	<u>67.56</u>	<u>66.44</u>
FOSTER* [32]	<u>71.17</u>	68.89	<u>65.07</u>	76.09	75.05	<u>70.96</u>	-	-
FOSTER w/ SDDR (ours)	**72.18**	**70.88**	**68.06**	**77.13**	<u>76.77</u>	**75.50**	-	-

Table 1. Average incremental accuracy (Top-1) on CIFAR100, ImageNet-Subset, and ImageNet with a base step containing half of the classes followed by 5, 10, and 25 incremental steps, using a growing memory of 20 exemplars per class. Results for iCaRL and LUCIR are reported from [10]; results for Mnemonics and BiC are reported from [20]. Results for DDE, BalancedS-CE, TPCIL, and DER are reported from their respective paper. Results marked with "*" correspond to our own experiments. Results on CIFAR100 and ImageNet-Subset are averaged over 3 random runs. Results on ImageNet are reported as a single run. Best result is marked in bold and second best is underlined.

This allows us to use the additional dataset for distillation but also for replay in the classification loss. The ablation study in Section 4.3 shows the advantages of our approach.

Our approach is designed as a complementary method that can be seamlessly combined with other standard methods for class-incremental learning. Algorithm 1 describes the training procedure of our approach SDDR when combined with a class-incremental learning model. Each step of the class-incremental learning procedure is divided into two distinct phases. During the first phase, the model θ_t is trained following the method it is combined with, using a mixed batch of data. As illustrated in Figure 2, each batch used to train the model is composed of half of synthetic images sampled from $\mathcal{S}$ and the other half of real data sampled from the memory and the current dataset $\mathcal{D}_t$. Then in the second phase, before moving to the next incremental step, the synthetic dataset $\mathcal{S}$ is extended by using Stable Diffusion to generate n synthetic samples for each new class encountered in the current dataset $\mathcal{D}_t$, following the method detailed in Section 3.1. Therefore, at the end of each incremental step, the additional dataset $\mathcal{S}$ would contain nN_t samples where $N_t = |\bigcup_{i=0}^{t} \mathcal{Y}_i|$ is the number of classes encountered up to the step t, included.

In our work, we considered this additional dataset of synthetic samples $\mathcal{S}$ as an offline dataset stored on the device itself. Nonetheless, if the storage is a constraint for the incremental learner, the additional dataset $\mathcal{S}$ can be used in an online way as an infinite stream of data, similarly to [19], by either using the generative model locally or by relying on a cloud-based model. Compared to concurrent works

relying on real datasets as a source of additional data, our method offers more flexibility depending on the constraint of the problem: it can be used with or without limited storage, computational budget, and capacity to communicate with external services.

The quality of the synthetic images, their fidelity to the target classes, and the variety of concepts available in the additional external dataset $\mathcal{S}$ are intrinsically limited by the pretrained generative text-to-image model used. Using a different and more specialized generative model or finetuning it on the specific target training dataset may further improve the performance of the incremental learning model it is combined with.

A naive and straightforward application of a pretrained text-to-image generative model in the context of class-incremental learning would be to use the synthetic datasets $\mathcal{S}$ as a complete replacement of the replay memory for exemplar-free class-incremental learning. However, as discussed in the ablation study in Section 4.3, this approach does not achieve competitive results.

4. Experiments

4.1. Experimental setups

Datasets: Experiments are conducted on three datasets: CIFAR100 [15], ImageNet-Subset and ImageNet [26]. CIFAR100 is composed of 60,000 32x32 RGB images from 100 classes with 500 training and 100 testing samples per class. ImageNet (ILSVRC 2012) is composed of around 1.28 million high-resolution images from 1,000 classes

Methods	CIFAR100			ImageNet-Subset		
	T=5	10	25	5	10	25
FOSTER† [32]	70.62	68.43	63.83	80.21	77.63	69.27
FOSTER w/ PlaceboCIL† [19]	71.97	70.31	67.02	**82.03**	**79.52**	72.79
Improvement in *p.p.*	**+1.35**	+1.88	**+3.19**	**+1.82**	**+1.89**	+3.52
FOSTER* [32]	71.17	68.89	65.07	76.09	75.05	70.96
FOSTER w/ SDDR (ours)	**72.18**	**70.88**	**68.06**	77.13	76.77	**75.50**
Improvement in *p.p.*	+1.01	**+1.99**	+2.99	+1.04	+1.72	**+4.54**

Table 2. Average incremental accuracy (Top-1) on CIFAR100, and ImageNet-Subset with a base step containing half of the classes followed by 5, 10, and 25 incremental steps, using a growing memory of 20 exemplars per class. Results marked with "*" correspond to our own experiments and results marked with "†" are reported from [19]. Results averaged over 3 random runs.

with about 1,300 training and 50 testing sample per class. ImageNet-Subset is a subset of ImageNet containing only 100 classes. Experiments on an alternative version of ImageNet-Subset can be found in the Supplementary materials. Following the experimental setting initially proposed by Hou *et al.* [10], an initial base step containing half of the classes is followed by T (5, 10, or 25) incremental steps containing the remaining classes evenly divided. The class order is defined by NumPy using the random seed 1993. The training images are normalized, randomly horizontally flipped, and cropped, and no more augmentation is applied. For FOSTER [32], following its authors, AutoAugment [4] is also used for augmentation.

Comparison methods: We combine our proposed method SDDR with the recent state-of-the-art approach FOSTER [32] and with two fundamental baselines for class-incremental learning: iCaRL [24] and LUCIR [10]. We selected these two methods because they are representative of the principal approaches for class-incremental learning: using different distillation losses (logits distillation and feature distillation) and difference bias mitigation methods (Nearest-Mean Exemplar classifier and cosine normalized classifier). For comparison, we also include the recent approach PlaceboCIL [19] which leverages external unlabeled datasets of real high-resolution images to improve the distillation loss of class-incremental learning methods.

Implementation details: Following the standard setting [10, 24], a 32-layer ResNet [7] is used for CIFAR100 and a 18-layer ResNet for ImageNet and ImageNet-Subset. Every considered method uses a growing memory containing 20 exemplars per class during the training procedure. When combining SDDR with an existing baseline for class-incremental learning, the same hyperparameters as the original method are used. The only difference is that the batch size during incremental steps is effectively doubled by appending synthetic images generated by the generative model to the original batch of real images from the union of the new dataset and the replay memory. Some methods for class-incremental learning use other losses in addition to the classification and distillation losses. Using synthetic

data for these other losses may improve or degrade the performance of the method on a case-by-case basis. Specifically, in our experiments on LUCIR [10], we decide not to use the synthetic data for the margin ranking loss as preliminary experiments suggested that it slightly decreases the performance of the method. Likewise, in our experiments on FOSTER [32], the synthetic data are used for the feature enhancement loss and the distillation loss during feature boosting and for the distillation loss during feature compression. We do not use the synthetic data for the logits aligned classification loss as it results in a significant decrease of the accuracy due to a bias of the classifier. For fair comparison with the other baselines, we report the accuracy of FOSTER after feature compression. For comparison of dual branch FOSTER before feature compression, see Supplementary materials. The generative model we use is Stable Diffusion version 1.4 [25] with a guidance scale of 2.0, 50 steps and the prompt "c, d_c" [27]. Synthetic images have been generated using fixed seeds before the incremental training and every method is using the same datasets of synthetic images. The same seeds have been used for generating all classes, resulting in closely related classes having some similar images (same background, subject's pose and position) where only the subject of the image changes. For CIFAR100, the synthetic dataset contains 500 32x32 images per class, and for ImageNet and ImageNet-Subset 1300 512x512 images per class. For every dataset, synthetic images are generated with size 512x512 and are only resized to 32x32 for CIFAR100 prior to the experiments.

Performance Measure: Models are evaluated and compared using the Average Incremental Accuracy defined by Rebuffi *et al.* [24]. It is the average of the Top-1 accuracy of the model on the test data of all the classes seen so far, at the end of each training step including the initial base step.

4.2. Comparison with baselines

In Table 1, the average incremental accuracy (Top-1) of our method and the different baselines are reported for CIFAR100, ImageNet-Subset, and ImageNet with 5, 10, and 25 incremental steps using a replay memory of 20 real ex-

| Methods | CIFAR100 | | | | | | | | | | | |
| | 5 exemplars/class | | | 10 exemplars/class | | | 20 exemplars/class | | | 50 exemplars/class | | |
	T=5	10	25	5	10	25	5	10	25	5	10	25
iCaRL* [24]	43.40	39.30	29.59	52.30	46.79	38.78	57.68	52.44	48.05	62.09	58.42	55.36
iCaRL w/ PlaceboCIL [19]	51.55	-	-	59.11	-	-	61.24	-	-	-	-	-
iCaRL w/ SDDR (ours)	55.36	50.80	48.15	58.53	55.70	52.51	62.01	59.24	56.47	65.53	63.92	62.18
Improvement in *p.p.*	+11.96	+11.50	+18.56	+6.23	+8.91	+13.73	+4.33	+6.80	+8.42	+3.44	+5.50	+6.82
LUCIR* [10]	53.14	52.79	44.67	60.77	57.55	50.77	63.37	60.88	57.07	65.63	62.39	60.80
LUCIR w/ PlaceboCIL [19]	62.74	-	-	64.79	-	-	65.28	-	-	-	-	-
LUCIR w/ SDDR (ours)	62.90	59.81	56.32	64.47	62.16	59.44	65.77	63.84	61.73	67.21	65.81	64.48
Improvement in *p.p.*	+9.76	+7.02	+11.65	+3.70	+4.61	+8.67	+2.40	+2.96	+4.66	+1.58	+3.42	+3.68

Table 3. Average incremental accuracy (Top-1) on CIFAR100 with a base step containing half of the classes followed by 5, 10, and 25 incremental steps depending on the number of exemplars saved in memory for each class. Results marked with "*" correspond to our own experiments. Results for PlaceboCIL reported from [19]. Improvement reported comparatively to the baseline method. Results averaged over 3 random runs.

emplars per class for every method. Due to significant differences in the reproduced accuracy for FOSTER [32] between our work and Liu *et al*. [19], we decided to make a more detailed comparison with PlaceboCIL in Table 2.

For FOSTER, LUCIR, and iCaRL, using our proposed approach SDDR leads to a significant improvement of the average incremental accuracy: from 1.01 percentage points (*p.p.*) to 4.54*p.p.* for FOSTER, from 0.87*p.p.* to 6.34*p.p.* for LUCIR and from 3.3*p.p.* to 8.44*p.p.* for iCaRL, depending on the dataset and the setting. This improvement is especially important in more challenging settings with numerous incremental steps or when the replay memory is small as discussed in Section 4.3. When looking more closely at the learned model, we can see that our approach improves the capacity of the model to preserve past knowledge without penalizing its plasticity. This is especially true in the most challenging settings with a small replay memory: for example on CIFAR100 with 10 incremental steps and 5 exemplars per class in the memory, by combining LU-CIR with SDDR, the final accuracy on the 50 bases classes increases from 34.15% to 42.06% while the accuracy on the 50 remaining classes also increases from 46.08% to 52.23%. By combining our proposed method SDDR with FOSTER, we achieve new state-of-the-art performance on several datasets and settings.

Our proposed method achieves competitive performances compared to other considered approaches Mnemonics [20], DDE [11] or BalancedS-CE [14] and could be even combined with them to further improve their performance. Compared to PlaceboCIL [19], our method achieves similar or higher performance while using a significantly smaller dataset of additional external data: 50,000 synthetic images compared to the 1.28 million of real images from ImageNet for the experiments on CIFAR100 and 130,000 synthetic images compared to about 1.1 million of real images sampled without overlapping from ImageNet for the experiments on ImageNet-Subset.

| Methods | CIFAR100 | | |
	T=5	10	25
LUCIR [10]	63.37	60.88	57.07
w/ SD Distillation	64.62	63.12	**62.15**
w/ SD Distillation w/o new	62.66	61.93	61.48
w/ SD Replay	64.95	62.44	59.59
w/ SDDR	**65.77**	**63.84**	61.73

Table 4. Average incremental accuracy (Top-1) on CIFAR100 with a base step containing half of the classes followed by 5, 10, and 25 incremental steps, using a growing memory of 20 exemplars per class. Results averaged over 3 random runs.

| Methods | CIFAR100 | | |
	T=5	10	25
LUCIR [10]	63.37	60.88	57.07
w/ SDDR $n=50$	64.84	62.71	60.64
w/ SDDR $n=500$	65.77	63.84	61.73
w/ SDDR $n=1000$	**66.41**	64.91	62.73
w/ SDDR $n=2000$	66.22	**65.03**	**62.79**

Table 5. Average incremental accuracy (Top-1) on CIFAR100 with a base step containing half of the classes followed by 5, 10, and 25 incremental steps depending on n, the number of synthetic samples generated for each class. Using a growing memory of 20 exemplars per class. Results averaged over 3 random runs.

4.3. Ablation Study

Components: By leveraging synthetic images supposedly belonging to the same classes as the training dataset, it is possible to use them for both replay and distillation. Table 4 compares the performance when using the synthetic images only for distillation (w/ SD Distillation), only for replay (w/ SD Replay), and for both combined (w/ SDDR). Distillation and Replay using synthetic data appear to be complementary, Replay achieving higher performances for a short number of incremental steps T and Distillation for a higher number of incremental steps. Combining both in

SDDR achieves an overall improvement over each used individually.

Size of the synthetic dataset: Table 5 shows the impact of the number of synthetic images n generated for each past class used during training. With only 50 synthetic images per class, our method can improve the average incremental accuracy by up to $3.57p.p.$ compared to the standard LUCIR. This can be further increased to $5.72p.p.$ by expanding the size of the synthetic dataset. However, it appears that beyond a certain limit, further increasing the size of the synthetic dataset does not yield any significant improvement. This supports our approach of not using Stable Diffusion in an online manner to generate each training batch if the storage is not a constraint of the system.

Size of the replay memory: Table 3 measures the improvement from our method depending on the number of incremental steps and the size of the replay memory containing exemplars of past classes. While the improvement compared to the baseline method is more significant in the case of a small memory size, our method also significantly improves the average incremental accuracy of both LUCIR and iCaRL when combined with a larger replay memory. Likewise, the improvement is more notable for more difficult settings containing numerous incremental steps. For example, when trained for 25 incremental steps with 5 samples per class in the replay memory, SDDR almost doubles the final overall accuracy for iCaRL, increasing it from 20.32% to 40.39% and augments the average incremental accuracy by $18.56p.p.$. Those results highlight that our method can boost the performance of the baseline method it is combined with independently of the setting. Additional results for CIFAR100 with 5 incremental steps can be found in the Supplementary materials.

Use of new data: In their recent work, Liu *et al*. [19] emphasized that using data from new classes for the distillation loss was detrimental and proposed to only use data of past classes from the memory in addition to the unlabeled samples from the additional external dataset. Following their study, we also report in Table 4 the results of our method while using the distillation loss only on the synthetic images of past classes and on the exemplar from the memory (w/ SD Distillation w/o new). However, it appears that in our case, this significantly decreases the performance of the model. As further discussed below, we suppose that it may be due to the gap between the distribution of synthetic images and the distribution of real images: while real samples from new images are generally harmful to the distillation loss, they may be useful for transferring general knowledge about real data in our case.

From synthetic to real images: As highlighted by Sariyildiz *et al*. [27], while the synthetic images generated by Stable Diffusion tend to be highly similar to real ones, there remains a gap that limits the generalization of the

| Memory | CIFAR100 | | | |
| | T=5 | | T=10 | |
	Average	Last	Average	Last
Real 20	**63.37**	**53.91**	**60.88**	**51.42**
Synthetic 20	52.38	33.68	42.79	21.83
Synthetic 100	54.54	36.56	48.56	27.93
Synthetic 500	55.77	38.78	52.19	32.58

Table 6. Performances of standard LUCIR [10] on CIFAR100 with a base step containing half of the classes followed by 5 and 10 incremental steps depending on the number of exemplars per class saved in the memory and whether the exemplars are real or synthetic images. "Average" is Average Incremental Accuracy (Top-1) and "Last" is the final overall accuracy of the model after the last incremental step. the Results averaged over 3 random runs.

model and may negatively impact the performance. We first observed this phenomenon when using the synthetic dataset generated by Stable Diffusion as a direct replacement for the replay memory in the exemplar-free class-incremental learning setting. As depicted in Table 6, when using synthetic images as exemplars for the past classes, LUCIR has difficulty generalizing to the real test dataset and achieves significantly lower accuracy compared to using real exemplars in the memory. Moreover, while it performs better than exemplar-based methods used without any replay memory, it is not competitive with approaches specifically designed for this challenging setting. Secondly, the impact of the gap between synthetic and real data may also be observed at the classifier level, especially when using our proposed SDDR method in the most challenging settings where the size of the replay memory is highly limited. Fixing old class embeddings learned using real data only, using an NME classifier of real exemplars only, or finetuning the classifier on a small dataset of real images may help in this situation: for example, on CIFAR100 with 25 incremental steps and 5 exemplars per class in the replay memory, using an NME classifier for LUCIR w/ SDDR would improve the average incremental accuracy by $0.89p.p.$, reaching 57.21% while for standard LUCIR it would decrease it by $0.01p.p.$. Sariyildiz *et al*. [27] addressed the sim-to-real gap by using strong augmentation policy, and multi-crops and experimentally found that it results in a large performance improvement: 51% relative increase of the accuracy compared to standard limited augmentation in their simplest variant of synthetic ImageNet-100. The same approach could be used for our method, especially as strong augmentations and larger models have been shown to be beneficial for incremental learning [16].

5. Conclusion

In this work, we showed that pretrained generative text-to-image model can be seamlessly combined with general

methods for class-incremental learning in order to further improve their performances. Our approach leverages Stable Diffusion to generate labeled synthetic images belonging to the same classes as the ones previously encountered by the model and use those synthetic data for both the distillation loss and the classification loss. Complete experiments show that our approaches significantly increase the average incremental accuracy of state-of-the-art methods for class-incremental learning, especially in settings with highly restricted memory, and achieve comparable or superior results compared to other competitive approaches relying on additional real data samples while offering more flexibility.

While our work focused on a simple and straightforward integration of the synthetic images into the class-incremental learning setting, it would be important in future works to consider how the generative models could be fine-tuned during the training and how the different losses could be modified to better take advantage of the synthetic images and reduce the synthetic-to-real gap. Moreover, we will also explore the different possible uses of those pretrained generative models for class-incremental learning: for example to prepare the model by learning in advance, during the initial step, classes that may be encountered in the future, based on already encountered classes.

Acknowledgments

References

[1] Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 844–853, October 2021.

[2] Victor Besnier, Himalaya Jain, Andrei Bursuc, Matthieu Cord, and Patrick Pérez. This dataset does not exist: training models from generated images. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2020.

[3] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 241–257, Cham, 2018. Springer International Publishing.

[4] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[5] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 86–102. Springer, 2020.

[6] Chen He, Ruiping Wang, Shiguang Shan, and Xilin Chen. Exemplar-supported generative reproduction for class incremental learning. In *BMVC*, page 98, 2018.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016.

[8] 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? *arXiv preprint arXiv:2210.07574*, 2022.

[9] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[10] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[11] Xinting Hu, Kaihua Tang, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. Distilling causal effect of data in class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3957–3966, June 2021.

[12] Ali Jahanian, Xavier Puig, Yonglong Tian, and Phillip Isola. Generative models as a data source for multiview representation learning. In *International Conference on Learning Representations*, 2022.

[13] Quentin Jodelet, Xin Liu, and Tsuyoshi Murata. Balanced softmax cross-entropy for incremental learning. In Igor Farkaš, Paolo Masulli, Sebastian Otte, and Stefan Wermter, editors, *Artificial Neural Networks and Machine Learning – ICANN 2021*, pages 385–396, Cham, 2021. Springer International Publishing.

[14] Quentin Jodelet, Xin Liu, and Tsuyoshi Murata. Balanced softmax cross-entropy for incremental learning with and without memory. *Computer Vision and Image Understanding*, 225:103582, 2022.

[15] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[16] Alexis Lechat, Stéphane Herbin, and Frédéric Jurie. Pseudo-labeling for class incremental learning. In *BMVC 2021: The British Machine Vision Conference*, 2021.

[17] Kibok Lee, Kimin Lee, Jinwoo Shin, and Honglak Lee. Overcoming catastrophic forgetting with unlabeled data in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 312–321, 2019.

[18] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[19] Yaoyao Liu, Yingying Li, Bernt Schiele, and Qianru Sun. Online placebos for class-incremental learning, 2023.

[20] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental

learning without forgetting. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020.

[21] Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pages 109 – 165. Academic Press, 1989.

[22] George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, nov 1995.

[23] Roger Ratcliff. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285, 1990.

[24] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[27] Mert Bülent Sarıyıldız, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8011–8021, June 2023.

[28] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

[29] James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. Always be dreaming: A new approach for data-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9374–9384, 2021.

[30] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. Topology-preserving class-incremental learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 254–270, Cham, 2020. Springer International Publishing.

[31] Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

[32] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *European conference on computer vision*, pages 398–414. Springer, 2022.

[33] Liyuan Wang, Kuo Yang, Chongxuan Li, Lanqing Hong, Zhenguo Li, and Jun Zhu. Ordisco: Effective and efficient usage of incremental unlabeled data for semi-supervised continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5383–5392, 2021.

[34] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019.

[35] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021.

[36] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deep-inversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8715–8724, 2020.

[37] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-incremental learning via deep model consolidation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1131–1140, 2020.

[38] Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Deep class-incremental learning: A survey, 2023.