FROM LOCAL CONDENSATION TO GLOBAL UTILITY: REPLAY-ENHANCED CONTINUAL DATASET DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Dataset distillation typically assumes that the entire training dataset is available simultaneously. In this work, we study *Continual Dataset Distillation* (CDD), a new setting in which class groups arrive sequentially and a synthetic memory must be constructed and maintained without retaining historical real samples. The resulting synthetic memory is evaluated by training fresh models on all observed classes, requiring synthetic subsets distilled at different steps to remain compatible as the label space expands. We identify a key failure mode in CDD, which we term *Local–Global Discrimination Collapse* (LGDC). Synthetic subsets distilled independently within individual class groups may remain effective for local classification, yet lose compatibility after aggregation for global classification. To address this issue, we propose *Replay-Enhanced Continual Dataset Distillation* (ReCDD), which replays historical synthetic samples during the distillation of new classes and refines the historical memory as the label space expands. To provide a consistent competition signal when future classes are unavailable, ReCDD introduces a non-semantic auxiliary reference that is discarded before evaluation. Experiments on four datasets and five representative methods spanning gradient and trajectory matching show that ReCDD improves final global accuracy in most evaluated settings. Further diagnostic analyses show that ReCDD reduces the local–global discrepancy and cross-step errors associated with LGDC.

1 INTRODUCTION

Dataset distillation (DD) aims to compress a large training dataset into a small set of synthetic samples such that training a model on the synthetic set achieves comparable performance to training on the original dataset (Wang et al., 2018). By replacing thousands or millions of real training samples with a compact synthetic dataset, DD provides an appealing paradigm for reducing the cost of data storage, model training, and data transmission. A variety of methods have been developed to improve the quality of distilled datasets. Early approaches optimize synthetic samples by matching the training dynamics of models trained on real and synthetic data, including gradient matching (Zhao et al., 2021; Zhao & Bilen, 2021) and trajectory matching (Cazenavette et al., 2022; Du et al., 2023; Guo et al., 2024). Other approaches have explored alternative objectives, architectures, and optimization strategies to improve the fidelity and efficiency of dataset distillation. Despite these advances, existing DD methods predominantly consider an *offline* setting, where the entire target dataset is available simultaneously during the distillation process. Consequently, the synthetic dataset is optimized with knowledge of the complete label space and the full data distribution.

This assumption, however, does not hold in many practical scenarios where data arrive sequentially. For example, new classes or domains may emerge over time while historical real data become inaccessible due to storage or privacy constraints. A straightforward solution is to independently distill each arriving subset and accumulate the resulting synthetic memories. However, this raises a fundamental question that has received little attention: *can a high-quality synthetic dataset itself be constructed and maintained continually as new data arrive, without retaining historical real samples?* We study this problem as *Continual Dataset Distillation* (CDD).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

A natural baseline for CDD is to independently distill newly arriving data at each step and accumulate the resulting synthetic subsets. Although simple, this strategy exposes a fundamental *local–global mismatch*. Specifically, each distillation step optimizes only the currently available label space. We observe that synthetic samples that are individually effective within their local label spaces can become mutually incompatible after aggregation, resulting in substantial degradation in global classification performance. We refer to this phenomenon as *Local–Global Discrimination Collapse* (LGDC).

To address this issue, we propose *Replay-Enhanced Continual Dataset Distillation* (ReCDD). Instead of independently optimizing only the current synthetic subset, ReCDD replays historical synthetic samples while distilling new classes and refines historical memory as the label space expands. In addition, because future classes are unavailable during early condensation, we introduce a non-semantic auxiliary reference to provide a consistent competition signal during optimization. The auxiliary reference is discarded before evaluation and does not increase the maintained memory.

Across MNIST (LeCun et al., 1998), CIFAR-10, CIFAR-100 (Krizhevsky, 2009), and Tiny ImageNet (Le & Yang, 2015), ReCDD improves the training utility of continually maintained synthetic memories across a range of gradient- and trajectory-matching methods and memory budgets. Further diagnostics on gradient-matching methods show that these improvements are accompanied by a smaller local–global discrepancy and fewer cross-step errors associated with LGDC.

Our main contributions are summarized as follows:

- We formulate *Continual Dataset Distillation* as the sequential maintenance of a synthetic dataset and evaluate its standalone training utility with fresh models over the accumulated label space.
- We identify *Local–Global Discrimination Collapse* (LGDC), where synthetic subsets that remain effective within their local label spaces can lose compatibility after aggregation, and characterize this failure through the local–global gap and cross-step errors.
- We propose *Replay-Enhanced Continual Dataset Distillation* (ReCDD), which replays and refines historical synthetic memory as the label space expands and introduces a non-semantic auxiliary reference to alleviate the lack of future-class competition during early condensation. Experiments across multiple datasets and matching paradigms show improved final and continual synthetic-memory utility.

2 Related Work

2.1 Dataset Distillation

Dataset distillation was initially formulated as bi-level optimization over synthetic samples and model training (Wang et al., 2018). Subsequent methods seek more tractable objectives. KIP formulates distillation through kernel ridge regression (Nguyen et al., 2021); DC and DSA match gradients (Zhao et al., 2021; Zhao & Bilen, 2021); CAFE and DM align intermediate features or class-wise feature distributions (Wang et al., 2022; Zhao & Bilen, 2023); and MTT, FTD, and DATM match training trajectories (Cazenavette et al., 2022; Du et al., 2023; Guo et al., 2024). Complementary work improves synthetic-data parameterization or scalability, including IDC, TESLA, and SRe2L (Kim et al., 2022; Cui et al., 2023; Yin et al., 2023). Despite these differences, most methods assume offline access to the target distribution and label space during condensation.

DD has also been used as compact memory in class-incremental learning. DM can synthesize classes independently, while its DSA baseline condenses only classes arriving at each step (Zhao & Bilen, 2023). IDC likewise synthesizes exemplars using only currently available classes at each stage (Kim et al., 2022). LoDM combines low-rank condensation with GDumb, which trains a fresh model on the current memory for evaluation (Yang et al., 2023; Prabhu et al., 2020). These studies show that condensed samples can be accumulated over a class stream, but earlier synthetic subsets are not explicitly adapted to the subsequently expanded label space. ReCDD targets this cross-step compatibility by replaying and refining historical synthetic images as new classes arrive.

2.2 CONTINUAL LEARNING

Replay-based continual learning (CL) mitigates forgetting by training an evolving model with stored or generated past experience (Rebuffi et al., 2017; Shin et al., 2017; Buzzega et al., 2020). Subsequent work improves which samples are retained, generated, or optimized under limited memory (Liu et al., 2020; Borsos et al., 2020; Wiewel & Yang, 2021; Sangermano et al., 2022; Gu et al., 2024). In these methods, memory primarily supports continual optimization of the learner. CDD instead treats the synthetic memory itself as the optimization target and evaluates its standalone utility by training a fresh model after each step.

Recent work brings data condensation closer to continual learning. Data-Distill-Net (DDN) observes that independently condensed task memories can miss cross-task relationships and jointly distills current data with previous memory (Liao et al., 2025). Its goal remains replay-based continual learning under a fixed-capacity buffer and an evolving learner. ReCDD instead maintains a fixed-IPC synthetic dataset whose historical images are revisited as the class space expands, and evaluates the resulting memory with a fresh global classifier.

Other methods adapt compressed memories under different constraints. Remember the Past learns shared addressable memories for fresh-model training under a fixed total budget (Deng & Russakovsky, 2022). Slimmable Dataset Condensation re-condenses an existing synthetic set when the available storage budget changes (Liu et al., 2023), while Domain Incremental Dataset Distillation updates a fixed-size synthetic set across domains sharing the same label space (Oh & Sim, 2026). ReCDD instead studies class-incremental construction under a fixed per-class budget, where synthetic subsets created at different steps must remain jointly useful as new classes arrive.

3 CONTINUAL DATASET DISTILLATION

We consider class-incremental streams in which a synthetic memory is updated as new class groups arrive and evaluated by training a fresh classifier from scratch.

3.1 PROBLEM FORMULATION

Let $\mathcal{D}_{1:T}$ be a T-step class-incremental stream, where $\mathcal{D}_t = \{(x_j, y_j)\}_{j=1}^{n_t}$ contains n_t real samples with class set $\mathcal{C}_t$, $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for $i \neq j$, and $\mathcal{C}_{1:t} = \bigcup_{k=1}^{t} \mathcal{C}_k$.

CDD maintains a synthetic memory $\mathcal{S}_t$, starting from $\mathcal{S}_0 = \emptyset$. At step t, real training examples used for memory construction come only from $\mathcal{D}_t$; real examples from previous steps are not retained, and only synthetic state is carried forward. With a fixed budget of K images per class (IPC) (Zhao et al., 2021; Cui et al., 2022), $|\mathcal{S}_t| = K|\mathcal{C}_{1:t}|$. We denote by $\mathcal{S}_t^{(i)}$ the current subset for classes introduced at step i, so that $\mathcal{S}_t = \bigcup_{i=1}^{t} \mathcal{S}_t^{(i)}$.

3.2 EVALUATION PROTOCOL

Following standard DD evaluation (Wang et al., 2018; Cui et al., 2022), once $\mathcal{S}_t$ has been updated, a fresh classifier is initialized with parameters $\theta_{t,0}$ and trained solely on $\mathcal{S}_t$. The resulting classifier is then evaluated on the test data of all classes observed up to step t:

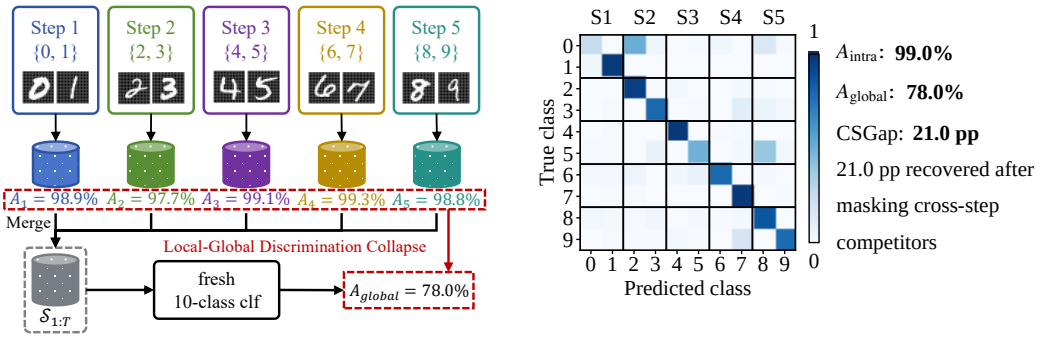
$$\theta_t^* = \mathrm{Train}(\theta_{t,0}, \mathcal{S}_t), \qquad A_t = \mathrm{Acc}\left(f_{\theta_t^*}, \mathcal{D}_{1:t}^{\mathrm{test}}\right). \tag{1}$$

No task identity is provided at test time, and the classifier predicts over the full observed label space $\mathcal{C}_{1:t}$. We use the final global accuracy A_T as the primary metric, while $\{A_t\}_{t=1}^{T}$ tracks the utility of the maintained synthetic memory throughout the stream.

For reference, *Joint* performs condensation using all real data in $\mathcal{D}_{1:T}$ under the same IPC budget. Because it requires simultaneous access to the full stream, it serves as an offline reference rather than a feasible CDD method.

3.3 LOCAL–GLOBAL DISCRIMINATION COLLAPSE

Naive CDD optimizes each synthetic subset to distinguish classes within $\mathcal{C}_t$, while the accumulated memory is ultimately used to train a classifier over $\mathcal{C}_{1:T}$. Since classes introduced at different steps



(a) High local utility does not guarantee global utility after aggregation.

(b) Masking cross-step competitors recovers global accuracy.

Figure 1: **LGDC with naive DC on canonical Split MNIST** (Zenke et al., 2017) under five steps and IPC=10. **(a)** Step-wise subsets retain high local utility, whereas their aggregated memory has substantially lower global utility. **(b)** Restricting the same global classifier to the sample's arrival-step classes reveals errors attributable to cross-step competitors.

do not directly compete during independent condensation, a subset that is effective within its local label space may become incompatible after aggregation. We refer to this failure mode as *Local–Global Discrimination Collapse* (LGDC).

Figure 1 illustrates LGDC on canonical Split MNIST. For example, samples synthesized for $\{0, 1\}$ are optimized without considering classes introduced later. Although each subset preserves high within-step utility, their aggregation can lead to degraded discrimination under the full label space.

To quantify the local–global mismatch, for each arrival step i, let a_i denote the accuracy of a fresh model trained on $\mathcal{S}_T^{(i)}$ and evaluated within $\mathcal{C}_i$. We define $A_{\text{local}} = \frac{1}{T}\sum_{i=1}^{T} a_i$, while $A_{\text{global}} = A_T$ denotes the accuracy of a fresh model trained on the full memory $\mathcal{S}_T$ and evaluated over $\mathcal{C}_{1:T}$. Their discrepancy is

$$\text{LGGap} = A_{\text{local}} - A_{\text{global}}. \tag{2}$$

To further identify whether cross-step competition contributes to this mismatch, we keep the globally trained classifier fixed and modify only the candidate classes during prediction. For a test sample (x, y) from step i, where $y \in \mathcal{C}_i$, we define the restricted prediction as

$$\hat{y}_{\text{intra}}(x) = \arg\max_{c \in \mathcal{C}_i} f_{\theta_T^*}(x)_c. \tag{3}$$

Let A_{intra} denote the accuracy under these restricted predictions. The *Cross-Step Competition Gap* is defined as

$$\text{CSGap} = A_{\text{intra}} - A_{\text{global}}. \tag{4}$$

The classifier, logits, and test samples remain unchanged; only classes from other steps are removed from consideration. Since a globally correct prediction remains correct after this restriction, CSGap measures the accuracy recovered by removing cross-step competitors. Step identity is used only for this diagnostic and is unavailable during actual global evaluation.

A nonzero CSGap is not unique to sequential condensation, since restricting the candidate set can benefit any classifier. Therefore, we compare it with Joint condensation as an offline control, computing the same diagnostic under the corresponding class grouping. Finally, LGDC differs from catastrophic forgetting in continual learning, where a continuously updated model loses performance on previously learned classes due to parameter changes (Zenke et al., 2017). In CDD evaluation, each classifier is trained from scratch on the maintained synthetic memory, and no model is transferred across steps. LGDC instead reflects incompatibility among synthetic subsets: samples that are locally useful within their original condensation context may fail to support discrimination when combined into the global label space.

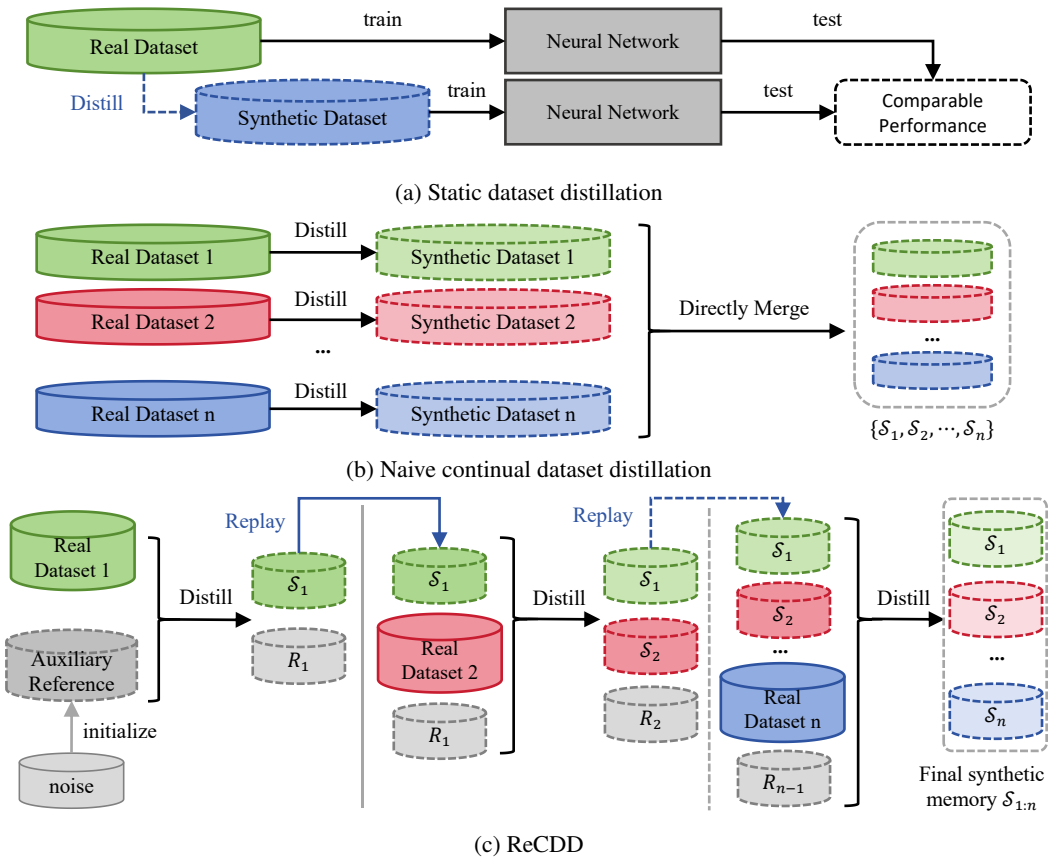


Figure 2: **Static DD, naive CDD, and ReCDD.** CDD methods construct class-specific synthetic memories as classes arrive. Naive CDD appends independently condensed subsets, whereas ReCDD further replays and refines synthetic history while maintaining an auxiliary reference.

4 Replay-Enhanced Continual Dataset Distillation

The diagnosis in Sec. 3 suggests that condensation should account for broader class contexts. ReCDD retains the base method's matching criterion while introducing historical synthetic replay and refinement. It additionally maintains an auxiliary reference during condensation, providing a stable non-semantic reference alongside class-associated synthetic samples. Figure 2 contrasts this process with static DD and naive CDD.

4.1 Historical Synthetic Replay and Refinement

At step t, ReCDD initializes the historical memory as $\mathcal{H}_t^{(0)} = \mathcal{S}_{t-1}$ and synthesizes new samples $\mathcal{N}_t^{(0)}$ for the arriving classes $\mathcal{C}_t$. Unlike naive CDD, which only optimizes samples for the current classes, ReCDD optimizes the maintained synthetic memory with respect to the accumulated label space $\mathcal{C}_{1:t}$. Specifically, historical synthetic samples provide additional cross-step context when optimizing new samples. For gradient matching methods, they participate in training the matching network used to compute gradients; for trajectory-based methods, they are included in expert training and student trajectory optimization. Thus, replay allows newly synthesized samples to adapt to previously maintained classes.

Beyond replay, ReCDD further updates historical synthetic samples to adapt the existing memory as new classes arrive. When $\mathcal{H}_t$ is nonempty, historical samples are refined using their stored labels

without requiring historical real data:

$$\mathcal{L}_{\text{hist}} = \frac{1}{|\mathcal{H}_t|} \sum_{(\tilde{x},y)\in\mathcal{H}_t} \ell_{\text{ce}}\Big(f_{\theta_t^{\text{ref}}}(\tilde{x}), y\Big), \tag{5}$$

where θ_t^{ref} denotes the fixed reference network state for refinement, corresponding to the matching network in DC/DSA and the expert target state in trajectory matching. During refinement, gradients are propagated to update historical images while keeping the reference parameters fixed. Replay provides optimization context for new samples, whereas $\mathcal{L}_{\text{hist}}$ directly maintains historical memory.

Since repeated refinement may alter older synthetic subsets, ReCDD applies a smaller update rate to historical samples than to newly initialized samples. For a sample introduced at step $s < t$, the update rate is decayed according to $\eta_{t,s}^{\text{hist}} = \eta_{\text{h}} \exp[-(t-s)]$, where η_{h} is smaller than the learning rate for new samples. This schedule reduces the influence of repeated updates on older subsets. After optimization, the refined historical memory and newly synthesized samples are merged as $\mathcal{S}_t = \mathcal{H}_t^* \cup \mathcal{N}_t^*$.

4.2 Auxiliary Reference

Replay enables arriving classes to interact with historical synthetic samples, but it cannot provide constraints from future classes when earlier samples are first synthesized. Therefore, early classes may be optimized under an incomplete label space and require later adaptation as new classes arrive. To provide a consistent non-semantic competitor throughout the stream, ReCDD maintains an auxiliary reference with one category $r \notin \mathcal{C}_{1:t}$ outside the observed label space and K auxiliary images $\mathcal{R}_t = \{\tilde{x}_{t,j}^r\}_{j=1}^K$. The auxiliary images are initialized from noise and propagated across steps, providing a stable reference during condensation without introducing additional semantic classes.

During condensation, the auxiliary reference introduces K additional optimization samples besides the $K|\mathcal{C}_{1:t}|$ semantic samples. These auxiliary samples are discarded before evaluation, and the evaluated memory therefore contains exactly K images per observed class. The auxiliary reference is independent of the underlying condensation objective: historical synthetic samples provide cross-step context through replay, while the auxiliary category only participates during optimization. The auxiliary reference follows the same method-specific integration as replay: it enters the matching network for gradient matching and the trajectory construction process for trajectory-based methods. Historical refinement is performed using Eq. 5.

5 Experiments

5.1 Experimental Setup

Datasets and task splits. We evaluate ReCDD on MNIST (LeCun et al., 1998), CIFAR-10 and CIFAR-100 (Krizhevsky, 2009), and Tiny ImageNet (Le & Yang, 2015) under the class-incremental CDD setting in Sec. 3. MNIST and CIFAR-10 use five steps with two classes per step; CIFAR-100 and Tiny ImageNet are split into ten steps with ten and twenty classes per step, respectively.

Methods and references. For DC, DSA, MTT, FTD, and DATM, we compare ReCDD with a sequential baseline (Seq.) that condenses each arriving class group independently and keeps previous synthetic images unchanged. Joint rows are offline references obtained by condensing the full dataset jointly before evaluation. Random‡ retains K randomly selected real examples per arriving class and serves as a real-data coreset reference.

Evaluation protocol. We report final accuracy under $K \in 1, 10, 50$ following Sec. 3.2. Sequential methods use three class orders generated from different seeds, and all compared methods share the same orders. For each order, we independently perform condensation and train a fresh evaluator from scratch, reporting mean$\pm$std over the resulting runs.

Joint rows use published full-data results at matched IPC: DC and DSA from Zhao et al. (2021); Zhao & Bilen (2021), MTT from Cazenavette et al. (2022); Du et al. (2023), and FTD/DATM from Guo et al. (2024). These values serve as offline references rather than sequential CDD results.

Table 1: Final global test accuracy (%) for gradient matching methods. Sequential results are mean±std over three class orders. Joint results are offline references at matched IPC.

Method	MNIST			CIFAR-10		
	IPC=1	IPC=10	IPC=50	IPC=1	IPC=10	IPC=50
Random‡	55.7 ± 3.6	91.1 ± 1.0	97.1 ± 0.3	15.7 ± 2.5	27.8 ± 1.8	45.2 ± 0.8
DC (Seq.)	84.9 ± 0.1	85.5 ± 2.3	83.9 ± 2.6	29.3 ± 0.6	40.1 ± 0.8	44.3 ± 0.5
ReCDD-DC	86.4 ± 0.3	93.7 ± 1.5	97.0 ± 0.1	29.2 ± 0.6	44.9 ± 0.3	49.7 ± 0.3
DC (Joint)	91.7 ± 0.5	97.4 ± 0.2	98.8 ± 0.2	28.3 ± 0.5	44.9 ± 0.5	53.9 ± 0.5
DSA (Seq.)	85.9 ± 0.3	90.9 ± 1.1	92.3 ± 2.7	28.8 ± 0.2	34.4 ± 1.0	46.8 ± 0.3
ReCDD-DSA	85.8 ± 0.3	95.3 ± 0.2	97.5 ± 0.1	29.4 ± 0.6	42.1 ± 0.9	51.7 ± 0.4
DSA (Joint)	88.7 ± 0.6	97.8 ± 0.1	99.2 ± 0.1	28.8 ± 0.7	52.1 ± 0.5	60.6 ± 0.5

Table 2: Final global test accuracy (%) for trajectory matching methods. Sequential results are mean±std over three class orders. Joint rows are offline references at matched IPC.

Method	CIFAR-10			CIFAR-100			Tiny ImageNet		
	IPC=1	IPC=10	IPC=50	IPC=1	IPC=10	IPC=50	IPC=1	IPC=10	IPC=50
Random‡	15.7 ± 2.5	27.8 ± 1.8	45.2 ± 0.8	4.5 ± 0.2	15.4 ± 0.2	30.3 ± 0.2	1.3 ± 0.3	4.6 ± 0.6	14.6 ± 0.6
MTT (Seq.)	27.0 ± 1.0	42.4 ± 1.4	42.2 ± 3.6	11.7 ± 0.4	17.6 ± 0.7	28.5 ± 0.0	4.4 ± 0.1	18.9 ± 0.4	25.4 ± 0.6
ReCDD-MTT	34.5 ± 0.5	50.9 ± 1.1	55.9 ± 1.2	16.8 ± 0.2	26.4 ± 0.5	36.9 ± 0.6	7.5 ± 0.7	18.9 ± 0.3	25.8 ± 0.5
MTT (Joint)	46.2 ± 0.8	65.4 ± 0.7	71.6 ± 0.2	24.3 ± 0.3	39.7 ± 0.4	47.7 ± 0.2	8.8 ± 0.3	23.2 ± 0.2	28.0 ± 0.3
FTD (Seq.)	27.7 ± 2.2	41.5 ± 1.6	41.4 ± 3.3	11.1 ± 0.9	17.3 ± 0.2	27.7 ± 0.8	4.1 ± 0.4	18.7 ± 0.3	25.6 ± 0.1
ReCDD-FTD	35.2 ± 0.1	50.8 ± 1.4	55.6 ± 1.6	16.4 ± 0.6	26.7 ± 0.3	37.1 ± 0.5	7.4 ± 0.4	18.5 ± 0.4	25.8 ± 0.3
FTD (Joint)	46.0 ± 0.4	65.3 ± 0.4	73.2 ± 0.2	24.4 ± 0.4	42.5 ± 0.2	48.5 ± 0.3	10.5 ± 0.2	23.4 ± 0.3	28.2 ± 0.4
DATM (Seq.)	26.7 ± 0.3	42.3 ± 1.7	45.2 ± 2.5	12.7 ± 1.1	20.2 ± 0.1	32.3 ± 0.1	7.0 ± 0.3	16.2 ± 0.1	27.2 ± 0.6
ReCDD-DATM	35.3 ± 0.4	49.9 ± 0.9	58.5 ± 0.7	18.1 ± 0.2	23.3 ± 0.7	35.7 ± 0.9	7.4 ± 0.3	18.3 ± 0.6	27.6 ± 0.2
DATM (Joint)	46.9 ± 0.5	66.8 ± 0.2	76.1 ± 0.3	27.9 ± 0.2	47.2 ± 0.4	55.0 ± 0.2	17.1 ± 0.3	31.1 ± 0.3	39.7 ± 0.3

For the LGDC diagnostic in Table 3, we additionally reproduce one-step Joint DC and DSA under our evaluation pipeline using three independent runs. These reproduced results are marked by † and provide diagnostic controls under the same classifier evaluation. For consistency with sequential streams, we assign the corresponding class groups to Joint samples and compute A_{intra} and CSGap using the restricted candidate-class evaluation defined in Sec. 3.

5.2 Main Results

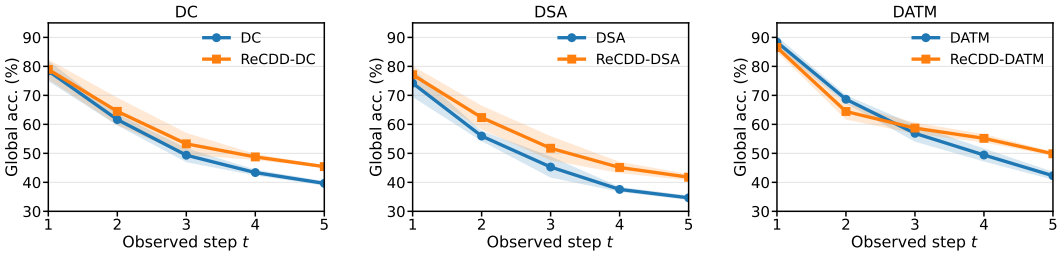
Gradient matching. ReCDD improves DC and DSA on MNIST and CIFAR-10, with clear gains at IPC=10 and IPC=50 (Table 1).

Trajectory matching. ReCDD improves MTT, FTD, and DATM across CIFAR-10 and CIFAR-100, while remaining effective on the more challenging Tiny ImageNet setting (Table 2).

Figure 3 further shows that the improvement is not limited to the final memory. ReCDD-DC and ReCDD-DSA maintain higher accuracy throughout the stream. For DATM, ReCDD initially follows the baseline but achieves higher accuracy after the third step, showing that its benefit emerges as the observed label space expands.

5.3 Analysis of Local–Global Discrimination Collapse

Table 3 reveals the local–global mismatch underlying LGDC across MNIST and CIFAR-10, DC and DSA, and multiple class orders. For naive condensation, A_{intra} remains close to A_{local} while A_{global} drops substantially, indicating that a large portion of global errors can be recovered by removing cross-step competitors. For example, CIFAR-10 DC achieves 81.8% accuracy when predictions are restricted to the true group but only 40.1% under global prediction, resulting in a CSGap of 41.8 pp.



(a) DC (b) DSA (c) DATM

Figure 3: **Synthetic-memory utility on CIFAR-10 at IPC=10.** Each point evaluates a fresh ConvNet trained on $\mathcal{S}_t$; curves show mean±std over three class orders.

Table 3: **LGDC analysis at IPC=10.** Sequential results are mean±std over three class orders; Joint† denotes reproduced one-step controls under the same class grouping. Dashes mark metrics not evaluated for Joint; accuracies are in % and gaps in pp.

Dataset	Method	$A_{\text{local}} \uparrow$	$A_{\text{intra}} \uparrow$	$A_{\text{global}} \uparrow$	LGGap↓	CSGap↓
MNIST	DC (Seq.)	98.6 ± 0.5	98.7 ± 0.4	85.5 ± 2.3	13.1 ± 2.3	13.3 ± 2.3
	ReCDD-DC	98.9 ± 0.1	99.3 ± 0.1	93.7 ± 1.5	5.1 ± 1.4	5.6 ± 1.5
	DC (Joint)†	–	99.3 ± 0.0	96.5 ± 0.1	–	2.9 ± 0.1
	DSA (Seq.)	98.8 ± 0.3	99.0 ± 0.3	90.9 ± 1.1	7.8 ± 0.8	8.1 ± 0.8
	ReCDD-DSA	98.8 ± 0.3	99.3 ± 0.1	95.3 ± 0.2	3.5 ± 0.2	4.0 ± 0.2
	DSA (Joint)†	–	99.2 ± 0.0	96.0 ± 0.0	–	3.2 ± 0.0
CIFAR-10	DC (Seq.)	82.5 ± 1.3	81.8 ± 2.1	40.1 ± 0.8	42.4 ± 2.0	41.8 ± 2.4
	ReCDD-DC	80.4 ± 0.7	82.8 ± 1.0	44.9 ± 0.3	35.5 ± 0.7	37.8 ± 0.9
	DC (Joint)†	–	82.7 ± 0.5	46.9 ± 0.2	–	35.8 ± 0.3
	DSA (Seq.)	78.9 ± 0.8	76.8 ± 2.2	34.4 ± 1.0	44.4 ± 1.0	42.3 ± 1.4
	ReCDD-DSA	79.8 ± 0.8	81.6 ± 0.5	42.1 ± 0.9	37.7 ± 1.7	39.5 ± 1.4
	DSA (Joint)†	–	82.3 ± 0.3	45.4 ± 0.3	–	36.9 ± 0.1

ReCDD reduces CSGap in all four sequential comparisons across datasets and methods, while reproduced Joint controls exhibit smaller gaps. For CIFAR-10 DC, CSGap decreases from 41.8 pp to 37.8 pp, compared with 35.8 pp for Joint. Under this matched diagnostic, independently condensed memories contain larger recoverable errors caused by cross-step candidates, and ReCDD reduces this discrepancy.

The improvement is not explained by stronger local discrimination alone. On CIFAR-10, ReCDD-DC decreases A_{local} from 82.5% to 80.4%, while increasing A_{global} from 40.1% to 44.9%. Meanwhile, A_{intra} improves from 81.8% to 82.8%, showing that the reduced CSGap is accompanied by improved global utility rather than degraded within-group discrimination.

5.4 Ablation Studies

We ablate DC-based ReCDD at IPC=10 by varying historical memory handling and the auxiliary reference. *None* keeps historical subsets unchanged without replay or refinement, *Frozen* replays them without pixel updates, and *Trainable* additionally enables historical refinement. Runs use fixed canonical class orders; macro accuracy equally weights arrival groups under the same fresh final classifier over $\mathcal{C}_{1:T}$.

The ablation reveals complementary roles of the two components (Table 4). Adding the auxiliary reference consistently improves all historical-memory settings. Simply replaying historical synthetic images is insufficient: *Frozen* performs worse than *None* on both datasets, suggesting that fixed historical subsets may conflict with the expanded label space. Allowing historical refinement reverses this trend and achieves the highest accuracy on both datasets, reaching 94.8% on MNIST and 45.6% on CIFAR-10. On CIFAR-10, the complete model achieves a similar performance level

Table 4: **DC component ablation at IPC=10.** Final macro accuracy (%), mean±std over three optimization seeds.

Historical memory	Aux. reference	MNIST	CIFAR-10
None	–	81.5 ± 1.3	40.0 ± 0.8
None	✓	90.1 ± 2.4	45.4 ± 0.3
Frozen	–	74.0 ± 1.2	30.1 ± 0.5
Frozen	✓	79.1 ± 1.5	32.2 ± 1.1
Trainable	–	85.6 ± 0.4	42.5 ± 1.1
Trainable	✓	$\mathbf{94.8 \pm 0.3}$	$\mathbf{45.6 \pm 0.4}$

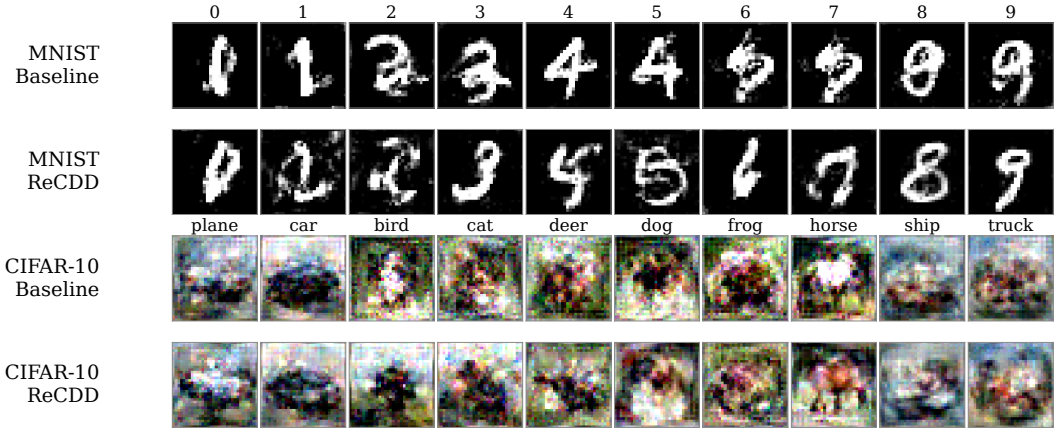


Figure 4: **Final synthetic-memory examples on MNIST and CIFAR-10 at IPC=10.** Sequential DC ("Baseline") and ReCDD-DC use the same class order, with one example shown per class.

to the auxiliary-reference-only setting (45.6% vs. 45.4%), while still improving over configurations without historical adaptation.

5.5 Visualizing the Synthetic Memory

Figure 4 compares final DC and ReCDD-DC memories at IPC=10; MNIST follows the canonical Split MNIST order $\{0, 1\}, \ldots, \{8, 9\}$. On MNIST, several DC samples from classes arriving in the same step exhibit similar digit shapes. This is visible for pairs such as $\{6, 7\}$ and $\{8, 9\}$, where local separation can be achieved through subtle within-pair differences, while the resulting samples appear less separated under the full label space. ReCDD reduces this within-step homogeneity and produces more distinct digit structures. On CIFAR-10, the difference is less pronounced, but ReCDD samples exhibit visually clearer class-related structures, while DC samples from some arrival groups remain more homogeneous. These qualitative observations complement the quantitative LGDC analysis in Table 3, which directly measures global discrimination and cross-step competition.

6 Conclusion

Continual Dataset Distillation requires synthetic memories constructed at different steps to remain compatible as the label space expands. We identify LGDC as a local–global mismatch and propose ReCDD, which replays and refines historical synthetic memory with an auxiliary reference. Across five matching methods and four datasets, ReCDD improves continually maintained synthetic-memory utility, with diagnostics showing reduced local–global discrepancy and cross-step competition. These results indicate that continual synthetic-memory construction benefits from adapting historical states rather than merely retaining them.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript drafting and editing, and to support code implementation and modification under author-specified designs and requirements. The authors developed the research questions, methodology, and experimental objectives. All AI-assisted outputs were reviewed and verified by the authors. The authors take responsibility for the final manuscript.

REFERENCES

Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. *Advances in neural information processing systems*, 33:14879–14890, 2020.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10718–10727. IEEE, 2022.

Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. DC-BENCH: Dataset condensation benchmark. *Advances in Neural Information Processing Systems*, 35:810–822, 2022.

Justin Cui, Ruochen Wang, Si Si, and Cho-Jui Hsieh. Scaling up dataset distillation to ImageNet-1K with constant memory. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 6565–6590. PMLR, 2023.

Zhiwei Deng and Olga Russakovsky. Remember the past: Distilling datasets into addressable memories for neural networks. In *Advances in Neural Information Processing Systems*, volume 35, pp. 34391–34404, 2022.

Jiawei Du, Yidi Jiang, Vincent YF Tan, Joey Tianyi Zhou, and Haizhou Li. Minimizing the accumulated trajectory error to improve dataset distillation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3749–3758. IEEE, 2023.

Jianyang Gu, Kai Wang, Wei Jiang, and Yang You. Summarizing stream data for memory-constrained online continual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12217–12225, 2024. doi: 10.1609/aaai.v38i11.29111.

Ziyao Guo, Kai Wang, George Cazenavette, Hui Li, Kaipeng Zhang, and Yang You. Towards lossless dataset distillation via difficulty-aligned trajectory matching. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/fc740286fb4840cb56f99400817967da-Abstract-Conference.html.

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 *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 11102–11118. PMLR, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. CS231N course project report, Stanford University, 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Wenyang Liao, Quanziang Wang, Yichen Wu, Renzhen Wang, and Deyu Meng. Data-Distill-Net: A data distillation approach tailored for reply-based continual learning. *arXiv preprint arXiv:2505.20135*, 2025. URL https://arxiv.org/abs/2505.20135.

Songhua Liu, Jingwen Ye, Runpeng Yu, and Xinchao Wang. Slimmable dataset condensation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3759–3768. IEEE, 2023.

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 (CVPR)*, pp. 12245–12254. IEEE, 2020.

Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=l-PrrQrK0QR.

Minyoung Oh and Jae-Young Sim. Asymmetric synthetic data update for domain incremental dataset distillation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/319772297aad04520fdb302e208acdc0-Abstract-Conference.html.

Ameya Prabhu, Philip Torr, and Puneet Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *The European Conference on Computer Vision (ECCV)*, August 2020.

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)*, pp. 2001–2010. IEEE, 2017.

Mattia Sangermano, Antonio Carta, Andrea Cossu, and Davide Bacciu. Sample condensation in online continual learning. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2022. doi: 10.1109/IJCNN55064.2022.9892299.

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

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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12196–12205. IEEE, 2022.

Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. URL https://arxiv.org/abs/1811.10959.

Felix Wiewel and Bin Yang. Condensed composite memory continual learning. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2021. doi: 10.1109/IJCNN52387.2021.9533491.

Enneng Yang, Li Shen, Zhenyi Wang, Tongliang Liu, and Guibing Guo. An efficient dataset condensation plugin and its application to continual learning. *Advances in Neural Information Processing Systems*, 36:67625–67642, 2023.

Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 73582–73603. Curran Associates, Inc., 2023. doi: 10.52202/075280-3219. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/e91fb65c6324a984ea9ef39a5b84af04-Paper-Conference.pdf.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3987–3995. PMLR, 2017.

Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12674–12685. PMLR, 2021.

Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 6514–6523. IEEE, 2023. doi: 10.1109/WACV56688.2023.00645.

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=mSAKhLYLSsl.