PAIRED-CONTROL CONTRIBUTION AUDITS FOR DIFFUSION-ASSISTED INCREMENTAL RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Class-incremental learning (CIL) and its few-shot variant (FSCIL) train a classifier on classes that arrive over time while it must keep recognizing earlier ones. Recent methods insert diffusion models into this pipeline, for example to generate replay images of earlier classes, to select among generated samples, or to supply frozen features. When such a method gains accuracy, the gain may come from the generated samples, from simply training on more data, from the selection step, from a stronger representation, or from extra training steps, and final accuracy alone cannot tell these apart. We propose a paired-control contribution audit, an evaluation protocol that tests which component supports such a gain. For each way a method uses diffusion, we rerun the same method with only that use replaced by a simple substitute, for example random instead of reward-based selection, while keeping the data, seeds, and training steps fixed, and we report the accuracy difference over repeated matched runs. We apply the audit to six diffusion-assisted CIL and FSCIL image-classification methods and check each conclusion against alternative substitutes. Each comparison shows what supports a gain only in the tested setting, and the answers differ across methods. Diffusion-generated image replay in one method raises average incremental accuracy by up to 10.1 points, and the gain is not explained by extra training steps, whereas in three other methods the diffusion component shows no reliable gain over a simpler substitute, such as an unselected pool of generated images or replay of stored class statistics. We therefore recommend reporting, alongside final accuracy, how each diffusion component compares with its simplest substitute and the settings in which each claim was tested. Code is available at `https://anonymous.4open.science/r/dca-iclr27-F078`.

1 INTRODUCTION

Incremental recognition asks a model to learn classes that arrive over a sequence of sessions while retaining earlier ones. In class-incremental learning (CIL) every session adds a disjoint set of classes (Rebuffi et al., 2017; Li & Hoiem, 2016; Hou et al., 2019; Wang et al., 2022); in few-shot class-incremental learning (FSCIL) later sessions provide only a few labeled examples per class (Tao et al., 2020; Zhang et al., 2021; Peng et al., 2022). Both settings invite generative assistance, and diffusion models are now used for text-to-image replay (Jodelet et al., 2023; Wu et al., 2025; Meng et al., 2024), feature-space replay (Zhang et al., 2024), frozen diffusion features (Kim et al., 2025b), few-shot synthetic augmentation (Kim et al., 2025a), reward-guided sample selection (Wu et al., 2025), and synthetic-to-real calibration (Meng et al., 2024).

This design space creates an attribution problem. A diffusion component brings prompts, generated caches, filtering rules, synthetic/real ratios, additional optimizer steps, and sometimes a different representation backbone. The same accuracy gain can therefore come from better samples, replay-induced regularization, selection over a weak pool, calibration, or a stronger frozen representation. Leaderboards compare complete systems, and standard ablations remove a module, but neither necessarily produces the counterfactual that a diffusion claim needs. Removing the reward in a reward-guided replay method leaves an unselected generated pool, not a no-generation baseline; removing feature diffusion does not test whether stored class statistics would suffice. A comparison is easiest to interpret when the method is run twice, with and without the component, in the same

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.

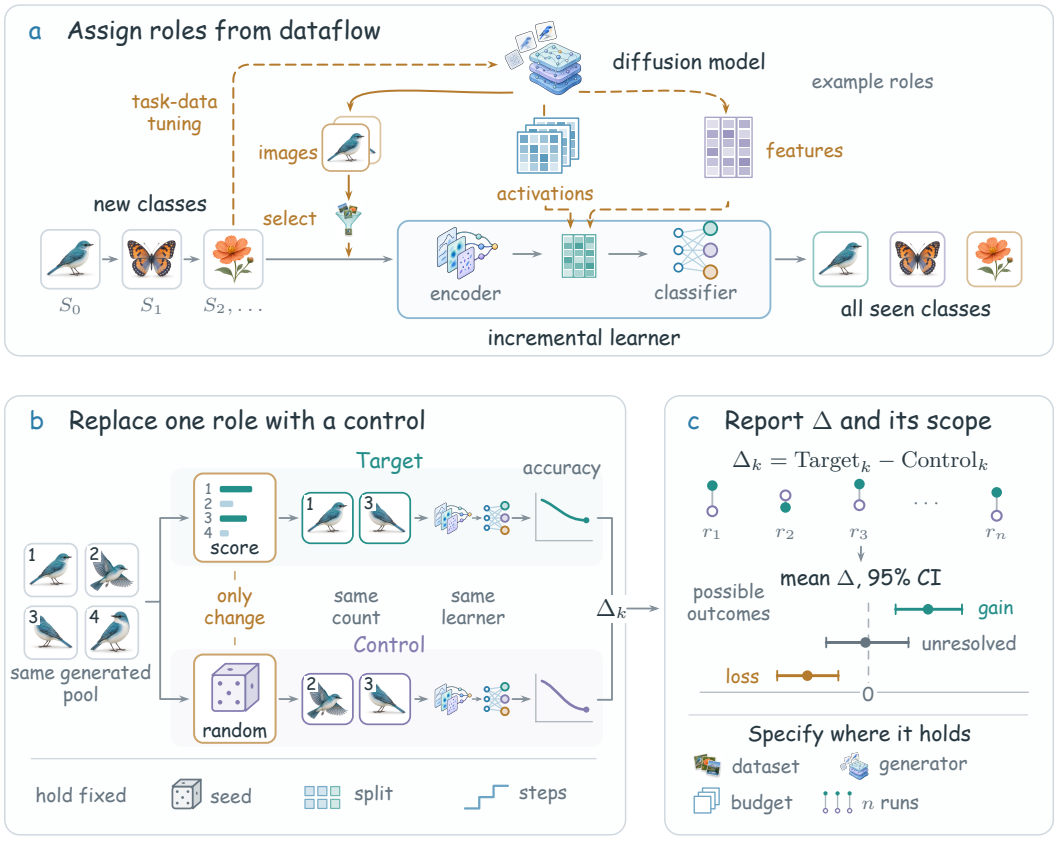

Figure 1: Overview of the contribution audit. (a) Assign roles from dataflow: we trace where a diffusion model enters an incremental learner that receives sessions $S_0, S_1, \ldots$, for example generated images with selection, diffusion activations, generated features, or generator tuning on task data, and assign roles accordingly (Table 1). (b) Replace one role with a control: the Control changes only that operation. Here, scoring the same generated pool is replaced by a random choice of the same count, and the seed, split, learner, and training steps stay matched. (c) Report Δ and its scope: matched runs give paired differences Δ, reported as a mean and 95% CI together with the settings they cover.

training code and setup, which we call the scaffold, and with the same data split and random seeds. We call this a paired comparison and use it as the unit of our audit.

The same attribution problem arises whenever a large pretrained generator is inserted into a learning pipeline, for instance when synthetic images are used as training data (He et al., 2023; Trabucco et al., 2024); retrieving real images from the generator's pretraining corpus can even outperform generating them (Geng et al., 2024). Controlled re-evaluations have changed conclusions in other areas of machine learning (Baek et al., 2019; Musgrave et al., 2020), and reporting run-to-run variability changes how small differences should be read (Bouthillier et al., 2021; Agarwal et al., 2021). We bring this evaluation practice to diffusion-assisted incremental recognition and ask a narrow question:

> *When a diffusion-assisted method improves, which component supports the improvement in a fully specified setting, and does the answer survive a different reasonable control?*

As Figure 1 illustrates, the audit first traces where diffusion enters a method, then reruns the method with only that use replaced by a simple substitute while everything else stays fixed, and finally reports the paired accuracy difference together with the settings in which it holds.

Our contributions are:

1. **A reusable audit protocol** (Section 3). It assigns each use of diffusion to a functional role by tracing the method's dataflow, independently of the results, and chooses the simplest

substitute for each role, with alternative substitutes as checks. An audit card records what was held fixed, what could not be matched, and the number of runs.

2. **An audit of six diffusion-assisted CIL/FSCIL methods** (Sections 4–5), with repeated matched runs, accuracy curves over sessions, and alternative substitutes that additionally match training compute, data exposure, labels, selection, replay budget, or the generator.
3. **Findings on what supports the gains** (Section 5). The answer differs across methods. Diffusion image replay in DiffClass (Meng et al., 2024) improves average incremental accuracy by up to 10.1 points on CIFAR-100, and a control with matched training steps shows that the gain is not explained by extra optimization. In three other methods, the diffusion component shows no reliable gain over a simpler substitute, such as the unselected pool of generated images or replay of stored class statistics. Several conclusions also change with the dataset, generator, or replay budget, so we recommend reporting each claim together with its substitute and the settings in which it was tested.

2 Related Work

Incremental recognition and generative replay. CIL and FSCIL methods retain old classes through rehearsal, distillation, prototype classifiers, calibration, or frozen backbones (Belouadah et al., 2020; Masana et al., 2022; Zhou et al., 2024; Tian et al., 2023; Zhang et al., 2025; Li et al., 2025). Generative replay predates diffusion and has synthesized images, features, feature statistics, and prompts (Shin et al., 2017; Smith et al., 2021; Liu et al., 2022; Huang et al., 2024; Frascaroli et al., 2024); replay, stored statistics, and representation strength can therefore help without pixel-level diffusion, which is why our controls include such non-diffusion alternatives; feature-level replay (van de Ven et al., 2020), Gaussian prototype augmentation (Zhu et al., 2021), feature translation (Petit et al., 2023), and class-statistics classifiers (Hayes & Kanan, 2020; Goswami et al., 2023) are close to the Gaussian and residual controls we use for DiffFR.

Diffusion-assisted incremental recognition. Diffusion models (Ho et al., 2020; Rombach et al., 2022) are used for image replay or augmentation (Gao & Liu, 2023; Jodelet et al., 2023; 2025; Kim et al., 2025a; Meng et al., 2024), reward-guided replay (Wu et al., 2025), feature-space replay (Zhang et al., 2024), diffusion-derived representations (Kim et al., 2025b), and classifier-guided replay (Cywiński et al., 2024), often combined with textual inversion, LoRA, or DreamBooth adaptation (Gal et al., 2023; Hu et al., 2022; Ruiz et al., 2023). Kim et al. (2025a) vary the generator, prompting, quantity, and timing of synthetic replay in FSCIL; our audit extends this design-axis view from synthetic-image replay to several diffusion roles across CIL and FSCIL. Related uses in detection, semantic segmentation, federated continual learning, and training-free FSCIL (Kim et al., 2024; Chen et al., 2023; Liang et al., 2024; Mei et al., 2024; Kang et al., 2025), and classifiers trained jointly with the diffusion model (Skierś & Deja, 2025), would require new roles and controls and are outside our scope.

Evaluation methodology and synthetic training data. Unified modular re-implementations (Baek et al., 2019) and controlled benchmarks (Musgrave et al., 2020) have shown that comparisons across papers are confounded by inconsistent settings and that many reported gains shrink under matched conditions; in continual learning, simple baselines (Prabhu et al., 2020) and compute-budgeted comparisons (Prabhu et al., 2023) likewise question reported progress. Our audit is complementary: it fixes the counterfactual inside each method's own pipeline, which keeps the tested component faithful to its method. Work on synthetic training data asks whether generated images help recognition (He et al., 2023; Trabucco et al., 2024; Sariyildiz et al., 2023; Azizi et al., 2023) or can serve as inputs for distillation (Yin et al., 2020); Geng et al. (2024) find that retrieved real images from the generator's pretraining data often do better, and diffusion models can reproduce training images (Somepalli et al., 2023; Carlini et al., 2023). These results motivate the direct overlap screen in Appendix I.

3 Contribution Audit Protocol

The audit has three steps, which match panels (a)–(c) of Figure 1: assigning roles from dataflow (Section 3.1), replacing one role with a control (Section 3.2), and reporting the paired difference Δ and its scope (Section 3.3).

Table 1: Diffusion roles and their primary controls. A role is assigned when its condition holds for at least one method branch, and the primary control removes or replaces only that role. Table 11 in Appendix C adds alternative and context controls, full conditions, and residual confounders.

Role	Assigned when	Primary control
Pixel-image replay or augmentation	generated images enter training or replay	same pipeline without generated images
Diffusion feature replay	a diffusion process synthesizes replay features	replay of stored class statistics
Diffusion-network representation	diffusion activations of real images enter prediction	none within the family; other encoders give context
Selection or reward guidance	a score filters, ranks, or steers generated candidates	same candidates and count without scoring
Generator or conditioning adaptation	task data tune the generator or its conditioning	fixed class-name conditioning
Latent or diffusion-feature augmentation	an extra diffusion view is added to training	base view repeated to match rows and steps
Synthetic-to-real calibration	a module aligns real and synthetic data	module disabled, all else fixed

3.1 Assigning roles from dataflow

Roles are assigned from dataflow, before any result is inspected. The annotator traces every executable path from a diffusion model, a diffusion-derived cache, or a diffusion-network activation to the recognition loss or prediction, and assigns a role if and only if all of its necessary conditions hold for at least one audited branch (Table 1; full conditions and examples in Appendix C). A method may receive several roles. Evidence is taken in the order executable code and configuration, then algorithm or pseudocode, then method prose; ambiguous assignments are recorded with an uncertainty flag. Generator identity, output quality, replay quantity, optimization compute, and generic distillation are treated as matching variables or residual confounders. The codebook gives operational definitions for reproducible assignment.

3.2 Replacing one role with a control

For each assigned role, the *Target* pipeline contains the diffusion component and the *Control* pipeline removes or replaces it. The **primary control** is the smallest such intervention that keeps the dataset split, class order, evaluator, classifier path, row and label budget, and unrelated optimization schedule fixed whenever feasible. When one control cannot rule out the relevant alternatives, we report two complementary primary controls, for example Gaussian class statistics and stored feature residuals for feature replay. **Alternative controls**, such as an exposure-matched repeat, a permuted label, another generator, or a replay-budget grid, test whether the conclusion from the primary control is stable (Section 5.2). Results such as real replay or strong non-diffusion encoders give scale and are reported as context. If one intervention changes several roles, we use a factorial design when feasible and otherwise report the combined effect as a package contrast. Any change that the intervention cannot avoid is recorded as a residual confounder.

3.3 Reporting Δ and its scope

A method observes sessions $S_0, \ldots, S_L$; S_0 is the base session. Let a_t be top-1 accuracy on all classes seen up to session t. We report average incremental accuracy $\mathrm{AA} = \frac{1}{L+1} \sum_{t=0}^{L} a_t$ and final accuracy a_L. For a diffusion role r, a *Target* pipeline contains the component and a *Control* pipeline replaces it by the intervention chosen in Section 3.2; everything else is held fixed where the method allows. For matched run k (same seed, class order, and split in both pipelines), the paired difference is $\Delta_k = m(\mathrm{Target}_k) - m(\mathrm{Control}_k)$ for a metric m. We report the mean of Δ_k, a two-sided Student-t 95% confidence interval (CI) over the n matched runs, and n. The unit of replication is the matched run. Differences are never pooled across families.

Each result reports n, the number of matched run pairs. Results that cover only part of the protocol, such as base-session accuracy with fixed feature caches, are reported as context and are not used for primary conclusions.

Table 2: Audited method families (condensed). Roles follow Table 1, and the diffusion source is the model used in our audit. Residual confounders are factors that the primary control leaves open; Table 5 and Appendix D probe them, and Table 12 gives the full descriptions. Split notation (k tasks, T10/T25, T5–T20) is defined in Appendix B.

Family	Regime and data	Diffusion source	Roles	Primary control(s)	Residual confounders
DCS (Wu et al., 2025)	FSCIL; CUB-200 5-shot (+ CUB-200 1-shot, miniImageNet 1/5-shot)	SD3.5-Medium	pixel replay; selection	no generation; weak SD (unselected pool)	reward-model compute; accepted samples
DiffClass (Meng et al., 2024)	CIL; CIFAR-100, ImageNet-100; 5/10/20 tasks	SD1.5 cache	pixel replay; selection; calibration	no generation; SSIA off; MDA off	extra rows and steps; generator recipe
SDDR (Jodelet et al., 2023)	CIL; CIFAR-100; T10/T25	SD1.5	pixel replay	label-matched noise cache	image semantics vs. statistics
Can Synthetic (Kim et al., 2025a)	FSCIL; CUB-200, miniImageNet; 5-shot	SD1.5 + Textual Inversion	pixel replay; adaptation	no synthetic replay; class-name prompts	extra rows; inversion compute
DiffFR (Zhang et al., 2024)	CIL; CIFAR-100; T5/T10/T20	feature diffusion	feature replay	Gaussian statistics or stored feature residuals	diffusion training compute
Beyond (Kim et al., 2025b)	FSCIL; CUB-200 5-shot	SD1.5 U-Net features	representation; augmentation	repeated base view F_{inv}	cache construction

Each audited contrast is summarized by an **audit card** (template in Appendix C.1) that records: the role and its evidence anchor; Target, Control, factors held fixed, and residual confounders; the repeated-run unit and n; the claim settings (dataset and split, class order, generator and version, replay and compute budgets); absolute results, paired differences, and CIs; and the conclusion the contrast supports. Across declared settings we apply a **stability rule**: each setting is labeled positive, negative, or unresolved according to whether its CI lies above zero, below zero, or covers zero, and a directional claim across settings requires that they agree. For an unresolved setting, the CI bounds give the largest effect in either direction that the data are compatible with. Table 4 in Section 5.1 shows a complete card.

Applying the protocol to a new method. GUIDE (Cywiński et al., 2024), which is outside the six audited families, shows how the protocol applies from a method description alone. Its current classifier steers the sampling trajectory of the previous-task diffusion model, and the resulting old-class images are replayed, so GUIDE receives two roles: image replay and selection or reward guidance. The primary control for guidance runs the same guidance path with the applied guidance vector set to zero, keeping the checkpoints, initial noise, sampler, labels, image count, replay exposure, optimizer, and evaluator fixed. Permuting the guidance target at the same generation cost and varying the guidance scale are alternative controls, and no replay and real replay give context. The replay role is assessed separately against the same pipeline without generated images (Appendix C.1).

4 Audited Families and Setup

Families and datasets. We call each audited method, together with the variants we run for it, a family. Table 2 lists the six families, their original methods, and the diffusion model used in our audit. Datasets follow each family's learning regime and the original method's benchmarks: CIL families use CIFAR-100 (Krizhevsky, 2009) (and ImageNet-100, a 100-class ImageNet subset (Russakovsky et al., 2015), for DiffClass); FSCIL families use CUB-200 (Wah et al., 2011) (100 base classes followed by 10 sessions of 10 classes) and miniImageNet (Vinyals et al., 2016) (60 base classes followed by 8 sessions of 5 classes). Because a primary dataset can differ from the one used in an original paper's main ablation (DCS reports its main ablation on CIFAR-100 and a sequential reward ablation on miniImageNet and CUB-200 in its appendix; our primary DCS audit uses CUB-200), the added controls repeat each family on further datasets where feasible: DCS on miniImageNet with 1- and 5-shot, DiffFR on CUB-200, ImageNet-100, and TinyImageNet, Can Synthetic on both FSCIL datasets, and a shared-scaffold comparison of DCS and SDDR on CUB-200, miniImageNet, and CIFAR-100 (Appendix F.1). Target and Control of each family run in the same fixed scaffold; Table 2 lists the audited branches and Appendix D the held-fixed settings.

Control terminology. For image replay, *no generation* is the same training scaffold with no synthetic images. In DCS, *reward-selected replay* is generated by the method's reward-guided SMC

Table 3: Primary Target−Control contrasts in average incremental accuracy (AA, pp): paired mean differences with 95% CIs over n matched runs. Green and orange mark CIs entirely above or below zero. Appendix E lists final accuracy, absolute values, and sources, and plots the same contrasts (Figure 3).

(a) DiffClass, $n = 6$ per cell

Target − Control	5 tasks	10 tasks	20 tasks
SD1.5 − no generation, CIFAR-100	$+3.75\,[+3.10, +4.40]$	$+7.28\,[+6.48, +8.07]$	$+10.13\,[+9.03, +11.22]$
SD1.5 − no generation, ImageNet-100	$+4.61\,[+3.91, +5.30]$	$+5.70\,[+4.97, +6.43]$	$+6.11\,[+5.32, +6.91]$
SD1.5 − cGAN, CIFAR-100	$+2.97\,[+2.16, +3.78]$	$+4.03\,[+2.80, +5.27]$	$+3.15\,[+0.74, +5.55]$
SSIA on − off, CIFAR-100	$+0.32\,[+0.17, +0.47]$	$+0.03\,[-0.30, +0.37]$	$+0.17\,[-0.63, +0.98]$

(b) Other families

Family	Target − Control	Setting	Δ AA [95% CI]	n
DCS	reward-selected − no generation	CUB-200, 5-shot	$-0.87\,[-0.93, -0.82]$	6
	reward-selected − weak SD	CUB-200, 5-shot	$+0.01\,[-0.04, +0.06]$	6
SDDR	SD cache − noise cache	CIFAR-100, T10	$+1.84\,[+1.42, +2.26]$	6
	SD cache − noise cache	CIFAR-100, T25	$+2.09\,[+1.64, +2.55]$	6
Can Synthetic	Textual Inversion − no synthetic (prototype head)	CUB-200, 5-shot	$+0.97\,[+0.37, +1.56]$	6
	Textual Inversion (1k steps) − no synthetic	CUB-200, 5-shot	$+0.74\,[-0.13, +1.62]$	6
	Textual Inversion (3k steps) − no synthetic	CUB-200, 5-shot	$+0.17\,[-0.29, +0.62]$	6
DiffFR	calibrated diffusion − Gaussian statistics	CIFAR-100, T10	$+0.27\,[-0.04, +0.58]$	3
	calibrated diffusion − stored feature residuals	CIFAR-100, T10	$-3.88\,[-4.90, -2.85]$	3
Beyond	F_{syn} − repeated F_{inv}	CUB-200, 5-shot	$+0.29\,[-1.61, +2.18]$	6
	F_{aug} − repeated F_{inv}	CUB-200, 5-shot	$+1.74\,[-0.24, +3.71]$	6

sampler, and *weak SD* is generated by the same generator with the same prompts, classes, and image count using plain classifier-free guidance, without reward guidance or selection; reward-selected replay is the Target in both DCS contrasts. Thus reward-selected replay minus no generation measures the net effect of selected synthetic replay, and reward-selected replay minus weak SD measures what selection adds to an unselected pool. DiffClass *compute-matched no generation* repeats real current-task samples to match the replay pipeline's training rows and optimizer steps without synthetic images. SDDR's *noise cache* replaces the Stable Diffusion images by label-matched noise images with the same count and losses. DiffFR's controls replay per-class diagonal Gaussian samples or re-centered empirical residuals of stored features. Beyond's base view F_{inv} is the Stable Diffusion U-Net feature of a real image, read out after encoding the image into the diffusion latent space; F_{syn} and F_{aug} add views from synthesized and augmented images. Appendix B defines all remaining terms.

5 Results

5.1 Primary paired contrasts

Table 3 lists the primary contrast for each role together with its n.

DCS. Reward selection changes AA by less than 0.1 pp over the unselected pool of generated images (CI within $[-0.04, +0.06]$), and reward-selected replay stays below no generation in both AA and final accuracy. Table 4 gives the complete audit card for the selection contrast.

DiffClass. Stable Diffusion replay improves over no generation at every split on both datasets, with a gain that grows with the number of tasks on CIFAR-100, and at 50 images per class it stays above the strongest non-diffusion generator (cGAN). The SSIA selection branch adds at most 0.32 pp, enabling the MDA calibration branch lowers AA by 18.56 to 23.64 pp, and weaker non-diffusion generators (pixel statistics, cVAE) that replay the same old-class labels and counts stay within 0.3 pp of no generation (Appendix E).

SDDR. The Stable Diffusion cache beats a label-matched noise cache at both task lengths.

Can Synthetic. Textual-Inversion replay gives small gains over no synthetic replay, and only the prototype-head protocol has an interval above zero; miniImageNet results are in Appendix E.

Table 4: Audit card for the DCS selection contrast. Appendix C.1 gives the template.

Role	Selection or reward guidance: a reward model steers and reweights candidates during sampling (reward-guided SMC)
Target / Control	Reward-selected replay / weak SD: the same generator, prompts, and replay count with plain classifier-free guidance and no reward
Held fixed within a pair	Dataset and split, class order, seed, replay amount, downstream training
Residual confounders	Search and reward-model compute; composition of accepted samples
Repeated-run unit	Training seed, varied across pairs; $n = 6$ matched runs
Claim setting	CUB-200, 5-shot FSCIL, SD3.5-Medium generator, 30/30/10 replay images per class (base/current/previous sessions), one fixed cache
Result	AA $+0.01\,[-0.04,\ +0.06]$ pp (unresolved); final accuracy $-0.08\,[-0.14,\ -0.03]$ pp
Alternative control	Reward-guided vs. zero-reward sampling on 2 datasets $\times$ 2 shots $\times$ 3 budgets: CI above, below, or covering zero in 4, 2, and 6 of 12 settings (AA)
Conclusion	In this setting, reward selection changes AA by less than 0.1 pp over the unselected pool; the guidance effect depends on the dataset, shot, and budget

DiffFR. In the CIFAR-100 T10 protocol, calibrated diffusion feature replay matches Gaussian class-statistics replay and falls below replay of stored feature residuals (the residuals of all stored training features). The match with Gaussian statistics also holds in a separate cached-feature scaffold, in eight settings with six matched runs each (CIFAR-100, CUB-200, ImageNet-100, and TinyImageNet, each with 10 and 20 phases), where the mean AA difference ranges from -0.02 to $+0.17$ pp (Table 5; Appendix F).

Beyond. Against the primary control, a repeated base view F_{inv} that matches the rows, labels, views, and optimizer steps of the added view, neither added view shows a resolved full-sequence gain (upper CI bounds $+2.18$ and $+3.71$ pp), while F_{aug} keeps a base-session gain of $+2.77\,[+0.80,\ +4.74]$ pp. The small F_{syn} effect is consistent with the original paper's own ablation, where adding F_{syn} to F_{inv} changes CUB-200 base accuracy by $+0.3$ and AA by -0.5 pp. Against F_{inv} alone without exposure matching, both added views gain about 9 pp in base-session accuracy, which we report only as context (Table 5).

Accuracy versus seen classes. AA can hide how a contrast evolves as classes accumulate, so Figure 2 plots the image-replay contrasts against the number of seen classes, using the stored per-session accuracy of every run (Appendix H). The DiffClass SD1.5 replay gain is positive, with a CI above zero, at all 64 post-base sessions across the six dataset–split settings, whereas the compute-matched control covers zero at 53 of them. DCS reward-selected replay is below no generation at all 10 post-base sessions. The AA results therefore hold across the whole sequence.

5.2 Alternative controls

Table 5 adds, for each family, a control that fixes a factor its primary control leaves open. Three conclusions survive. The DiffClass replay gain does not come from extra optimization: a control repeating real samples to match rows and steps stays at no-generation accuracy. SD1.5 stays above cGAN with new caches at 50 images per class; at 10–20 per class the ordering reverses for 20 tasks. The DiffFR match with Gaussian statistics also holds in a separate cached-feature scaffold. The SDDR cache is positive, and replay alone and distillation alone each improve over using neither. Three conclusions change, narrow, or depend on the control. The DCS guidance effect, measured against a zero-reward sampler that shares the sampling path, prompts, and seeds, is positive in some dataset–shot–budget settings and negative in others (Table 6a). For Beyond, the conclusion depends on the control: against F_{inv} alone without exposure matching, run as context in a separate fixed-cache scaffold, both added views show base-session gains near $+9$ pp. The best synthetic-replay strategy in Can Synthetic and the best source in DiffClass depend on the dataset, shot, budget, and cache. Each conclusion therefore holds only with the controls under which it was tested. Appendix G lists, for each conclusion, whether it holds in one setting, replicates across datasets, or depends on the setting.

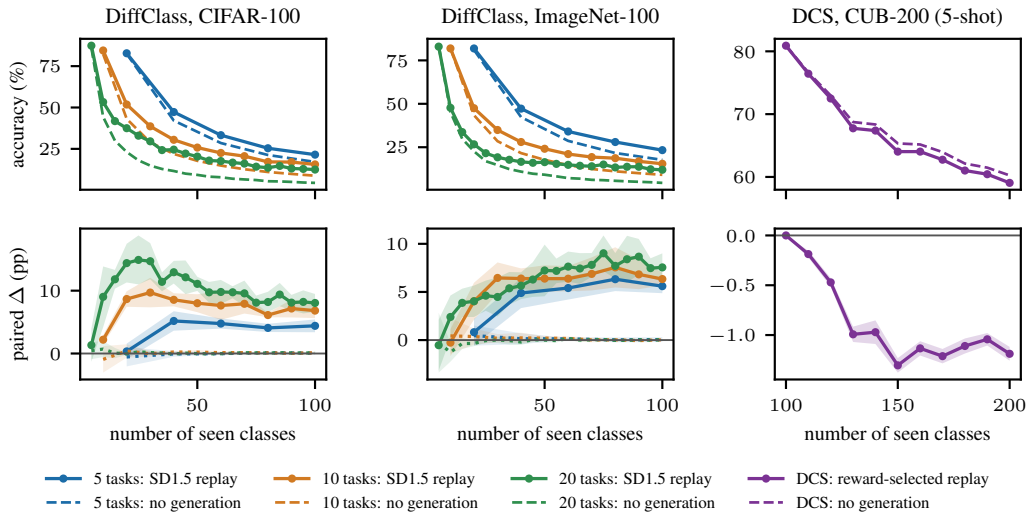

Figure 2: Accuracy versus number of seen classes. Top: seed-mean accuracy of the Target (solid) and the no-generation Control (dashed). Bottom: paired Target−Control difference with 95% CI bands; dotted lines in the DiffClass panels show compute-matched no generation minus no generation. Six seeds per setting.

Table 5: Alternative controls (condensed). Each row adds a control that fixes a factor the primary control leaves open; the last row drops exposure matching and is context only. Counts are of paired 95% CIs over AA and final accuracy where both are tested, and values are ΔAA (pp); six matched runs per setting. Table 16 in Appendix F gives the full designs and results.

Family	Alternative control	Result	Conclusion
DCS	Zero-reward sampling on the same path; 12 settings	CI above / below / covering zero: 4 / 2 / 6 (Table 6a)	Guidance effect depends on the setting
DiffClass	Compute-matched no generation	6/6 CIs cover zero	Gain is not extra optimization
	Random selection from the same pool	11/12 CIs cover zero	No stable selection advantage
	New caches: 4 sources, 10–50 images per class	SD1.5 above cGAN at 50/class; reversed at 10–20/class (20 tasks)	Source ranking depends on the setting
SDDR	Same images, deranged labels	3/4 CIs cover zero	Label effect detected at one endpoint only
	Replay only; distillation only	4/4 CIs above zero	Each alone improves over neither
Can Synth.	240-run factorial: dataset, shot, budget, source	Best strategy varies	No stable direction
DiffFR	Four datasets, 10 and 20 phases	vs. Gaussian statistics: -0.02 to $+0.17$ (8 setting means)	Matches class-statistics replay
Beyond	Second-view labels permuted	Permutation costs F_{inv} 16.0–19.4 more than $F_{\text{syn}}/F_{\text{aug}}$	Alignment dependence is source-specific
	Context: F_{inv} alone, no exposure matching	F_{syn} +9.24, F_{aug} +8.85 (base-session acc.)	Context only (separate fixed-cache scaffold)

5.3 Generator versions

With the DiffClass classifier and settings fixed, a Stable Diffusion 3.5 cache beats an SD1.5 cache in all nine split–budget settings, by $+1.55$ to $+3.42$ pp AA (Table 6b) and $+1.67$ to $+4.02$ pp final accuracy. Two separately built SD1.5 cache packages agree within noise (8/9 AA and 9/9 final CIs cover zero). Because the generator and its generation recipe change together, this measures sensitivity

Table 6: Two alternative controls whose results depend on the setting (ΔAA in pp, paired 95% CIs, six matched runs per cell). Green and orange mark CIs entirely above or below zero. (a) DCS reward-guided minus zero-reward sampling with q replay images per class. (b) DiffClass SD3.5 minus SD1.5 cache on CIFAR-100 under a fixed classifier. Appendix F gives final accuracy and absolute values.

(a) DCS guidance Setting	$q = 1$	$q = 3$	$q = 5$
CUB-200, 1-shot	$+0.08\,[-0.21,\ +0.37]$	$+0.36\,[+0.01,\ +0.71]$	$+0.44\,[+0.12,\ +0.76]$
CUB-200, 5-shot	$+0.21\,[-0.18,\ +0.60]$	$+0.60\,[+0.15,\ +1.04]$	$+0.89\,[+0.43,\ +1.36]$
miniImageNet, 1-shot	$-0.13\,[-0.21,\ -0.05]$	$-0.06\,[-0.11,\ +0.00]$	$-0.08\,[-0.12,\ -0.05]$
miniImageNet, 5-shot	$-0.05\,[-0.25,\ +0.15]$	$-0.05\,[-0.14,\ +0.03]$	$+0.01\,[-0.07,\ +0.09]$
(b) DiffClass generator Setting	$q = 10$	$q = 25$	$q = 50$
CIFAR-100, 5 tasks	$+1.55\,[+1.25,\ +1.85]$	$+1.62\,[+1.44,\ +1.81]$	$+1.65\,[+1.19,\ +2.11]$
CIFAR-100, 10 tasks	$+2.22\,[+1.47,\ +2.97]$	$+2.27\,[+1.44,\ +3.10]$	$+2.81\,[+2.36,\ +3.26]$
CIFAR-100, 20 tasks	$+3.42\,[+2.79,\ +4.06]$	$+2.79\,[+2.01,\ +3.57]$	$+3.40\,[+2.79,\ +4.01]$

to the generator package; conclusions of the form "diffusion replay helps" are therefore conditional on the generator.

6 Limitations

- **Family-local design.** Each family is audited in its own scaffold, so the audit identifies what supports a gain inside that pipeline, not whether the conclusion transfers to other datasets, backbones, or generators. A unified modular benchmark that runs every role on the same datasets and generators (Baek et al., 2019) would test that transfer; we ran such a shared scaffold only for DCS and SDDR, whose ordering changes with the dataset (Appendix F.1).
- **Residual confounding.** Controls are local counterfactuals. Offline generation, storage, and reward-model compute are generally not matched, and some added controls change more than one factor and are reported as package contrasts (Appendix D).
- **Randomness.** Most CIs cover downstream seeds and class orders with fixed caches. Cache regeneration enters the CIs of the DCS replay-budget grid, where every seed generates its own cache, and is probed for DiffClass through pool partitions and separately built SD1.5 caches.
- **Task scope.** All results concern incremental image classification; detection, segmentation, or federated settings would need new roles and controls.

7 Conclusion

Paired, role-specific controls change how diffusion-assisted incremental-recognition results should be read. Within the evaluated settings, Stable Diffusion image replay in DiffClass supports a gain that survives compute matching and a second dataset, whereas DCS reward selection changes accuracy by less than 0.1 points over its unselected pool, DiffFR's diffusion feature replay is matched by class-statistics replay, and Beyond's added feature views show no resolved full-sequence gain against an exposure-matched base view; several conclusions also change with the dataset, generator, or replay budget. We recommend that such papers compare each diffusion component against its simplest substitute, report alternative substitutes separately from reference results, and state each claim only for the datasets, generators, and budgets in which it was tested, together with repeated-run uncertainty. Future work includes a unified multi-dataset, multi-generator benchmark that tests whether these conclusions transfer across pipelines, and extending the roles and controls to detection, segmentation, and federated continual learning.

AI USE STATEMENT

In this work, we used generative AI tools for grammar and sentence polishing and for literature survey. We have reviewed all AI-assisted content and take responsibility for the final content of this work.

ETHICS STATEMENT

This work evaluates existing methods on public benchmarks and releases no new generative model or dataset. No human subjects were involved.

REPRODUCIBILITY STATEMENT

Section 3 and Appendix C define the audit procedure, role codebook, and audit card; Section 4 and Appendices B and D specify datasets, splits, controls, and held-fixed factors. Appendices E–H list every reported contrast with its n, CI, and evidence source. Appendix A states which results can be rerun from the code and run scripts in the anonymous repository (`https://anonymous.4open.science/r/dca-iclr27-F078`) and which added analyses are specified only in the appendices.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves ImageNet classification. *Transactions on Machine Learning Research*, 2023.

Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4715–4723, 2019.

Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 2020. doi: 10.1016/j.neunet.2020.12.003. arXiv:2011.01844.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Naz Sepah, Edward Raff, Kanika Madan, Vikram Voleti, et al. Accounting for variance in machine learning benchmarks. In *Proceedings of Machine Learning and Systems*, volume 3, 2021.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium*, 2023.

Jingfan Chen, Yuxi Wang, Pengfei Wang, Xiao Chen, Zhaoxiang Zhang, Zhen Lei, and Qing Li. DiffusePast: Diffusion-based generative replay for class incremental semantic segmentation, 2023.

Bartosz Cywiński, Kamil Deja, Tomasz Trzciński, Bartlomiej Twardowski, and Lukasz Kuciński. GUIDE: Guidance-based incremental learning with diffusion models. In *European Conference on Artificial Intelligence*, 2024. arXiv:2403.03938.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, 2024.

Emanuele Frascaroli, Aniello Panariello, Pietro Buzzega, Lorenzo Bonicelli, Angelo Porrello, and Simone Calderara. CLIP with generative latent replay: A strong baseline for incremental learning. In *Proceedings of the British Machine Vision Conference*, 2024.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *International Conference on Learning Representations*, 2023.

Rui Gao and Weiwei Liu. DDGR: Continual learning with deep diffusion-based generative replay. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10744–10763. PMLR, 2023.

Scott Geng, Cheng-Yu Hsieh, Vivek Ramanujan, Matthew Wallingford, Chun-Liang Li, Pang Wei Koh, and Ranjay Krishna. The unmet promise of synthetic training images: Using retrieved real images performs better. In *Advances in Neural Information Processing Systems*, 2024.

Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost van de Weijer. FeCAM: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. In *Advances in Neural Information Processing Systems*, 2023.

Tyler L. Hayes and Christopher Kanan. Lifelong machine learning with deep streaming linear discriminant analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *International Conference on Learning Representations*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

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*, pp. 831–839, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Zitong Huang, Ze Chen, Zhixing Chen, Erjin Zhou, Xinxing Xu, Rick Siow Mong Goh, Yong Liu, Chun-Mei Feng, and Wangmeng Zuo. Learning prompt with distribution-based feature replay for few-shot class-incremental learning, 2024.

Quentin Jodelet, Xin Liu, Yin Jun Phua, and Tsuyoshi Murata. Class-incremental learning using diffusion model for distillation and replay. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 3417–3425, 2023. arXiv:2306.17560.

Quentin Jodelet, Xin Liu, Yin Jun Phua, and Tsuyoshi Murata. Memory augmented using diffusion model for class-incremental learning. *Image and Vision Computing*, 2025. doi: 10.1016/j.imavis.2025.105600.

Haidong Kang, Ketong Qian, and Yi Lu. Breaking forgetting: Training-free few-shot class-incremental learning via conditional diffusion, 2025.

Junsu Kim, Hoseong Cho, Jihyeon Kim, Yihalem Yimolal Tiruneh, and Seungryul Baek. SDDGR: Stable diffusion-based deep generative replay for class incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2024.

Junsu Kim, Yunhoe Ku, and Seungryul Baek. Can synthetic images conquer forgetting? beyond unexplored doubts in few-shot class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2025a. arXiv:2507.13739.

Junsu Kim, Yunhoe Ku, Dongyoon Han, and Seungryul Baek. Beyond synthetic replays: Turning diffusion features into few-shot class-incremental learning knowledge, 2025b. arXiv:2503.23402.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Linhao Li, Yongzhang Tan, Siyuan Yang, Hao Cheng, Yongfeng Dong, and Liang Yang. Adaptive decision boundary for few-shot class-incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 18359–18367, 2025.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *Computer Vision – ECCV 2016*, pp. 614–629. Springer, 2016.

Jinglin Liang, Jin Zhong, Hanlin Gu, Zhongqi Lu, Xingxing Tang, Gang Dai, Shuangping Huang, Lixin Fan, and Qiang Yang. Diffusion-driven data replay: A novel approach to combat forgetting in federated class continual learning. In *Computer Vision – ECCV 2024*. Springer, 2024. arXiv:2409.01128.

Huan Liu, Li Gu, Zhixiang Chi, Yang Wang, Yuanhao Yu, Jun Chen, and Jin Tang. Few-shot class-incremental learning via entropy-regularized data-free replay. In *Computer Vision – ECCV 2022*, pp. 146–162. Springer, 2022.

Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. Class-incremental learning: Survey and performance evaluation on image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. doi: 10.1109/TPAMI.2022.3213473. arXiv:2010.15277.

Yongsheng Mei, Liangqi Yuan, Dong-Jun Han, Kevin S. Chan, Christopher G. Brinton, and Tian Lan. Using diffusion models as generative replay in continual federated learning – what will happen?, 2024.

Zichong Meng, Jie Zhang, Changdi Yang, Zheng Zhan, Pu Zhao, and Yanzhi Wang. Diffclass: Diffusion-based class incremental learning. In *Computer Vision – ECCV 2024*. Springer, 2024. arXiv:2403.05016.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Computer Vision – ECCV 2020*, pp. 681–699. Springer, 2020.

Can Peng, Kun Zhao, Tianren Wang, Meng Li, and Brian C. Lovell. Few-shot class-incremental learning from an open-set perspective. In *Computer Vision – ECCV 2022*, pp. 382–397. Springer, 2022.

Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. FeTrIL: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.

Ameya Prabhu, Philip H. S. Torr, and Puneet K. Dokania. GDumb: A simple approach that questions our progress in continual learning. In *Computer Vision – ECCV 2020*, 2020.

Ameya Prabhu, Hasan Abed Al Kader Hammoud, Puneet K. Dokania, Philip H. S. Torr, Ser-Nam Lim, Bernard Ghanem, and Adel Bibi. Computationally budgeted continual learning: What does matter? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

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*, pp. 2001–2010, 2017.

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*, pp. 10684–10695, 2022.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22500–22510, 2023.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Mert Bulent Sariyildiz, 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*, 2023.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Paweł Skierś and Kamil Deja. Joint diffusion models in continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

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*, pp. 9374–9384, 2021.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12183–12192, 2020.

Song Tian, Lusi Li, Weijun Li, Hang Ran, Xiaoyan Ning, and Prayag Tiwari. A survey on few-shot class-incremental learning. *Neural Networks*, 2023. arXiv:2304.08130.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *International Conference on Learning Representations*, 2024.

Gido M. van de Ven, Hava T. Siegelmann, and Andreas S. Tolias. Brain-inspired replay for continual learning with artificial neural networks. *Nature Communications*, 11:4069, 2020.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. FOSTER: Feature boosting and compression for class-incremental learning. In *Computer Vision – ECCV 2022*, pp. 398–414. Springer, 2022.

Ruitao Wu, Yifan Zhao, Guangyao Chen, and Jia Li. Diffusion-classifier synergy: Reward-aligned learning via mutual boosting loop for FSCIL. In *Advances in Neural Information Processing Systems*, pp. 64229–64256. Curran Associates, Inc., 2025. arXiv:2510.03608.

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 DeepInversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12455–12464, 2021.

Jichuan Zhang, Yali Li, Xin Liu, and Shengjin Wang. Diffusion model meets non-exemplar class-incremental learning and beyond, 2024. arXiv:2408.02983.

Jinghua Zhang, Li Liu, Olli Silvén, Matti Pietikäinen, and Dewen Hu. Few-shot class-incremental learning for classification and object detection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. doi: 10.1109/TPAMI.2025.3529038. arXiv:2308.06764.

Da-Wei Zhou, Qiwen Wang, Zhiyuan Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Class-incremental learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. doi: 10.1109/TPAMI.2024.3429383. arXiv:2302.03648.

Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

A Release, Artifact Map, and Rerun Requirements

The anonymous repository contains the family runners with the options needed for the primary contrasts, run scripts, per-family method cards, the audit schema, and the cache-quality diagnostic of Table 22; it is available at `https://anonymous.4open.science/r/dca-iclr27-F078`. Reported numbers are given in this paper and are not duplicated in the repository. It also contains the entry points and aggregation modules of the added controls, the common-head block, the direct overlap screen, the per-session curve analysis, and the plotting scripts for Figures 2 and 3; its README maps each group of results to these files. Cluster submission wrappers and per-run outputs are not included, and the designs are specified in Appendices D–I. The role codebook, family assignments, and audit card are given in full in Appendix C. Table 7 lists, for each group of results, whether it can be rerun from the repository. The repository does not redistribute raw datasets, generated caches, feature tensors, or pretrained checkpoints; reruns require CIFAR-100, CUB-200-2011, ImageNet (for ImageNet-100, miniImageNet, and TinyImageNet protocols), Stable Diffusion 1.5 and 3.5-Medium, and the CLIP, DINOv2, and ResNet-18 checkpoints where applicable.

Table 7: Map from reported results to the anonymous repository.

Results	Where reported	In the repository
Primary contrasts of the six families	Table 3; Figure 3; Tables 14, 15	Runners and run scripts for every family (DCS generation, reward, and training; DiffClass with no-generation variants and cGAN/cVAE caches; SDDR with SD and noise caches; Can Synthetic with class-name and Textual-Inversion modes; DiffFR with Gaussian and residual replay; Beyond feature runner); repeated-run aggregation not included
DiffClass compute-matched control	Table 5	Yes (runner variant)
DCS replay-budget grid; DiffClass SSIA vs. random, source/budget grid, pool partitions, source packages; SDDR derangement and component arms; Can Synthetic factorial; DiffFR four-dataset launchers; Beyond exposure and alignment controls	Tables 16, 17, 18	Entry points and aggregation modules; cluster submission wrappers are not included
Common-head block and random-initialization anchor	Table 19	Yes (feature-cache builder and common-head runner)
Direct overlap screen	Appendix I	Yes
Cache-quality diagnostics	Table 22	Yes
Per-session curves; Figures 3, 2	Section 5.1	Yes (analysis and plotting scripts)
Role codebook, assignments, audit card	Appendix C	Given in full in this paper

B Glossary

Table 8: Terms and abbreviations.

Scope	Term	Meaning in this paper
Setting	CIL / FSCIL	Class-incremental learning / few-shot class-incremental learning.
Metric	AA	Average over sessions of accuracy on all classes seen so far.
Metric	final	Accuracy after the last session.
Metric	base acc.	Accuracy after the base session only.
Split	k tasks (DiffClass)	CIFAR-100 or ImageNet-100 divided into $k \in \{5, 10, 20\}$ tasks of $100/k$ classes, with 50 replay images per class unless stated.
Split	T10 / T25 (SDDR)	50 base classes followed by 10 tasks of 5 classes or 25 tasks of 2 classes.
Split	T5 / T10 / T20 (DiffFR)	5, 10, or 20 incremental phases on CIFAR-100.
Split	q	Replay images per class. For DCS the base/current/previous-session allocation is $3q/3q/q$.
Audit	Target / Control	The two arms of a contrast; Δ is always Target minus Control, in percentage points (pp).
Audit	n	Number of matched run pairs (same seed, class order, and split in both arms).
Audit	Tier A / B / C	Evidence coverage recorded in the ledger (Appendix E).
Audit	P / S / C controls	Primary, alternative (sensitivity), and context controls of Section 3.2.
Audit	package contrast	A contrast whose intervention changes more than one factor; read as the combined effect.
Image replay	no generation	The same scaffold trained without synthetic images.
Image replay	compute-matched no generation	No synthetic images; real current-task samples are repeated to match the replay arm's rows and optimizer steps.
Image replay	cache	A fixed set of generated images or features loaded during training.
Image replay	source package	A cache together with the generator, prompts, and materialization recipe that produced it.

Scope	Term	Meaning in this paper
DCS	reward-selected replay	Replay pool after reward-based reranking and selection; the Target of both DCS contrasts (previously also called generated replay).
DCS	weak SD	Images from the same generator, prompts, classes, and count, generated with plain classifier-free guidance and no reward guidance or selection.
DCS	guided / zero-reward	Reward-guided sampling versus the same sampler with a zero reward; the zero-reward arm does not run the reward model.
DiffClass	SD1.5 replay	Replay from a Stable Diffusion 1.5 cache ("vanilla" generation).
DiffClass	SSIA	Selective Synthetic Image Augmentation, DiffClass's candidate-selection branch.
DiffClass	MDA	Multi-Domain Adaptation, a gradient-reversal domain classifier aligning real and synthetic images.
DiffClass	pixel-stat, cVAE, cGAN	Non-diffusion image sources: per-class pixel statistics, a conditional VAE, and a conditional GAN.
DiffClass	random selection	Uniform choice from the same 100-candidate pool used by SSIA, with the score still computed.
SDDR	SD cache / noise cache	Stable Diffusion replay images / label-matched noise images with the same count and losses.
SDDR	deranged labels	The same images with labels mapped by $c \mapsto (c+1) \bmod K$ within the K seen classes.
Can Synthetic	TI / class-name	Textual-Inversion conditioning / fixed class-name prompts for the same generator.
Can Synthetic	prototype head	The classifier-head mode used in the prototype-head CUB-200 protocol.
Can Synthetic	real (label-matched)	Stored real images matched on labels, rows, and steps.
DiffFR	PC	Prototype calibration applied to diffusion-generated features.
DiffFR	LA+SIM	DiffFR variant with rotation label augmentation and similarity-based self-supervision.
DiffFR	Gaussian / residual	Replay sampled from per-class diagonal Gaussians / from re-centered and re-scaled empirical residuals of stored features.
Beyond	F_{inv}	Stable Diffusion U-Net feature of a real image obtained through the inversion path; the base view.
Beyond	F_{syn} / F_{aug}	Additional feature views from synthesized / augmented images.
Beyond	repeated F_{inv}	F_{inv} repeated to match the rows, labels, views, and steps of the F_{syn}/F_{aug} arms.
Representation	common head	Cosine nearest-class-mean classifier applied to frozen features with identical support and evaluation data.

C Role Codebook, Family Assignments, and Audit Card

Global rules. (G1) Trace every executable path from a diffusion model, a diffusion-derived cache, or a diffusion-network activation to the recognition loss or prediction; assign roles from that path, never from the method name or an observed accuracy difference. (G2) Assign a role if and only if all its conditions hold for at least one audited branch; a family may receive several roles. (G3) Use evidence in the order executable code and configuration, algorithm or pseudocode, method prose. (G4) Aggregate branch decisions by logical OR at the family level and record the positive branch and its evidence anchor. (G5) If evidence is ambiguous, record the best binary decision and set an uncertainty flag with an explanation. (G6) The matched control is the smallest intervention that removes or replaces the assigned role while retaining split, class order, evaluator, classifier path, row and label budget, and unrelated optimization schedule whenever feasible; unavoidable accompanying changes are residual confounders, not new roles.

Table 9: Full role codebook. All conditions in the second column must hold.

Role	Conditions (all required)	Excluded cases	Residual confounders of the minimal control
Pixel-image replay or augmentation	(i) A diffusion generator emits decoded pixel images; (ii) the images enter the labeled support, augmentation, or replay set in at least one session; (iii) they can be removed or replaced without redefining the task.	Images used only for inspection; diffusion activations used as features; feature-space synthesis without decoded pixels.	Sample count, class balance, source quality, optimizer steps, replay schedule.
Diffusion feature replay	(i) A learned diffusion or denoising process synthesizes feature vectors or a feature distribution; (ii) the synthetic features are labeled replay in updates across sessions; (iii) stored empirical statistics with the same dimension and rows can replace them.	Activations of real images from a frozen diffusion network; concurrent views used only as augmentation; statistics replay without a diffusion process.	Feature geometry, calibration, capacity of the stored statistics, sampling compute.

Role	Conditions (all required)	Excluded cases	Residual confounders of the minimal control
Diffusion-network representation	(i) Real images are encoded by a diffusion model or its U-Net/encoder; (ii) the activation enters the classifier, prototype, or prediction path; (iii) replacing the encoder with the head and protocol fixed defines the contrast.	Generated images encoded by an ordinary backbone; feature-space diffusion replay; offline caches that never supply recognition features.	Pretraining data, capacity, input resolution, feature dimension, preprocessing.
Selection or reward guidance	(i) Multiple diffusion candidates or sampling trajectories exist per class; (ii) a candidate-specific score ranks, accepts, rejects, reweights, or steers them before the recognition update; (iii) scoring does not persistently update generator state.	Fixed random subsampling for a budget; content-independent class balancing; persistent prompt or LoRA training without selection.	Reward-model compute, trajectory dynamics, pool quality, accepted count, selection-induced class balance.
Generator or conditioning adaptation	(i) Task examples optimize persistent generator or conditioning parameters before the audited outputs are produced; (ii) the adapted state generates class-specific outputs; (iii) fixed class-name conditioning is a meaningful control.	Fixed class-name prompts; scoring with a frozen generator; calibration only inside the recognizer.	Adaptation compute, prompt expressivity, number of support examples, sampling randomness.
Latent or diffusion-feature augmentation	(i) An additional latent or diffusion-feature view is created from an existing example or representation; (ii) it is added concurrently to training or prototype construction rather than serving as replay of past classes; (iii) the base view alone is a meaningful control.	The base diffusion representation itself; replay of old-class features; decoded pixel augmentation.	Extra training rows, view correlation, augmentation compute, aggregation rule.
Synthetic-to-real calibration	(i) A module, loss, or transformation uses real-versus-synthetic identity or a domain-discrepancy objective; (ii) it changes recognizer or classifier optimization to align the domains; (iii) the replay source can be held fixed while it is disabled.	Generic normalization, balancing, or distillation; prototype calibration without domain identity; changing the generator.	Auxiliary-loss weight, extra optimization, domain-label balance, gradient interaction.

Non-role axes. Generic distillation or replay regularization, generator identity and output quality, replay quantity and schedule, optimization compute, and real-replay oracles are recorded as matching variables, sensitivity axes, or context rows. An observed positive or negative accuracy difference is never used to assign a role.

Family assignments. Table 10 lists the roles assigned to the six families by applying the codebook to the audited branches. These assignments were made by the authors; independent multi-annotator agreement was not measured. To repeat the assignment, an annotator fills one row per family and role with the binary assignment, an uncertainty flag, the evidence anchor, and a one-sentence note stating which conditions hold.

Table 10: Role assignments of the audited families (multi-label). Pix: pixel-image replay; Feat: diffusion feature replay; Repr: diffusion-network representation; Sel: selection or reward guidance; Adapt: conditioning adaptation; Aug: latent/diffusion-feature augmentation; Cal: synthetic-to-real calibration. DiffClass is not assigned Adapt: the audited branches use a frozen SD1.5 cache, and the method's MDM LoRA fine-tuning of the generator, which would satisfy the Adapt conditions, is outside the audited branches.

Family	Pix	Feat	Repr	Sel	Adapt	Aug	Cal	Positive branch
DCS	✓			✓				SD3.5 replay cache; reward-based reranking/guidance of candidates
DiffClass	✓			✓			✓	SD1.5 replay cache; SSIA selection; MDA domain classifier
SDDR	✓							SD1.5 replay cache (distillation is a non-role axis)
Can Synthetic	✓				✓			SD1.5 replay cache; Textual-Inversion token optimization
DiffFR		✓						Feature-space diffusion replay (prototype calibration is not domain calibration)
Beyond			✓			✓		SD1.5 U-Net features of real images; added $F_{\mathrm{syn}}/F_{\mathrm{aug}}$ views

C.1 Audit card and a worked example

An audit card has the following fields. *Role and evidence anchor*: the assigned role, the code or algorithm location that supports it, and any uncertainty flag. *Comparison*: Target, Control (P, and

Table 11: Role codebook with all controls (full version of Table 1). A role is assigned when its conditions hold for at least one method branch. P: primary control for the named effect; S: alternative control (Section 3.2) that tests whether the primary conclusion is stable; C: context result that gives scale without identifying the role. Table 9 gives full conditions and residual confounders.

Role	Assign if (dataflow condition)	Primary control (P)	Sensitivity (S) / context (C)
Pixel-image replay or augmentation	Decoded generated images enter labeled training or replay in at least one session	Same pipeline without generated images	S: exposure-matched real repeats, noise cache, other generator. C: real replay
Diffusion feature replay	A learned diffusion process synthesizes feature vectors used as labeled replay	P1: class-wise Gaussian replay; P2: empirical-residual replay (same labels, rows, dimension, head)	S: calibration on/off. C: stored real features
Diffusion-network representation	Diffusion-network activations of real images enter the prediction path	None inside the family without matching architecture and pretraining	C: other encoders under an identical head
Selection or reward guidance	A candidate-specific score filters, ranks, reweights, or steers generated candidates	Same candidates and count without scoring (random choice or unscored pool); for guidance, zero applied guidance with the guidance computation still executed	S: zero guidance without running the scorer (package contrast), permuted score target, budget grid. C: no generation
Generator or conditioning adaptation	Task data optimize persistent generator or conditioning state (e.g., Textual Inversion)	Fixed class-name conditioning, same generator and budget	S: other fixed conditioning. C: no synthetic replay
Latent or diffusion-feature augmentation	An extra latent or diffusion-feature view is added concurrently to training	Same base view with rows and optimizer steps matched	S: fixed-cache label permutation. C: base view without exposure matching; non-diffusion encoders
Synthetic-to-real calibration	A module or loss aligns domains using real-versus-synthetic identity	Module disabled; rows, cache, classifier, and schedule fixed	S: vary strength. C: no generation

Table 12: Audited method families (full version of Table 2). Roles follow Table 1. The diffusion source is the model used in our audit. Residual confounders are factors that the primary control does not remove; Table 5 and Appendix D report the controls that probe them.

Family	Regime and data	Diffusion source	Roles	Primary control(s)	Main residual confounders
DCS (Wu et al., 2025)	FSCIL, CUB-200 5-shot; added: CUB-200 1-shot, miniImageNet 1/5-shot	SD3.5-Medium (Esser et al., 2024)	pixel replay; reward selection	no generation; weak SD (plain generation, no reward)	reward-model and search compute; composition of accepted samples
DiffClass (Meng et al., 2024)	CIL, CIFAR-100, ImageNet-100 (5/10/20 tasks)	frozen SD1.5 cache; MDM LoRA fine-tuning is not audited	pixel replay; selection (SSIA); calibration (MDA)	no generation; SSIA off; MDA off	extra replay rows and steps; generator recipe
SDDR (Jodelet et al., 2023)	CIL, CIFAR-100, 50 base classes, T10/T25	SD1.5; the original method used SD1.4	pixel replay (distillation is a non-role axis)	label-matched noise cache	image semantics vs. statistics; interaction with distillation
Can Synthetic (Kim et al., 2025a)	FSCIL, CUB-200, miniImageNet, 5-shot	SD1.5 with Textual Inversion	pixel replay; conditioning adaptation	no synthetic replay; class-name conditioning	extra replay rows; inversion compute
DiffFR (Zhang et al., 2024)	CIL, CIFAR-100, T5/T10/T20 phases	feature diffusion trained on task features	feature replay	P1 Gaussian and P2 residual class-statistics replay	diffusion training compute and storage; feature calibration
Beyond (Kim et al., 2025b)	FSCIL, CUB-200 5-shot	SD1.5 U-Net features	diffusion representation; feature augmentation	base view F_{inv} repeated to match rows, labels, views, and steps	cache construction; fixed caches

S/C where used), factors held fixed, factors changed, and residual confounders. *Repeated-run unit*: training seed, class order, or independent cache draw, and the number of matched runs n. *Claim settings*: dataset and split, class order, model, generator and version, replay and compute budgets. *Statistical report*: absolute results of both arms, paired differences, paired 95% CIs, and tier. *Stability and claim*: the positive/negative/unresolved label of each declared setting and the conclusion the evidence supports.

Worked example: GUIDE. To show how the protocol applies beyond the six audited families, we fill the card for GUIDE (Cywiński et al., 2024) from its published method description; no result is implied. *Dataflow and roles*: the current classifier steers the sampling trajectory of the previous-task diffusion model, and the resulting old-class images are replayed; roles are pixel-image replay and selection or reward guidance (trajectory guidance). *Question and P for guidance*: does classifier guidance improve the replay content? Keep the diffusion and classifier checkpoints, initial noise, sampler, labels, image count, downstream replay exposure, optimizer, and evaluator fixed, and run

the same guidance path with the applied guidance vector set to zero. *S and C*: permute the guidance target at the same generation cost and vary the guidance scale (S); no replay and real replay give net-replay and scale context (C) and do not substitute for P. *Repeats and report*: vary generation and downstream training seeds jointly, declare dataset, split, class order, and replay budget, and report guided-minus-P accuracy and forgetting with paired intervals. *Claim boundary*: a positive P contrast would support a contribution of classifier guidance within GUIDE; the replay role is assessed separately through its own P (same pipeline without generated images).

D Control Isolation by Family

Table 13 states, for each family, what the primary control holds fixed, what it changes, which confounders remain, and which added controls address them.

Table 13: Control isolation cards.

Family	Held fixed	Changed	Residual confounders	Added controls (Table 5)
DCS	Dataset and split, class order, seed, declared replay amount, downstream training	Replay content: reward-guided vs. plain generation; with vs. without generated replay	Search and reward-model compute; exact downstream rows and generator calls were not logged for the local pair	Guided vs. zero-reward grid (same generator, prompt, image noise, labels, count); the zero-reward arm omits reward-model compute, so it is a package contrast
DiffClass	Classifier scaffold, seed, class order, split, epochs; for SSIA vs. random also the candidate pool, score computation, retained count, rows, labels, and steps	Presence of replay rows (no generation); cache identity and source recipe (source controls); retained image identity (selection)	Extra rows and exposure; generator, prompt, and materialization recipe differ between sources; one cache realization per source; MDM fine-tuning outside the audited branches	Compute-matched control; SSIA vs. random; source $\times$ budget grid; five pool partitions; three source packages
SDDR	Dataset and split, replay budget, SDDR losses, model, optimizer, seed; for derangement also the 5,000 images, per-session rows and order, and label histogram	Image content and label semantics (SD vs. noise); only the image–label assignment (derangement)	Offline generation compute; one cyclic derangement; component controls change the loss package	Same-image derangement; replay-only and distillation-only arms
Can Synthetic	Dataset and split, shot, class order, seed, head, schedule, cache budget	Use of the cache and its exposure (vs. no synthetic); conditioning and cache identity (TI vs. class-name)	Synthetic vs. no synthetic is not row-matched; Textual Inversion adds offline optimization; caches materialized once	240-run factorial including label-matched real replay
DiffFR	Frozen extractor, seed, class order, replay count per class, head, schedule	Feature source and distribution	Diffusion training compute, storage, and capacity are not matched	Four-dataset replication; raw vs. calibrated diffusion features
Beyond	CUB-200 split, shot, seed, frozen SD backbone, classifier, evaluation; for the primary control also rows, labels, views, and optimizer steps (repeated F_{inv})	Identity of the added view	Cache construction compute; fixed caches; alignment tested only on base-class second views	Base view alone without exposure matching (context); fixed-cache label permutation
Common head	Cosine NCM head, support and evaluation sets, sessions, seeds	Encoder	Architecture, capacity, objective, and pretraining data differ (except supervised vs. random ResNet-18, which differ only in weights; one fixed random initialization)	Random-initialized ResNet-18 anchor

E Detailed Contribution Ledger

Figure 3 plots the primary contrasts of Table 3, and Tables 14 and 15 expand them. "T / C" gives the mean of each arm over the matched runs when archived; "–" means only paired differences are archived. Intervals are paired two-sided Student-t 95% CIs; rows without an interval are single runs or have no archived per-run values for the control. The Tier column records evidence coverage: A, at least six matched run pairs over the full incremental sequence; B, fewer matched pairs, or a contrast whose per-run values are not archived so that no paired interval can be computed; C, a contrast that covers only part of the protocol (for example, base-session accuracy with fixed feature caches) or a deterministic diagnostic without a repeated-run unit.

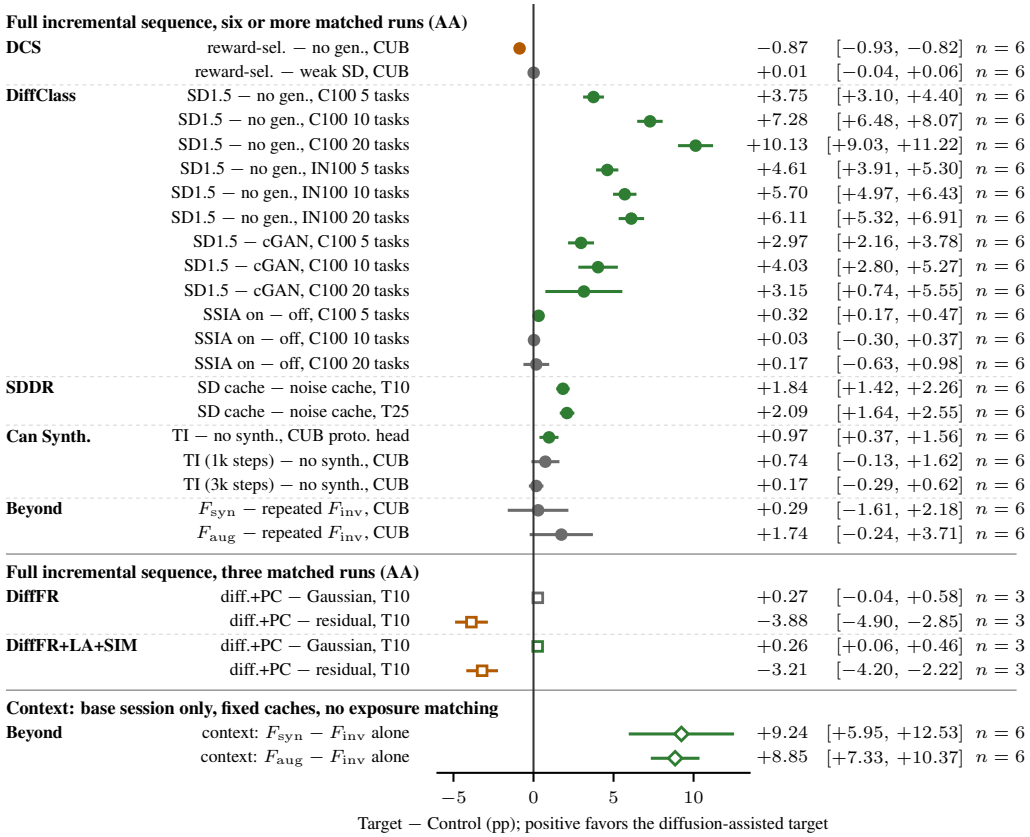

Figure 3: Primary Target−Control contrasts of Table 3, grouped by protocol coverage and number of matched runs n. Markers show paired means and bars show paired 95% CIs; green and orange mark CIs entirely above or below zero, gray marks CIs that cover zero. C100 and IN100 denote CIFAR-100 and ImageNet-100. Context rows cover only part of the protocol. The DiffClass MDA contrasts (-23.64 to -18.56 pp) fall outside the axis. Appendix E lists absolute values, final-accuracy contrasts, and sources for every row.

Table 14: Image-level ledger.

Family	Target − Control	Setting	Metric	T / C	Δ [95% CI]	n	Tier
DCS	reward-selected − weak SD	CUB 5-shot	AA	–	+0.01 [−0.04, +0.06]	6	A
	reward-selected − weak SD	CUB 5-shot	final	–	−0.08 [−0.14, −0.03]	6	A
	reward-selected − no generation	CUB 5-shot	AA	66.93 / 67.81	−0.87 [−0.93, −0.82]	6	A
	reward-selected − no generation	CUB 5-shot	final	59.06 / 60.24	−1.19 [−1.26, −1.12]	6	A
DiffClass	SD1.5 − no generation	C100, 5 tasks	AA	42.04 / 38.29	+3.75 [+3.10, +4.40]	6	A
	SD1.5 − no generation	C100, 10 tasks	AA	32.40 / 25.13	+7.28 [+6.48, +8.07]	6	A
	SD1.5 − no generation	C100, 20 tasks	AA	26.16 / 16.03	+10.13 [+9.03, +11.22]	6	A
	SD1.5 − no generation	C100, 5/10/20	final	21.50/17.10; 15.60/8.78; 12.48/4.44	+4.40 [+3.38, +5.42]; +6.83 [+5.64, +8.01]; +8.04 [+6.61, +9.48]	6	A
	SD1.5 − no generation	IN100, 5 tasks	AA	42.89 / 38.29	+4.61 [+3.91, +5.30]	6	A
	SD1.5 − no generation	IN100, 10 tasks	AA	30.78 / 25.08	+5.70 [+4.97, +6.43]	6	A
	SD1.5 − no generation	IN100, 20 tasks	AA	21.87 / 15.76	+6.11 [+5.32, +6.91]	6	A
	SD1.5 − no generation	IN100, 5/10/20	final	23.32/17.72; 15.46/9.12; 12.02/4.47	+5.60 [+4.87, +6.33]; +6.34 [+5.67, +7.01]; +7.55 [+6.09, +9.01]	6	A
	compute-matched − no generation	C100, 5/10/20	AA	–	−0.09 [−0.40, +0.23]; +0.04 [−0.30, +0.37]; +0.11 [−0.03, +0.26]	6	A

Family	Target − Control	Setting	Metric	T / C	Δ [95% CI]	n	Tier
	compute-matched − no generation	IN100, 5/10/20	AA	–	$+0.16\,[-0.12,\ +0.45]$; $+0.16\,[-0.02,\ +0.34]$; $-0.11\,[-0.25,\ +0.02]$	6	A
	SD1.5 − cGAN	C100, 5 tasks	AA	42.04 / 39.07	$+2.97\,[+2.16,\ +3.78]$	6	A
	SD1.5 − cGAN	C100, 10 tasks	AA	32.40 / 28.37	$+4.03\,[+2.80,\ +5.27]$	6	A
	SD1.5 − cGAN	C100, 20 tasks	AA	26.16 / 23.01	$+3.15\,[+0.74,\ +5.55]$	6	A
	SSIA on − off	C100, 5 tasks	AA	42.36 / 42.04	$+0.32\,[+0.17,\ +0.47]$	6	A
	SSIA on − off	C100, 10 tasks	AA	32.44 / 32.40	$+0.03\,[-0.30,\ +0.37]$	6	A
	SSIA on − off	C100, 20 tasks	AA	26.33 / 26.16	$+0.17\,[-0.63,\ +0.98]$	6	A
	MDA on − off	C100, 5/10/20	AA	18.40/42.04; 10.79/32.40; 7.60/26.16	$-23.64\,[-24.79,\ -22.50]$; $-21.61\,[-23.03,\ -20.20]$; $-18.56\,[-19.41,\ -17.70]$	6	A
	MDA on − off	C100, 5/10/20	final	–	$-20.38\,[-21.53,\ -19.23]$; $-14.26\,[-15.79,\ -12.72]$; $-11.48\,[-12.85,\ -10.11]$	6	A
	cGAN − no generation	C100, 5/10/20	AA	39.07/38.29; 28.37/25.13; 23.01/16.03	$+0.78\,[+0.16,\ +1.40]$; $+3.24\,[+2.70,\ +3.79]$; $+6.98\,[+5.03,\ +8.93]$	6	A
	pixel-stat+SSIA − no generation	C100, 5/10/20	AA	38.08/38.29; 25.11/25.13; 16.15/16.03	-0.21; -0.02; $+0.12$	6	B
	cVAE − no generation	C100, 5/10/20	AA	38.01/38.29; 24.84/25.13; 16.07/16.03	-0.28; -0.29; $+0.04$	6	B
SDDR	SD cache − noise cache	C100, T10	AA	–	$+1.84\,[+1.42,\ +2.26]$	6	A
	SD cache − noise cache	C100, T25	AA	–	$+2.09\,[+1.64,\ +2.55]$	6	A
	SD cache − noise cache	C100, T10/T25	final	62.60/61.22; 58.89/56.52	$+1.38$; $+2.37$	1	B
	SD cache − no replay	C100, T10/T25	AA	70.72/68.93; 67.76/64.82	$+1.79$; $+2.95$	1	B
Can Synth.	TI − no synthetic (prototype head)	CUB	AA	59.53 / 58.56	$+0.97\,[+0.37,\ +1.56]$	6	A
	TI − no synthetic (prototype head)	CUB	final	–	$+1.94\,[+1.41,\ +2.48]$	6	A
	class-name − no synthetic (prototype head)	CUB	AA	–	$+0.38\,[-0.06,\ +0.82]$	6	A
	TI − class-name (prototype head)	CUB	AA	–	$+0.59\,[+0.04,\ +1.14]$	6	A
	TI (1k steps) − no synthetic	CUB	AA	–	$+0.74\,[-0.13,\ +1.62]$	6	A
	TI (3k steps) − no synthetic	CUB	AA	–	$+0.17\,[-0.29,\ +0.62]$	6	A
	TI (3k steps, LR 10^{-3}) − no synthetic	CUB	AA	57.03 / 56.91	$+0.12$	1	B
	TI − no synthetic	miniImageNet	AA	51.24 / 50.94	$+0.30\,[-0.24,\ +0.84]$	4	B
	class-name − no synthetic	miniImageNet	AA	50.95 / 50.94	$+0.01\,[-1.27,\ +1.29]$	4	B
	TI − class-name	miniImageNet	AA	51.24 / 50.95	$+0.29\,[-0.52,\ +1.10]$	4	B

Table 15: Feature-level ledger.

Family	Target − Control	Setting	Metric	T / C	Δ [95% CI]	n	Tier
DiffFR	diffusion+PC − Gaussian	C100, T10	AA	60.98 / 60.71	$+0.27\,[-0.04,\ +0.58]$	3	B
	diffusion+PC − residual	C100, T10	AA	60.98 / 64.86	$-3.88\,[-4.90,\ -2.85]$	3	B
	raw diffusion − diffusion+PC	C100, T10	AA	56.26 / 60.98	-4.72	3	B
	diffusion+PC − Gaussian / residual	C100, T5	AA	62.02 / 61.78; 62.02 / 65.01	$+0.24$; -2.99	2	B
	diffusion+PC − Gaussian / residual	C100, T20	AA	59.19 / 58.79; 59.19 / 63.22	$+0.40$; -4.02	2	B
DiffFR+ LA+SIM	diffusion+PC − Gaussian	C100, T10	AA	65.63 / 65.37	$+0.26\,[+0.06,\ +0.46]$	3	B
	diffusion+PC − residual	C100, T10	AA	65.63 / 68.84	$-3.21\,[-4.20,\ -2.22]$	3	B
	raw diffusion − diffusion+PC	C100, T10	AA	60.36 / 65.63	-5.27	3	B
	diffusion+PC − Gaussian / residual	C100, T5	AA	66.39 / 66.26; 66.39 / 69.35	$+0.13$; -2.96	2	B
	diffusion+PC − Gaussian / residual	C100, T20	AA	64.35 / 64.06; 64.35 / 67.70	$+0.29$; -3.36	2	B

Family	Target − Control	Setting	Metric	T / C	Δ [95% CI]	n	Tier
DiffFR (4 datasets)	diffusion+PC − Gaussian	C100, CUB, IN100, Tiny; T10/T20	AA	–	range -0.02 to $+0.17$	6	A
	raw diffusion − diffusion+PC	same	AA	–	range -84.8 to -19.9	6	A
Beyond	F_{syn} − repeated F_{inv}	CUB 5-shot	AA	–	$+0.29\,[-1.61,\ +2.18]$	6	A
	F_{aug} − repeated F_{inv}	CUB 5-shot	AA	–	$+1.74\,[-0.24,\ +3.71]$	6	A
	F_{syn} − repeated F_{inv}	CUB 5-shot	base acc.	–	$+0.49\,[-1.87,\ +2.85]$	6	C
	F_{aug} − repeated F_{inv}	CUB 5-shot	base acc.	–	$+2.77\,[+0.80,\ +4.74]$	6	C
	context: F_{syn} − F_{inv} alone (fixed caches)	CUB 5-shot	base acc.	60.28 / 51.04	$+9.24\,[+5.95,\ +12.53]$	6	C
	context: F_{aug} − F_{inv} alone (fixed caches)	CUB 5-shot	base acc.	59.89 / 51.04	$+8.85\,[+7.33,\ +10.37]$	6	C
	context: F_{aug} − F_{syn} (fixed caches)	CUB 5-shot	base acc.	59.89 / 60.28	$-0.39\,[-4.26,\ +3.48]$	6	C

Pairing notes. DCS contrasts use six matched seeds (3, 4, 6, 9, 12, 13) fixed in advance without inspecting accuracies. DiffClass contrasts use the same six seeds in both arms; the 5-task no-generation AA is 38.29 on both CIFAR-100 and ImageNet-100 (38.290 and 38.286 before rounding). Per-seed cGAN values follow from the per-seed SD1.5 − cGAN differences, so the cGAN rows are paired over the same six seeds; per-run values of the pixel-stat and cVAE arms were not retained, so those rows report six-seed means without intervals. The Beyond fixed-cache repeats reuse the same three feature caches, so their CI covers downstream training only.

F Alternative-Control Details

Table 16: Alternative controls (full version of Table 5). Each row adds a control that fixes a factor the primary control leaves open; the first Beyond row instead drops exposure matching and is context only. Results are paired Target−Control differences (pp) with 95% CIs over $n = 6$ matched runs per setting. Appendix D gives the designs.

Family	Added control (what it additionally fixes)	Result	Revised conclusion
DCS	Reward-guided vs. zero-reward sampling; same generator, prompt, image noise, labels, and count; CUB-200/miniImageNet × 1/5-shot × 3 replay budgets	CI above / below / covering zero in 4/2/6 of 12 settings (AA) and 3/2/7 (final)	Guidance effect depends on dataset, shot, and budget; no stable selection gain
DiffClass	Compute-matched no generation (rows and steps matched)	6/6 CIs cover zero (-0.11 to $+0.16$ AA)	Replay gain is not extra optimization
	SSIA vs. random selection from the same 100-candidate pool, matched score computation	11/12 CIs cover zero (N = 5/10/20 × q = 10/25 × AA/final); exception $+0.55\,[+0.05,\ +1.04]$	No stable selection advantage
	Source × budget × cache: new 50/class caches, 4 sources, 3 budgets, 5 pool partitions, SD1.5 vs. SD3.5	SD1.5−cGAN at 50/class: $+2.68$, $+2.71$, $+2.91$ (all CIs > 0); at 10–20/class the ordering reverses for 20 tasks	SD1.5 over cGAN holds at 50/class only; source ranking is setting-dependent
SDDR	Same 5,000 images, labels deranged (rows, order, label histogram fixed)	3/4 CIs cover zero; T25 final $+0.92\,[+0.44,\ +1.41]$	Label effect detected at one endpoint only
	Replay-only / distillation-only vs. neither	Replay $+0.56$, $+1.38$; distillation $+1.50$, $+1.95$ (T10, T25 AA; all CIs > 0)	Replay alone and distillation alone each improve over neither
Can Synth.	Factorial: 2 datasets × 1/5-shot × 1/3/5 images per class × {real, class-name, Textual Inversion} replay, plus no replay (240 runs)	Best strategy changes with dataset, shot, and budget	No stable direction for synthetic replay
DiffFR	CIFAR-100, CUB-200, ImageNet-100, TinyImageNet × T10/T20	Calibrated diffusion−Gaussian statistics: setting means -0.02 to $+0.17$; uncalibrated diffusion features 19.9–84.8 below calibrated	Performance tracks calibration and class statistics
Beyond	Base view alone, no exposure matching (fixed caches, base-session scaffold)	F_{syn} $+9.24\,[+5.95,\ +12.53]$, F_{aug} $+8.85\,[+7.33,\ +10.37]$ (base acc.)	Unmatched comparison shows large gains; context only
	Fixed caches, base-class second-view labels permuted	Permutation costs F_{inv} 16.0–19.4 more AA points than F_{syn}/F_{aug}; 8/8 CIs exclude zero	Alignment dependence is source-specific

DCS replay-budget grid. Reward-guided and zero-reward sampling are matched on generator, prompt, image noise, output label, replay amount, seed, and downstream protocol; the replay allocation is $3q/3q/q$ images per class for base/current/previous sessions. The zero-reward arm runs the same SMC sampler with a constant zero reward and zero reward gradient, so its samples are those of the guided path with the applied reward set to zero, and Table 17 isolates the reward's effect on sampling; only the reward model's compute time is not matched. On CUB-200 the gap widens with q (5-shot $q5 - q1$ interaction: AA $+0.69\,[+0.18,\ +1.20]$, final $+1.23\,[+0.38,\ +2.08]$), but both arms lose accuracy as q grows (guided -4.09/-6.80, zero-reward -4.78/-8.03 AA/final from $q = 1$ to $q = 5$); the gap widens because the zero-reward arm drops faster.

Table 17: DCS guided − zero-reward, six matched seeds per setting (pp).

Dataset	Shot	q	AA [95% CI]	Final [95% CI]
CUB-200	1	1	$+0.08\,[-0.21,\ +0.37]$	$-0.08\,[-0.32,\ +0.15]$
CUB-200	1	3	$+0.36\,[+0.01,\ +0.71]$	$+0.51\,[+0.03,\ +1.00]$
CUB-200	1	5	$+0.44\,[+0.12,\ +0.76]$	$+0.51\,[-0.04,\ +1.06]$
CUB-200	5	1	$+0.21\,[-0.18,\ +0.60]$	$+0.26\,[-0.30,\ +0.83]$
CUB-200	5	3	$+0.60\,[+0.15,\ +1.04]$	$+1.16\,[+0.53,\ +1.78]$
CUB-200	5	5	$+0.89\,[+0.43,\ +1.36]$	$+1.49\,[+0.71,\ +2.27]$
miniImageNet	1	1	$-0.13\,[-0.21,\ -0.05]$	$-0.25\,[-0.44,\ -0.07]$
miniImageNet	1	3	$-0.06\,[-0.11,\ +0.00]$	$-0.11\,[-0.25,\ +0.02]$
miniImageNet	1	5	$-0.08\,[-0.12,\ -0.05]$	$-0.21\,[-0.31,\ -0.12]$
miniImageNet	5	1	$-0.05\,[-0.25,\ +0.15]$	$-0.08\,[-0.45,\ +0.29]$
miniImageNet	5	3	$-0.05\,[-0.14,\ +0.03]$	$-0.09\,[-0.24,\ +0.05]$
miniImageNet	5	5	$+0.01\,[-0.07,\ +0.09]$	$+0.04\,[-0.16,\ +0.24]$

DiffFR scaffolds and stored statistics. The CIFAR-100 contrasts in Table 3 use an end-to-end scaffold: a ResNet-18 extractor is trained on the base classes and then frozen, and all replay arms of a run share its features. The four-dataset replication uses a separate scaffold on precomputed, L2-normalized frozen features of each dataset, so its CIFAR-100 T10 estimate and its calibration gap (uncalibrated diffusion features 19.9–84.8 pp below calibrated ones, against 4.72 pp in Table 15) are not comparable with the end-to-end numbers. Gaussian replay stores a mean and a diagonal standard deviation per class (2×512 values in the end-to-end scaffold); residual replay additionally stores the standardized residuals of all training features of every seen class (500 per class on CIFAR-100) and draws residuals from this pool before rescaling them with the target class statistics; diffusion replay stores the class statistics used for calibration and a feature-diffusion model.

DiffClass generator package. Table 18 holds the DiffClass classifier, splits, and replay budgets fixed and compares an SD3.5 cache with an SD1.5 cache. The generator and its generation recipe change together, so the contrast measures sensitivity to the source package. Across three source packages (two SD1.5, one SD3.5; 162 runs), the two SD1.5 packages agree within noise (8/9 AA and 9/9 final CIs cover zero), while comparisons involving the SD3.5 package produce a different ordering. Across four sources (pixel statistics, cGAN, SD1.5, real) at $q \in \{10, 25, 50\}$ and across five disjoint 20-image partitions of fixed 100-image SD1.5 and cGAN pools, the SD1.5-over-cGAN ordering at $q \in \{10, 20\}$ is positive in all partitions for 5 tasks, negative in all partitions for 20 tasks, and mixed for 10 tasks, so the best source depends on the split, budget, and cache.

Table 18: DiffClass on CIFAR-100: SD3.5 − SD1.5 cache under a fixed classifier; six matched seeds; absolute values are six-seed means with 95% CIs.

Tasks	q	SD1.5 AA	SD3.5 AA	ΔAA	Δfinal
5	10	$39.25\,[38.53,\ 39.97]$	$40.80\,[40.12,\ 41.48]$	$+1.55\,[+1.25,\ +1.85]$	$+1.67\,[+1.15,\ +2.19]$
5	25	$40.71\,[39.78,\ 41.64]$	$42.33\,[41.38,\ 43.29]$	$+1.62\,[+1.44,\ +1.81]$	$+2.04\,[+1.59,\ +2.49]$
5	50	$42.05\,[41.09,\ 43.01]$	$43.70\,[42.88,\ 44.52]$	$+1.65\,[+1.19,\ +2.11]$	$+2.05\,[+1.45,\ +2.64]$
10	10	$27.46\,[26.63,\ 28.30]$	$29.68\,[28.58,\ 30.78]$	$+2.22\,[+1.47,\ +2.97]$	$+2.48\,[+1.70,\ +3.25]$
10	25	$29.87\,[28.72,\ 31.02]$	$32.14\,[31.28,\ 32.99]$	$+2.27\,[+1.44,\ +3.10]$	$+2.98\,[+2.15,\ +3.82]$
10	50	$31.71\,[30.47,\ 32.95]$	$34.52\,[33.20,\ 35.84]$	$+2.81\,[+2.36,\ +3.26]$	$+2.87\,[+1.75,\ +4.00]$
20	10	$19.04\,[18.55,\ 19.52]$	$22.46\,[21.47,\ 23.45]$	$+3.42\,[+2.79,\ +4.06]$	$+3.48\,[+2.96,\ +3.99]$
20	25	$22.68\,[21.54,\ 23.82]$	$25.47\,[24.49,\ 26.46]$	$+2.79\,[+2.01,\ +3.57]$	$+4.02\,[+3.06,\ +4.98]$
20	50	$25.96\,[24.73,\ 27.19]$	$29.36\,[27.83,\ 30.88]$	$+3.40\,[+2.79,\ +4.01]$	$+3.75\,[+2.74,\ +4.75]$

Common-head representations and a pretraining anchor. The SD1.5 feature in this block is extracted from 512×512 inputs by one U-Net forward pass at timestep 0 with an empty prompt; the outputs of eight U-Net blocks (down blocks 1–3, the mid block, and up blocks 0–3) are globally average-pooled and concatenated, and a cosine nearest-class-mean head is applied. This single-step

pooled feature differs from Beyond's multi-scale spatial features with learned aggregation, so the SD rows are a context anchor for pretrained representations, not an evaluation of diffusion representations for FSCIL. Table 19 reports the same-architecture anchor: a supervised ImageNet-1K ResNet-18 and the same architecture with one fixed random initialization share preprocessing, class order, support sets, the cosine NCM head, and all sessions. Supervised minus random isolates the effect of pretrained weights for this architecture; SD minus random shows that SD1.5 features carry usable signal but also changes architecture and pretraining. Relative to SD1.5 features under the same head, the supervised ResNet-18 is higher by 39.98 (CUB, 1-shot), 47.74 (CUB, 5-shot), 43.32 (miniImageNet, 1-shot), and 50.26 pp (miniImageNet, 5-shot) AA; every interval excludes zero.

Table 19: Common-head AA with a random-initialization anchor (six seeds; pp). RN18: ResNet-18. SD1.5: pooled single-step U-Net feature (see text), a context anchor.

Setting	Rand. RN18	Sup. RN18	Sup. − rand.	SD1.5	SD − rand.	Sup. − rand. (final)
CUB, 1-shot	2.81	49.31	+46.50 [46.40, 46.59]	9.32	+6.51 [6.41, 6.61]	+33.69 [33.50, 33.88]
CUB, 5-shot	2.86	57.34	+54.49 [54.27, 54.70]	9.60	+6.74 [6.51, 6.98]	+47.95 [47.56, 48.33]
mini, 1-shot	6.38	77.08	+70.69 [70.04, 71.35]	33.76	+27.37 [27.16, 27.59]	+61.43 [60.56, 62.31]
mini, 5-shot	6.92	88.02	+81.10 [80.74, 81.47]	37.76	+30.84 [30.63, 31.06]	+78.92 [78.33, 79.50]

F.1 Shared-scaffold comparison of two complete systems

To check whether cross-family orderings depend on the dataset, DCS and SDDR were run as complete systems in one evaluation scaffold that fixes the ImageNet-pretrained ResNet-18 initialization (whose pretraining classes include those of miniImageNet), class and session order, support and evaluation examples, evaluator, and a maximum of 30 replay candidates per class, on CUB-200, miniImageNet, and CIFAR-100 with 1 and 5 shots and five matched seeds (Table 20). DCS is higher on CUB-200, lower on miniImageNet, and indistinguishable in AA on CIFAR-100. The two systems keep different training procedures, so the difference is not attributed to a single component; for seeds 2–4, 16 of the 60 runs have a producer status recorded as partial, although each contains a complete per-session accuracy trace, paired between the two systems.

Table 20: DCS − SDDR in a shared scaffold (five matched seeds; pp; df = 4).

Dataset	Shot	DCS AA	SDDR AA	ΔAA	Δfinal
CUB-200	1	52.08	51.43	+0.65 [+0.40, +0.91]	+0.61 [+0.16, +1.06]
CUB-200	5	56.21	53.30	+2.90 [+2.77, +3.04]	+3.18 [+2.69, +3.66]
miniImageNet	1	35.54	49.17	−13.63 [−14.46, −12.80]	−9.29 [−10.39, −8.19]
miniImageNet	5	34.51	49.72	−15.21 [−15.96, −14.46]	−12.35 [−13.24, −11.47]
CIFAR-100	1	63.14	62.96	+0.18 [−0.04, +0.39]	+0.90 [+0.15, +1.65]
CIFAR-100	5	63.45	63.29	+0.16 [−0.16, +0.47]	+0.83 [+0.18, +1.48]

G Scope of Each Conclusion

Table 21 states the scope of each conclusion in Section 5. Two conclusions are replicated across datasets within a family, three hold in a single setting, and two are setting-dependent. The cross-family statement is qualitative: the component that supports a gain differs between families. In DCS, DiffFR, and Beyond the diffusion component does not resolve a gain over a simpler primary control (the unselected generated pool, stored class statistics, an exposure-matched repeat), whereas in DiffClass it beats both no generation and a compute-matched control. When DCS and SDDR are run in a shared scaffold with a common initialization, support and query sets, evaluator, and replay budget, DCS is higher on CUB-200, lower on miniImageNet, and indistinguishable on CIFAR-100 (Appendix F.1); numbers from different families should therefore not be combined into a single ranking.

H Accuracy Versus Seen Classes

Figure 2 (Section 5.1) plots per-session accuracy against the number of seen classes for the DiffClass and DCS image-replay contrasts. The curves use the stored per-session accuracy of every run, with no interpolation or reconstruction from AA, and the mean of each curve reproduces the reported AA. For DiffClass, the SD1.5 replay gain over no generation has a CI above zero at every post-base session in all six dataset–split settings (64/64 sessions), and it is largest in the early and middle sessions of the 20-task splits; the compute-matched control covers zero at 53 of 64 sessions. For DCS,

Table 21: Scope map. *Single setting*: one family on one dataset. *Replicated*: the same direction in every declared setting of one family. *Setting-dependent*: declared settings disagree under the stability rule. *Cross-family*: a qualitative pattern across families; no numbers are pooled.

Conclusion	Scope	Basis
DiffClass: SD1.5 replay beats no generation, not because of extra rows or steps	Replicated	2 datasets × 3 splits (Tier A); compute control 6/6 null; 64/64 sessions
DiffFR: diffusion feature replay is matched by class-statistics replay	Replicated	4 datasets × 2 splits (Tier A ranges) and CIFAR-100 T5–T20 (Tier B)
SDDR: SD cache beats noise cache; replay and distillation both contribute	Single setting	CIFAR-100, T10 and T25 (Tier A)
DCS: no gain from reward selection over the unselected pool; below no generation	Single setting	CUB-200 5-shot (Tier A)
DCS: reward-guided vs. zero-reward sampling	Setting-dependent	2 datasets × 2 shots × 3 budgets
Can Synthetic: synthetic replay effects are small	Setting-dependent	3 CUB-200 protocols (Tier A), miniImageNet (Tier B), 240-run factorial
Beyond: against an exposure-matched base view, added feature views show no resolved full-sequence gain; F_{aug} keeps a base-session gain	Single setting	CUB-200 5-shot (Tier A); unmatched comparison is Tier C context
The component that supports a gain differs across families, and ablating a module is not the same as testing its role	Cross-family	All six audits; the DCS–SDDR ordering flips across datasets in a shared scaffold (Appendix F.1)

reward-selected replay is below no generation at all 10 post-base sessions, with the gap opening after the first incremental sessions. This version plots only the DiffClass and DCS image-replay contrasts. Per-session curves for DCS guided-versus-unguided sampling, DiffFR, SDDR, and the common-head block were computed from the same kind of stored logs (112 six-seed groups including DiffClass) and agree with the corresponding AA readings (no consistent per-session DCS guidance advantage; DiffFR calibrated-minus-Gaussian differences near zero in most sessions; SD1.5 features below CLIP and DINOv2 at every session), but they are not plotted here.

I Cache Diagnostics and Direct Overlap Screen

Cache quality. Table 22 reports lightweight diagnostics for the image-replay caches where quality affects how a control should be read. We sample up to 1,000 generated and 1,000 real training images per source and use one CLIP ViT-B/32 encoder for the CLIP-space rows; CLIP-Fréchet and CLIP-KID measure distance to the matched real sample (lower is closer). Textual Inversion brings the Can Synthetic CUB-200 cache closer to real images than class-name prompting (CLIP-Fréchet $0.0916 \rightarrow 0.0576$, FID $26.68 \rightarrow 23.49$). The DiffClass SD1.5 cache is much closer to real CIFAR-100 images than the non-diffusion sources (CLIP-Fréchet 0.0563 vs. 0.1308 for cGAN, 0.2462 for cVAE, 0.4089 for pixel statistics), so cGAN is the most comparable non-diffusion source and pixel statistics and cVAE are stress controls.

Table 22: Generated-cache diagnostics. cls: classes; label: CLIP class-prompt score; Frét.: CLIP-Fréchet; div.: CLIP diversity; NN cos.: nearest-real cosine (a domain check, not a leakage test); FID: Inception FID where computed.

Source	n	cls	label	Frét.	KID	div.	NN cos.	FID	pixel mean/std
Can Synth. CUB class-name	1000	200	0.3267	0.0916	0.0002	0.1064	0.8854	26.68	0.4450/0.1737
Can Synth. CUB Textual Inversion	1000	200	0.3313	0.0576	0.0001	0.1013	0.8885	23.49	0.4363/0.1901
DiffClass C100 SD1.5	1000	100	0.2692	0.0563	0.0001	0.1582	0.8981	80.38	0.4366/0.1771
DiffClass C100 pixel-stat	1000	100	0.2316	0.4089	0.0013	0.0163	0.8593	396.29	0.4761/0.1813
DiffClass C100 cVAE	1000	100	0.2339	0.2462	0.0007	0.0758	0.9160	169.67	0.4650/0.2171
DiffClass C100 cGAN	1000	100	0.2457	0.1308	0.0003	0.1013	0.9107	n/a	0.4662/0.2795

Direct overlap screen. The screen covers 59 saved caches (69,940 images): DiffClass SD1.5, cGAN, and cVAE caches on CIFAR-100 and the SD1.5 cache on ImageNet-100; Can Synthetic class-name and Textual-Inversion caches on CUB-200 and miniImageNet; 48 DCS caches (two datasets, 1/5-shot, guided and zero-reward, six seeds); and the SDDR SD1.5 cache. For each cache and the full downstream training and test sets we compare SHA-256 hashes of decoded RGB arrays and resolved file paths, and for a deterministic stratified sample we compute 64-bit perceptual-hash nearest neighbors; all three checks found zero collisions. CLIP and DINO nearest-neighbor similarities are retained as continuous scores calibrated against train-to-test similarity and are not thresholded into a leakage decision. The screen can detect exact copies of benchmark images in the saved caches. It cannot establish whether benchmark images or classes are members of the generators' pretraining

corpora, cannot detect near-duplicates outside the sampled pairs, and cannot detect semantic or label-level knowledge inherited through the text encoder.