STUDENT AS STRONG AS TEACHER: SCALING SYNTHETIC DATA IN KNOWLEDGE DISTILLATION OF CLASSIFIERS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present a pipeline for training a high-performing small classifier through knowledge distillation (KD) augmented with samples generated from a fine-tuned diffusion model. Using this pipeline, we find that student performance scales predictably with the size of the synthetic data. We further improve the pipeline by dropping all synthetic data midway through training, thereby halving the training cost while surprisingly improving performance. Experiments on five classification benchmarks and a semantic segmentation setting consistently show gains from both scaling and dropping synthetic data. To the best of our knowledge, we present the first examples of a ResNet-18 (11M parameters) matching or exceeding a ViT-L teacher (307M parameters) through cross-architecture distillation. Our results suggest that small neural networks can achieve substantially stronger performance when provided with sufficiently rich distillation data.

1 INTRODUCTION

Small models remain indispensable even in the era of scaling laws (Kaplan et al., 2020) and the "bitter lesson" (Sutton, 2019). Although large models offer strong performance, their memory footprint, power consumption, and latency often make them unsuitable when inference resources are limited. Applications such as perception on mobile and edge devices (Zhou et al., 2025), autonomous robots and drones (Ali et al., 2025), and real-time industrial inspection (Lin et al., 2024) require models to respond quickly and reliably on constrained hardware. Small models are well suited to these deployment conditions but generally underperform larger models. Closing this performance gap without increasing inference-time cost is therefore a central challenge for resource-constrained deployment.

Knowledge distillation (KD; Hinton et al., 2015) is a technique for boosting the performance of a small model ("student") by using a larger model ("teacher") as an additional source of supervision. Despite its widespread adoption, however, a substantial performance gap between the teacher and student often remains. In this paper, we hypothesize that this gap mainly stems from the *pointwise* nature of the standard KD objective. Specifically, the student is encouraged to match the teacher only at the training inputs it observes, receiving no direct guidance on how the teacher behaves elsewhere in the input space (Figure 1). Consequently, when the training inputs provide only sparse coverage of the underlying data distribution, the student may struggle to faithfully recover the teacher's function.

We address this supervision gap with synthetic data. In the pipeline we propose in this paper, we use a fine-tuned diffusion model to supply additional inputs on which the KD loss is enforced. When the synthetic distribution sufficiently covers the target distribution, increasing the number of synthetic samples reduces the teacher–student gap toward a limit controlled by the student's approximation error under the synthetic distribution. We test this prediction by training the student on the union of real and synthetic images while systematically scaling the synthetic pool. We found that the student's accuracy and test loss improve predictably with the number of synthetic samples, often making the student as strong as the teacher.

The addition of a large body of synthetic data incurs significant training time. We therefore propose **SynDrop**, a two-stage schedule that first trains with real and synthetic data, then drops all synthetic samples and continues on real data alone. SynDrop removes a substantial fraction of synthetic-data computation and, surprisingly, also consistently improves final accuracy in the large-pool regime

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.

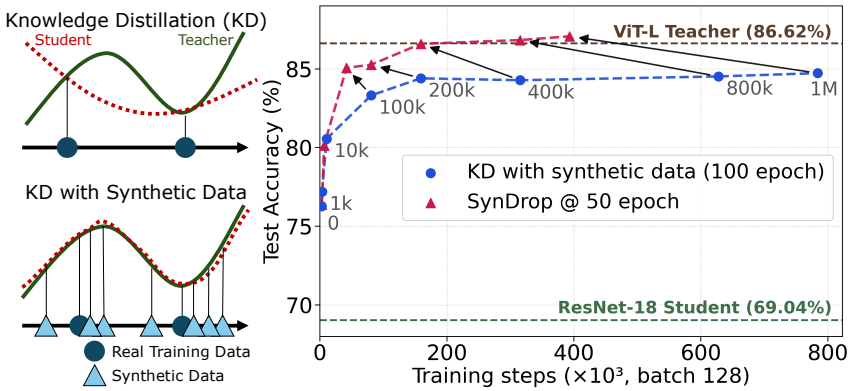

Figure 1: **Left:** Illustration of training with real and synthetic data in knowledge distillation. **Right:** Substantial gains from scaling and dropping synthetic data on Aircraft. Accuracy grows with the size of the synthetic pool (blue). Dropping the synthetic data midway through training and continuing with the real data alone (red; SynDrop) boosts the performance at reduced training cost. Arrows connect runs with the same synthetic data pool.

(Figure 1). This result suggests that synthetic data are most useful for establishing representations early in training, whereas prolonged exposure can bias the student toward imperfections of the synthetic distribution. Finishing on real data preserves the benefit of scale while returning the model to the target distribution. Altogether, synthetic data scaling and SynDrop carry the ResNet-18 beyond its ViT-L teacher on Aircraft and Stanford Cars.

Diffusion synthetic data has already been explored in diverse settings, including non-KD classification (Azizi et al., 2023; Fan et al., 2024) and data-free KD (Li et al., 2023b). When synthetic data is introduced in non-KD classification (Azizi et al., 2023), the performance improvement can be limited in some settings and does not necessarily scale monotonically with the quantity of synthetic data. In data-free KD (Li et al., 2023b), distillation with synthetic data exhibits counterintuitive behaviors, such as inferior performance under a stronger teacher and synthetic-data quality not correlating with distillation performance. Cross-architecture distillation also remains unexplored. Here, we investigate the role of synthetic data in a highly practical real-plus-synthetic, cross-architecture, large-scale setting and achieve significant performance improvements. Our pipeline closes the performance gap across a 30-fold parameter difference and extends to semantic segmentation. Taken together, our results establish synthetic data as a scalable and broadly applicable medium for knowledge transfer across architectures, model scales, and tasks.

Our contributions are summarized as follows:

- We present KD+Syn, a pipeline that uses diffusion synthetic data to augment KD, where student performance follows predictable scaling behavior as the synthetic pool grows.
- We propose SynDrop, which removes synthetic data partway through training to reduce computation while consistently improving accuracy at large synthetic scales.
- We demonstrate large-scale, cross-architecture distillation in which a ResNet-18 approaches or surpasses a ViT-L, and show that conclusions drawn in synthetic-only distillation do not necessarily carry over to the real-plus-synthetic regime.

2 Related Work

Scaling Synthetic Data. Generative data augmentation has been widely explored for increasing training diversity in image classification (Trabucco et al., 2024; Wang et al., 2024; Roy et al., 2025). Ensuring reliable supervision for generated samples remains important, motivating label-preserving generation (Islam et al., 2024), as well as CLIP-based filtering and soft labels (He et al., 2023). Prior work shows synthetic scaling limitations (Fan et al., 2024), while distribution matching (Yuan et al., 2024) and learner-guided generation (Askari-Hemmat et al., 2025) can improve its effectiveness. Notably, simply adding synthetic data does not guarantee sustained improvements: gains can peak and subsequently decline (Azizi et al., 2023), or saturate even with improved generation and

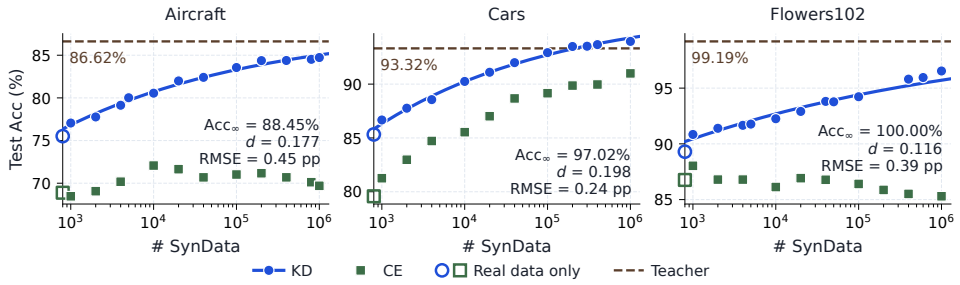

Figure 2: Scaling behavior of synthetic data under knowledge distillation: test accuracy of a ResNet-18 student trained with KD and with CE alone, against a ViT-L teacher fine-tuned on real data only. KD uses the synthetic label in the CE term on Aircraft and Cars, and the KL term alone on Flowers. Cross-entropy alone scales poorly on Aircraft and Flowers, whereas KD well follows the scaling law of Section 4.2 and carries the student close to or past the teacher (test loss in Appendix B.3).

training strategies (Yu et al., 2025). These findings motivate more robust supervision as synthetic data scale, particularly when synthetic labels may be imperfect. This motivates us to address this problem through knowledge distillation, using teacher predictions to complement synthetic labels as the dataset scales.

Data Augmentation for Knowledge Distillation. Prior work shows that carefully chosen augmentation applied to real training data can improve KD performance (Wang et al., 2022; Li et al., 2022). Importantly, Beyer et al. (2022) shows that "consistent" augmentation, and "patient" KD training (up to 1M epochs) can substantially narrow the accuracy gap using only the real training data. Building on this strong potential for knowledge transfer, we pursue a complementary direction by expanding the transfer dataset with diffusion-generated images, beyond transformations of the available real examples.

Data-Free Knowledge Distillation. Synthetic transfer data have also been studied when real training data are unavailable (Lopes et al., 2017; Chen et al., 2019; Luo et al., 2020; Yin et al., 2020; Fang et al., 2021). Li et al. (2023b) uses diffusion-generated images as the transfer set for KD without access to real training data, with student performance improving as synthetic data size scales. We instead utilize the available real data and scale synthetic data under KD, and observe empirical scaling laws that enable ResNet-18 and ResNet-50 students to exceed their ViT-L teachers.

3 Motivation: Expanding Distillation Support

We aim to boost a small model's performance to match that of a large model by approximating the large model's solution. This goal may be more attainable than it first appears. Prior work suggests that the interaction between architecture and optimization can cause large models to favor solutions that are simple in terms of norm or effective complexity (Neyshabur et al., 2019; Brutzkus & Globerson, 2019), including low-rank (Huh et al., 2022), smooth (Jin et al., 2020; Sahs et al., 2022), or compressible (Arora et al., 2018) functions. Thus, the irreducible approximation error of a small model relative to a large model's solution may be small. The remaining challenge is how to guide the small model toward such an approximation.

Limitations of Knowledge Distillation. Knowledge distillation (KD) (Hinton et al., 2015) provides a natural mechanism for such guidance. A small "student" is trained by imposing a pointwise loss that penalizes discrepancies from the predictions of a larger "teacher" at each training input. Yet a substantial teacher–student gap often remains: agreement at a finite set of training examples does not constrain the student over the rest of the input distribution. As illustrated in the left panel of Figure 1, sparse transfer inputs can therefore leave many student functions indistinguishable under the empirical distillation objective, even when they behave differently away from those inputs.

Expanding Distillation with Synthetic Inputs. We use synthetic data to expand the set of inputs on which knowledge is transferred. If a synthetic distribution Q constructed by a generative model provides good coverage of the real data distribution P_X, drawing more samples from Q makes the pointwise distillation objective constrain the student over the support of the data distribution. The student should then approach the best approximation to the teacher available within its own function

class under Q. This argument does not require the teacher to be exactly realizable by the student: any mismatch that the student class cannot represent remains as an approximation error. The following proposition formalizes this separation between approximation error and the finite-sample error that additional synthetic inputs can reduce.

Proposition 3.1 (Function Approximation with Synthetic Data; Informal)**.** *Suppose the synthetic distribution Q covers the target input distribution P_X, in the sense that the density ratio dP_X/dQ never exceeds some ρ, and that the student is searched over the radius-B ball $\mathcal{F}_B$ of a bounded-kernel RKHS. Let $\epsilon^2_{\mathrm{app}}(B)$ be the smallest teacher–student squared error that such a student can achieve on Q. A student $\hat{f}_{S,m}$ is trained by matching a bounded teacher f_T on m inputs drawn i.i.d. from Q. Then, with probability at least $1-\delta$,*

$$\underbrace{\|\hat{f}_{S,m} - f_T\|^2_{L_2(P_X)}}_{\text{gap on real data}} \leq \underbrace{\rho\epsilon^2_{\mathrm{app}}(B)}_{\text{coverage-adjusted approximation upper bound}} + \underbrace{O\left(\rho\sqrt{\frac{\log(2/\delta)}{m}}\right)}_{\text{shrinks as we generate more data}}, \tag{1}$$

where the hidden constant depends only on the kernel bound, B, and the teacher bound.

Appendix A.1 gives the precise constants and proof. As the number of synthetic inputs m grows, the finite-sample term shrinks and the bound approaches the coverage-adjusted approximation upper bound $\rho\epsilon^2_{\mathrm{app}}(B)$. Since synthetic data can be generated in large quantities, scaling the synthetic pool lets us test how far a fixed student can be improved before its function class, rather than sparse teacher supervision, becomes the limiting factor.

4 Scaling Synthetic Data for Knowledge Distillation

4.1 Knowledge Distillation Augmented with Generated Data

Here, we describe our KD+Syn protocol: KD on the union of the real training data and diffusion-generated synthetic data. Deviating from the regression setting in Proposition 3.1, we focus on classification problems from now on.

Algorithm 1 Synthetic Data Scaling in KD

Input: fine-tuned diffusion model Q, teacher T, real data $\mathcal{D}_{\mathrm{real}}$
$\mathcal{D}^M_{\mathrm{syn}} \leftarrow \textsc{Generate}(Q)$ ▷ *data generation*
$\mathcal{D}_{\mathrm{train}} \leftarrow \mathcal{D}_{\mathrm{real}} \cup \mathcal{D}^M_{\mathrm{syn}}$ ▷ *data augmentation*
for epoch $e = 0, \ldots, E-1$ **do**
  if **SynDrop** and $e = E_{\mathrm{drop}}$ **then**
    $\mathcal{D}_{\mathrm{train}} \leftarrow \mathcal{D}_{\mathrm{real}}$ ▷ **SynDrop**
  end if
  $\textsc{TrainOneEpoch}(\mathcal{D}_{\mathrm{train}}, T)$ ▷ *KD*
end for

Synthetic Data Generation Using Diffusion Models. We obtain synthetic data by fine-tuning a diffusion model on the real training data $\mathcal{D}_{\mathrm{real}} = \{(x_i, y_i)\}_{i=1}^N$ so that the synthetic data distribution Q approximates the data distribution P_X. We fine-tune pretrained diffusion models, using LoRA (Hu et al., 2022). Although pre-training a diffusion model is not necessary for our protocol, it improves performance and helps prevent memorization of the training data, especially when the training dataset is very small, as in Flowers-102. We fine-tune in a class-conditional manner, enabling classifier-free guidance (CFG) at sampling time.

We then a build synthetic data pool $\mathcal{D}^M_{\mathrm{syn}} = \{(\tilde{x}_j, \tilde{y}_j)\}_{j=1}^M$ by sampling from the fine-tuned model with classifier-free guidance. For each sample, we first draw a class label $\tilde{y}$ uniformly, and then generate an image $\tilde{x}$ conditioned on $\tilde{y}$. The guidance scale w is a hyperparameter of the pipeline; it trades class fidelity against diversity, and we examine its effect on distillation in Section 6.3.1.

Knowledge Distillation. We apply the standard KD loss (Hinton et al., 2015) to the augmented dataset obtained by merging the real and the synthetic datasets $\mathcal{D}_{\mathrm{aug}} = \mathcal{D}_{\mathrm{real}} \cup \mathcal{D}^M_{\mathrm{syn}}$. In Hinton et al.'s KD, a student network S is trained using both the ground-truth labels and the softened predictions of a fine-tuned teacher network T. Given a labeled example $(x, y) \in \mathcal{D}_{\mathrm{aug}}$, the objective is given as follows:

$$\mathcal{L}_{\mathrm{KD}}(x, y) = \alpha\, \mathcal{L}_{\mathrm{CE}}(p_S(x), y) + (1-\alpha)\tau^2 \mathrm{KL}(p^\tau_T(x) \,\|\, p^\tau_S(x)),$$

where $p^\tau_S(x) = \mathrm{softmax}(z_S(x)/\tau)$ and $p^\tau_T(x) = \mathrm{softmax}(z_T(x)/\tau)$. Here, z_S and z_T denote the student and teacher logits, respectively. $\tau > 0$ is the temperature controlling the softness of the predictive distributions, and $\alpha \in [0, 1]$ balances supervision from hard labels and the teacher. We

Table 1: Classification results measured by test top-1 accuracy (%). Student models outperforming ViT-L (the teacher used for KD) are underlined, and the best student-model accuracy for each dataset is shown in **bold**. Student models are trained for 100 epochs with the same learning rate schedule.

Model (#Params)	Method	Syn. data size	Aircraft	Cars	Flowers	CUB200	SUN397
ViT-L (307M)	CE	0	86.62	93.32	99.19	89.32	79.30
ResNet-18	CE	0	68.89	79.58	86.76	70.31	56.14
(11M)	KD	0	75.52	85.33	89.30	73.82	57.13
	CE+Syn	1M	69.70	91.00	85.30	75.42	56.18
	KD+Syn	1M	84.73	93.97	94.60	83.64	66.92
	+SynDrop@50epoch	1M	**87.07**	**94.99**	96.18	84.79	**67.56**
	KD+Syn (no SynLabel)	1M	83.80	92.79	96.54	83.55	66.28
	+SynDrop@50epoch	1M	84.49	93.33	**97.02**	**85.05**	66.87
ResNet-50	CE	0	72.25	85.08	89.25	76.08	60.98
(26M)	KD	0	79.66	89.18	91.66	77.08	60.39
	CE+Syn	1M	76.99	93.33	88.68	79.50	62.78
	KD+Syn	1M	86.59	94.48	96.85	86.11	68.20
	+SynDrop@50epoch	1M	**87.76**	**94.89**	97.48	**86.50**	68.50
	KD+Syn (no SynLabel)	1M	85.66	93.25	97.72	86.00	69.45
	+SynDrop@50epoch	1M	86.50	93.68	**97.98**	86.28	**69.96**

treat τ and α as hyperparameters and tune them based on the validation performance. $\mathcal{L}_{\mathrm{CE}}$ denotes cross-entropy loss.

Note that, with this KD loss, we deviate from the discussion in Section 3 and Proposition 3.1, as we additionally use real and synthetic label information in the cross-entropy loss. If these labels provide useful information, the student is not necessarily prevented from outperforming the teacher.

Ignoring Synthetic Label ("No SynLabel"). Optionally, we may choose not to trust the synthetic label $\tilde{y}$, as the diffusion model's class-conditional distribution may deviate from the true distribution. In this case, we set $\alpha = 0$ for all synthetic data in a minibatch. This choice ignores synthetic labels during training, leaving the teacher as the sole source of supervision only for synthetic data points. This option is especially useful when the teacher is very strong, as in our Flowers experiment.

4.2 Scaling Behavior of Synthetic Data in Knowledge Distillation

An important property of our pipeline is that we can increase the number of synthetic samples M as much as computational constraints allow. As implied by Proposition 3.1, the gap between the teacher and the student is expected to decrease as M grows. Indeed, in our experiments, we observe that both test cross-entropy loss $\mathcal{L}_{\mathrm{test}}$ and test accuracy $\mathcal{A}_{\mathrm{test}}$ exhibit predictable scaling-law behavior:

$$\mathcal{L}_{\mathrm{test}}(M) \approx \mathcal{L}_\infty + aM^{-b} \quad \text{and} \quad \mathcal{A}_{\mathrm{test}}(M) \approx \mathcal{A}_\infty - cM^{-d} \quad (M > 0),$$

where $a, b, c, d > 0$, and $\mathcal{L}_\infty$ and $\mathcal{A}_\infty$ denote the asymptotic test loss and accuracy, respectively. Many factors can influence the asymptotic performance $\mathcal{L}_\infty$ and $\mathcal{A}_\infty$, including teacher performance, student capacity, and the informativeness of real and synthetic labels. In our experiments (Table 1), we find that $\mathcal{A}_\infty$ is surprisingly higher than the original student performance and much closer to that of the teacher. However, achieving $\mathcal{A}_\infty$ requires large M, which proportionally increases the training time of a student and quickly becomes nontrivial.

5 SynDrop: Dropping Synthetic Data During Training

To address the training cost of scaling M, we propose SynDrop, which drops the synthetic data midway through training. Let E denote the total number of epochs and $E_{\mathrm{drop}} < E$ the epoch at which the synthetic data are dropped. For the first E_{drop} epochs the student is trained on $\mathcal{D}_{\mathrm{aug}} = \mathcal{D}_{\mathrm{real}} \cup \mathcal{D}_{\mathrm{syn}}^{M}$, and for the remaining $E - E_{\mathrm{drop}}$ epochs on $\mathcal{D}_{\mathrm{real}}$ alone (Algorithm 1). The student is not reinitialized at the drop, and neither the loss nor the learning-rate schedule changes. Only the training set changes. We use $E_{\mathrm{drop}} = E/2$ throughout and examine the choice of E_{drop} in Section 6.2.

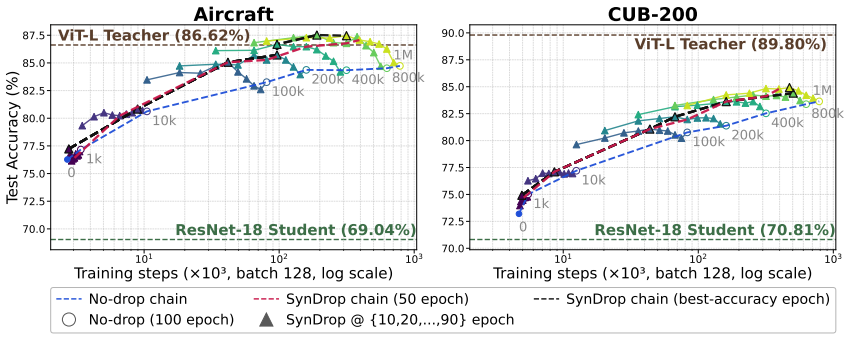

Figure 3: SynDrop under KD on Aircraft (left) and CUB200 (right) with a ResNet-18 student. For each synthetic pool size M from 1K to 1M (colors), triangles show training steps and final test accuracy for $E_{\text{drop}} \in \{10, 20, \ldots, 90\}$ with $E = 100$, from left (early drop) to right; the rightmost point for each pool (circle) is no-drop training for all 100 epochs. For $M \geq 100$K, SynDrop at $E_{\text{drop}} = 50$ (red) consistently outperforms no-drop training and is close to the best drop epoch for each pool (black): the accuracy is robust to the choice of E_{drop} around the middle of training.

We observe that SynDrop not only reduces training cost but also substantially improves final accuracy when M is large. Specifically, since $M \gg |\mathcal{D}_{\text{real}}|$ in our setting, the number of training steps is dominated by the synthetic pool, and SynDrop removes nearly a fraction $1 - E_{\text{drop}}/E$ of them, i.e., roughly half for $E_{\text{drop}} = E/2$. At the same time, for large M, the final accuracy exceeds that of training with the synthetic data retained throughout (Figure 1, Figure 3).

We attribute this to a division of labor between the two stages. The large synthetic pool first gives dense coverage of the teacher's function, from which the student learns a representation. After that, further training with the synthetic data adds little but keep pulling the student toward the synthetic distribution Q. Dropping the synthetic pool lets the student spend its remaining epochs adapting that representation to the real data alone. This two-stage view is consistent with prior work on large-scale semi-supervised and self-supervised learning (Yalniz et al., 2019; Chen et al., 2020) and on pretraining with diffusion models and diffusion-generated images (Li et al., 2023a; Basu et al., 2025), all of which build representations on a large auxiliary set before fine-tuning on the target data.

6 Experiments

We evaluate our method on five fine-grained classification benchmarks: Aircraft (Maji et al., 2013), Stanford Cars (Krause et al., 2013), Oxford Flowers-102 (Nilsback & Zisserman, 2008), CUB200 (Wah et al., 2011), and SUN397 (Xiao et al., 2010). Synthetic pools are sampled from a text-conditional model SDXL (Podell et al., 2024), and a class-conditional model EDM2-S (Karras et al., 2024), each adapted to the training split of the target dataset with LoRA (Hu et al., 2022). We distill a fine-tuned ViT-Large (Dosovitskiy et al., 2021) teacher into ResNet-18 and ResNet-50 students (He et al., 2016) with vanilla logit distillation (Hinton et al., 2015). Appendix B.1 reports further details on the dataset, the generation settings, and the training recipe.

6.1 Training Students as Strong as Teachers

Scaling behavior. In Figure 2, we show that test accuracy improves predictably with the synthetic pool size M, and in Appendix B.3 that the test cross-entropy loss behaves the same way. On Flowers, we observe this scaling under KD (no SynLabel) setting of Section 4.1, which is direct empirical evidence for Proposition 3.1: more samples steadily close the teacher–student gap. On Aircraft and Cars, we train in KD setting with synthetic labels, and the scaling behavior survives the departure from the assumptions of the proposition.

Fitting $\mathcal{L}_{\text{test}}(M)$ and $\mathcal{A}_{\text{test}}(M)$ with the functional forms of Section 4.2 by nonlinear least squares yields estimates of the accuracy our method converges to. On Aircraft and Flowers, the fitted $\mathcal{A}_\infty$ (88.45% and 100.00%) lands within a point or two of the teacher (86.62% and 99.19%), as the proposition would predict. On Cars, $\mathcal{A}_\infty = 97.02\%$ sits well above the teacher's 93.32%: with

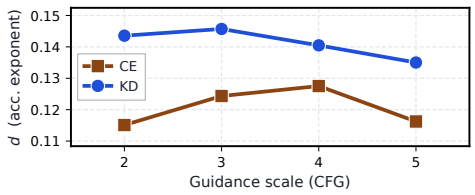

Figure 4: Accuracy scaling exponent (d) across different CFG values in Stanford Cars. The optimal CFG is lower under KD than under CE.

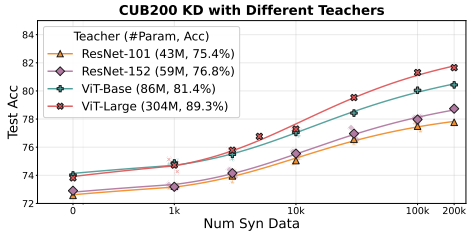

Figure 5: Test accuracy of KD+Syn with various teachers in CUB200. Stronger (and larger) teacher yields better scaling.

synthetic labels in play, the student is not bounded by the teacher, consistent with the remark in Section 4.1.

Cross-entropy training on the same pools behaves very differently from KD. On Aircraft and Flowers it does not scale at all and accuracy stays flat in M, and on Cars it does scale, but stays below KD at every pool size. We attribute the gap to the generator: conditioned on a class y, it does not reliably produce an image that belongs to y. While cross-entropy treats the intended label as ground truth and is pushed toward the wrong class by every such failure, KD matches the teacher's logits on the image actually generated and is therefore far less sensitive to an imperfect generator.

Results at 1M images. When we scaled the synthetic data pool to 1M images, our pipeline produces students that surpass their own teacher in some cases (Table 1): it lifts a ResNet-18 past a ViT-L teacher nearly $30\times$ its size, showing that the smaller architecture does not preclude teacher-level task accuracy in these settings. At this scale KD outperforms cross-entropy on every dataset and for both students. Using synthetic labels helps most of the time, but dropping synthetic labels yields better accuracy on Flowers, where the teacher is strongest with 99.19% accuracy. Once the teacher is more reliable than the generator's class assignment, it pays to trust the teacher alone.

Applying SynDrop at the halfway point of training improves accuracy in every configuration we ran, with and without synthetic labels. Together, scaling and SynDrop carry both students past the teacher on Aircraft and Cars, and they do so at lower cost: at $M = 1$M, SynDrop takes student training from 28–31 hours on a single GPU down to roughly half (Appendix B.2).

We additionally confirm that our performance gains are not an artifact of the generator reproducing test images by copy-detection audit of the full synthetic pools in Appendix B.4, so the improvement reflects generalization acquired from synthetic data under distillation.

6.2 When and How Does SynDrop Help?

Section 6.1 applied SynDrop at a fixed $E_{\text{drop}} = E/2$. We now examine how the gain depends on the drop epoch and the pool size M under KD. We vary $E_{\text{drop}} \in \{10, 20, \ldots, 90\}$ with $E = 100$, together with no-drop training, and $M \in \{1\text{K}, 10\text{K}, 100\text{K}, 200\text{K}, 400\text{K}, 800\text{K}, 1\text{M}\}$. Figure 3 plots test accuracy against training steps for each drop epoch on Aircraft and CUB200, chosen for their contrasting teacher–student gaps, with a ResNet-18 student. The best drop epoch for each pool is marked by the black chain. We also evaluate SynDrop under cross-entropy training and report the results in Appendix B.5.

When to drop synthetic data. The gains of SynDrop are robust to the choice of E_{drop} near $E_{\text{drop}} = E/2$. For $M \geq 100$K, dropping anywhere between epochs 30 and 60 stays within 0.5 points of the best drop epoch on Aircraft, and between epochs 30 and 80 on CUB200 (red vs. black chains). In this range, the accuracy varies little, so a fixed $E_{\text{drop}} = E/2$ suffices in practice. Two effects bound this window. Dropping too early leaves the student with too few passes over the synthetic data to learn the representation. On the other hand, dropping too late leaves too few epochs for the real data, with the learning rate already nearly annealed, and loses most of the gain.

Lower cost, higher accuracy. With $E_{\text{drop}} = 50$, SynDrop halves the training steps, since the real data are a negligible fraction of $\mathcal{D}_{\text{aug}}$. Surprisingly, this saving comes with a substantial gain in accuracy rather than a loss. Along the $E_{\text{drop}} = 50$ chain, accuracy increases monotonically with M on both datasets and exceeds no-drop training by $+1.8$ to $+3.0$ points on Aircraft ($M \geq 100$K) and

$+0.8$ to $+1.2$ on CUB200 ($M \geq 200$K), so SynDrop shifts the scaling curve of Section 4.2 upward. On Aircraft the student surpasses its ViT-L teacher for $M \geq 400$K. Note that this gain requires a large pool, and for $M \leq 10$K SynDrop does not improve on no-drop training. A larger pool also allows an earlier drop. On Aircraft, the earliest drop within 0.5 points of the best moves from epoch 40 at $M = 100$K to epoch 10 at $M = 1$M. On CUB200, it moves from epoch 30 to epoch 20. This is because a larger pool gives the student more synthetic samples per epoch, so it reaches an adequate representation sooner and the synthetic data can be discarded earlier.

6.3 What Affects the Scaling Behavior?

6.3.1 Diffusion Model

Guidance scale. The guidance scale has an optimum value that maximizes the scaling exponent as shown in Figure 4, because it trades class fidelity against diversity and a pool short on either scales poorly (Fan et al., 2024). In diffusion sampling, a large guidance scale pulls generation toward the conditioning signal, producing images that are more faithful to the intended class, at the cost of covering less of each class. Both properties matter for scaling: label correctness determines how much of the supervision is trustworthy, and diversity how much new information each additional sample contributes.

Table 2: Effect of generative model on scaling, measured on Stanford Cars with a ViT-L teacher and a ResNet-18 student. EDM2-S generates images closer to the real data, but the images generated by SDXL are more diverse and results a better student.

	SDXL	EDM2-S
FD-DINOv2 ↓	225.6	**70.5**
Teacher top-1 (%) ↑	82.6	**97.3**
Mean per-class Vendi ↑	**15.83**	8.88
Acc. at M=400k (%) ↑	**93.67**	92.74
Acc. scaling exponent d ↑	**0.198**	0.111

We find that this optimum moves under distillation. A cross-entropy objective uses only the assigned label, so class fidelity falls entirely on the generator. In contrast, the teacher's soft predictions stay informative even when the generator misses the intended class. Some label correctness can be given up at little cost, and the diversity bought in exchange improves the student. Thus, the best guidance scale under distillation is lower than under cross-entropy.

Generator choice. The same diversity and fidelity trade-off separates generators. Repeating the experiment on Stanford Cars with EDM2-S, student accuracy scales with the pool size there as well, but SDXL produces the stronger student at every pool size. We characterize each pool by its distance to the real data distribution (FD-DINOv2, Stein et al., 2023), its class fidelity (top-1 accuracy of the teacher on the pool), and its diversity (mean per-class Vendi score, Friedman & Dieng, 2023). EDM2-S is closer to the real distribution and far more class-faithful, and SDXL leads only on diversity (Table 2), which we attribute to SDXL's much larger pretraining corpus.

6.3.2 Teacher

As shown in Figure 5, we evaluate different teachers, including ResNet-101, ResNet-152, ViT-B, and ViT-L, on CUB200. Stronger teachers yield steeper scaling curves as the amount of synthetic data increases. While CUB200 originally exhibits a "knowledge gap," where a stronger teacher does not necessarily improve KD performance (Cho & Hariharan, 2019; Xu et al., 2020), this trend changes with synthetic data. At $M = 0$, ViT-Base outperforms ViT-Large, but with sufficient synthetic data, stronger teachers consistently lead to substantially better KD+Syn performance.

6.3.3 Student

Student capacity sets where the scaling curve saturates. On Aircraft and Cars at $M = 1$M, where the curves have largely flattened (Figure 2), ResNet-50 ends above ResNet-18 under KD+Syn (Table 1). We read this as a larger approximation floor $\epsilon^2_{\text{app}}(B)$ in Proposition 3.1 rather than a slower rate: the smaller student flattens out lower, and additional synthetic data does not close that part of the gap.

6.4 Extension to Semantic Segmentation

We extend our KD+Syn pipeline to semantic segmentation, which is essentially a pixel-wise classification. On Cityscapes (Cordts et al., 2016), we distill a PSPNet-ResNet18 student from two teachers

Table 3: **mIoU (%) on the Cityscapes validation split.** The student reaches 72.57 with cross-entropy alone, and the teachers 79.60 (PSPNet-ResNet101) and 86.09 (UperNet InternImage-XL). Synthetic data improves the student at every budget, and SynDrop improves it further.

Teacher	Extra data	KD (# extra images)		
		0	5k	10k
PSPNet ResNet-101	Syn.	74.34	74.65	74.99
UperNet InternImage-XL	Syn.	75.15	75.85	76.07
	+ SynDrop	75.15	76.26	76.19
	Real	75.15	77.04	78.19

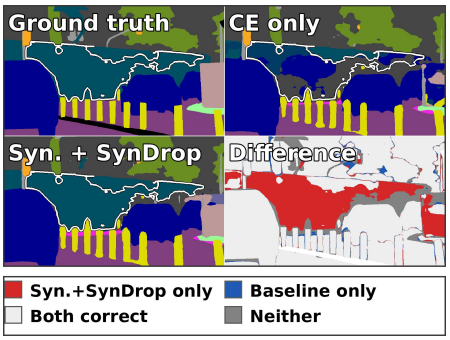

Figure 6: Qualitative example from Cityscapes *val*. Ours (Syn+SynDrop) produce segmentation masks much more similar to Ground Truth.

of differing capacity, PSPNet-ResNet101 (Zhao et al., 2017) and UperNet InternImage-XL (Xiao et al., 2018; Wang et al., 2023), with synthetic images sampled from an SDXL model LoRA-fine-tuned on the same dataset. We apply the KD objective of section 4.1 pixel-wise, weighting its two terms equally rather than as $\alpha : 1 - \alpha$, following prior work on segmentation distillation (Ali et al., 2026), and set $\tau = 2$. Every synthetic image follows the *No SynLabel* setting and contributes the KL term alone. Cityscapes further provides a pool of additional real images without fine annotations, which we treat as unlabeled data, allowing synthetic and real sources to be compared at matched quantities.

The scaling behavior observed in classification also holds for segmentation, where student mIoU increases with the number of synthetic images for both teachers (Table 3). SynDrop remains effective as well, improving over the corresponding synthetic KD runs for the UperNet InternImage-XL teacher. At matched budgets, real extra images remain the stronger source, but scaling synthetic data recovers part of the same gain.

7 Limitations and Future Work

Our study is bounded in four respects, each suggesting a direction for future work. First, every benchmark we study has a small training split, the regime in which the data bottleneck dominates; whether the same exponents hold once real data is abundant is untested, and extending the measurement to a large-scale setting such as ImageNet-1K would settle it. Second, for several teacher–student pairs the pool at which the fitted law predicts teacher parity lies far beyond what we could generate and train on, so parity there is an extrapolation we could not verify; more useful than a larger budget would be a method that brings parity within an affordable sample count. Third, we treat only discriminative tasks, and whether the recipe transfers to generative modeling, such as autoregressive language models and diffusion models, is entirely open. Finally, our generators are sampled independently of the student, so the pool carries no information about where the student is weak; several methods pursue such feedback (Hemmat et al., 2024; Askari-Hemmat et al., 2025), but our attempts to reproduce their reported gains in the fine-grained regime studied here were unsuccessful, and we leave generation-side sample selection under distillation to future work.

8 Conclusion

In this paper, we presented a recipe that substantially improves the performance of small neural network classifiers by combining generative data augmentation, knowledge distillation, and synthetic data dropping. Our pipeline offers a novel way to scale training-time compute in order to obtain a highly lightweight student model that incurs minimal inference cost. We expect that further development of our approach will benefit a wide range of applications under severe resource constraints.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods. The synthetic data sets we study are generated by diffusion models that we fine-tuned and sampled ourselves, as described in Section 6; these models are the object of our study rather than an assistant that helped us carry out the research, and no large language model took part in their training or in the generation of their samples. Apart from this, we have not used generative AI tools for any other subtask on ICLR's required-disclosure list. Additionally, we used generative AI tools to source and search for information, to identify relevant literature, and to edit the paper to improve readability. We have reviewed all AI-assisted work. LLM-generated code for our experiments was read and reviewed by the authors, and every result we report was produced by code in that reviewed state; every work we found through an LLM was retrieved and read by the authors before being cited; and each part of the paper was drafted by a human author and then polished with LLM assistance, with every resulting sentence checked by an author. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses only publicly available benchmark datasets (FGVC-Aircraft, Stanford Cars, Oxford Flowers-102, CUB-200-2011, and SUN397) under their respective licenses, and involves no human subjects or personal data. The synthetic images are generated by diffusion models (SDXL and EDM2) fine-tuned on these datasets and are used solely as training data for the student; they may nevertheless inherit biases present in the pretrained generators and in the benchmark datasets themselves. Since SDXL is pretrained on a web-scale corpus scraped from the internet, the generated images inherit the provenance, copyright, and content concerns of that corpus; we use them only as training data for a classifier and do not release them. We see no direct path from our method to harmful applications.

REPRODUCIBILITY STATEMENT

Our framework of knowledge distillation augmented with synthetic data and SynDrop is described in Section 4.1. Details of the datasets, data splits, preprocessing, and synthetic data generation, including the fine-tuning of the generative model, are provided in Appendix B.1. The teacher and student models, training hyperparameters, and evaluation protocols are also described in Appendix B.1.

REFERENCES

Muhammad Ali, Kevin Alexander Laube, Madan Ravi Ganesh, Lukas Schott, Niclas Popp, and Thomas Brox. The surprising effectiveness of canonical knowledge distillation for semantic segmentation. *arXiv preprint arXiv:2604.25530*, 2026.

Syed W. Ali, Angelos Angelopoulos, Denver Massey, Sarah Haddix, Alexander Georgiev, Joseph Goh, Rohan Wagle, Prakash Sarathy, James H. Anderson, and Ron Alterovitz. On the necessity of real-time principles in gpu-driven autonomous robots. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8086–8092, 2025. doi: 10.1109/ICRA55743.2025.11128627.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 254–263. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/arora18b.html.

Reyhane Askari-Hemmat, Mohammad Pezeshki, Elvis Dohmatob, Florian Bordes, Pietro Astolfi, Melissa Hall, Jakob Verbeek, Michal Drozdzal, and Adriana Romero-Soriano. Improving the scaling laws of synthetic data with deliberate practice. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=0LZRtvK871.

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. ISSN 2835-8856. URL https://openreview.net/forum?id=DlRsoxjyPm.

Abhipsa Basu, Aviral Gupta, Abhijnya Bhat, and R. Venkatesh Babu. Harnessing diffusion-generated synthetic images for fair image classification, 2025. URL https://arxiv.org/abs/2511.08711.

Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10925–10934, June 2022.

Alon Brutzkus and Amir Globerson. Why do larger models generalize better? A theoretical perspective via the XOR problem. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 822–830. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/brutzkus19b.html.

Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners, 2020. URL https://arxiv.org/abs/2006.10029.

Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training ... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7382–7392, June 2024.

Gongfan Fang, Jie Song, Xinchao Wang, Chengchao Shen, Xingen Wang, and Mingli Song. Contrastive model invertion for data-free knolwedge distillation. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 2374–2380. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/327. URL https://doi.org/10.24963/ijcai.2021/327. Main Track.

Dan Friedman and Adji Bousso Dieng. The Vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research (TMLR)*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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 *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=nUmCcZ5RKF.

Reyhane Askari Hemmat, Mohammad Pezeshki, Florian Bordes, Michal Drozdzal, and Adriana Romero-Soriano. Feedback-guided data synthesis for imbalanced classification. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=IHJ5OohGwr.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

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 (ICLR)*, 2022.

Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks, 2022. URL https://openreview.net/forum?id=dn4B7Mes2z.

Khawar Islam, Muhammad Zaigham Zaheer, Arif Mahmood, and Karthik Nandakumar. Diffusemix: Label-preserving data augmentation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27621–27630, 2024.

Pengzhan Jin, Lu Lu, Yifa Tang, and George Em Karniadakis. Quantifying the generalization error in deep learning in terms of data distribution and neural network smoothness. *Neural Networks*, 130:85–99, 2020. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2020.06.024. URL https://www.sciencedirect.com/science/article/pii/S0893608020302392.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 554–561, 2013.

Daiqing Li, Huan Ling, Amlan Kar, David Acuna, Seung Wook Kim, Karsten Kreis, Antonio Torralba, and Sanja Fidler. Dreamteacher: Pretraining image backbones with deep generative models, 2023a. URL https://arxiv.org/abs/2307.07487.

Wei Li, Shitong Shao, Weiyan Liu, Ziming Qiu, Zhihao Zhu, and Wei Huan. What role does data augmentation play in knowledge distillation? In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pp. 2204–2220, December 2022.

Zheng Li, Yuxuan Li, Penghai Zhao, Renjie Song, Xiang Li, and Jian Yang. Is synthetic data from diffusion models ready for knowledge distillation? *arXiv preprint arXiv:2305.12954*, 2023b.

Tzu-Hsuan Lin, Chien-Ta Chang, and Alan Putranto. Tiny machine learning empowers climbing inspection robots for real-time multiobject bolt-defect detection. *Engineering Applications of Artificial Intelligence*, 133:108618, 2024. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2024.108618. URL https://www.sciencedirect.com/science/article/pii/S0952197624007760.

Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks, 2017. URL https://arxiv.org/abs/1710.07535.

Liangchen Luo, Mark Sandler, Zi Lin, Andrey Zhmoginov, and Andrew Howard. Large-scale generative data-free distillation, 2020. URL https://arxiv.org/abs/2012.05578.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, 2013. URL https://arxiv.org/abs/1306.5151.

Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. The role of over-parametrization in generalization of neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BygfghAcYX.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729, 2008. doi: 10.1109/ICVGIP.2008.47.

Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14532–14542, 2022.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations (ICLR)*, 2024.

Aniket Roy, Anshul Shah, Ketul Shah, Anirban Roy, and Rama Chellappa. Cap2aug: Caption guided image data augmentation. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 9107–9117, February 2025.

Justin Sahs, Ryan Pyle, Aneel Damaraju, Josue Ortega Caro, Onur Tavaslioglu, Andy Lu, Fabio Anselmi, and Ankit B. Patel. Shallow univariate relu networks as splines: Initialization, loss surface, hessian, and gradient flow dynamics. *Frontiers in Artificial Intelligence*, Volume 5 - 2022, 2022. ISSN 2624-8212. doi: 10.3389/frai.2022.889981. URL https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2022.889981.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2022.

George Stein, Jesse C. Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L. Caterini, J. Eric T. Taylor, and Gabriel Loaiza-Ganem. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Richard S. Sutton. The bitter lesson, March 2019. URL http://www.incompleteideas.net/IncIdeas/BitterLesson.html. March 13, 2019.

Brandon Trabucco, Kyle Doherty, Max A Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ZWzUA9zeAg.

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.

Huan Wang, Suhas Lohit, Michael Jeffrey Jones, and Yun Fu. What makes a "good" data augmentation in knowledge distillation - a statistical perspective. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=6avZnPpk7m9.

Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14408–14419. IEEE, 2023.

Zhicai Wang, Longhui Wei, Tan Wang, Heyu Chen, Yanbin Hao, Xiang Wang, Xiangnan He, and Qi Tian. Enhance image classification via inter-class image mixup with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17223–17233, June 2024.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3485–3492, 2010.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European conference on computer vision*, pp. 432–448. Springer, 2018.

Kunran Xu, Lai Rui, Yishi Li, and Lin Gu. Feature normalized knowledge distillation for image classification. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, pp. 664–680, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58595-2.

I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification, 2019. URL https://arxiv.org/abs/1905.00546.

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 *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Zhuoran Yu, Chenchen Zhu, Sean Culatana, Raghuraman Krishnamoorthi, Fanyi Xiao, and Yong Jae Lee. Diversify, don't fine-tune: Scaling up visual recognition training with synthetic images. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=YCt8lsIDwA. Featured Certification.

Jianhao Yuan, Jie Zhang, Shuyang Sun, Philip Torr, and Bo Zhao. Real-fake: Effective training data synthesis through distribution matching. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=svIdLLZpsA.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890, 2017.

Chong Zhou, Chenchen Zhu, Yunyang Xiong, Saksham Suri, Fanyi Xiao, Lemeng Wu, Raghuraman Krishnamoorthi, Bo Dai, Chen Change Loy, Vikas Chandra, and Bilge Soran. Edgetam: On-device track anything model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13832–13842, June 2025.

A Proof of Theorems

A.1 Teacher-function recovery

Setup and assumptions. Let $\mathcal{X}$ be a measurable input space, let P_X be the target input distribution, and let Q be the distillation input distribution. Let $\hat{f}_T : \mathcal{X} \to \mathbb{R}$ be a measurable teacher function, and let K_S be a measurable kernel with a separable reproducing kernel Hilbert space $\mathcal{H}_S$.

The teacher, student kernel, and distillation distribution may be constructed using the same labeled training set D_n. Throughout, condition on D_n and any randomness used to construct these objects, so that $\hat{f}_T$, K_S, and Q remain fixed as the number of additional distillation inputs increases.

Assume the following conditions.

(A1) **Bounded student kernel.** For some $\kappa < \infty$,

$$\sup_{x \in \mathcal{X}} K_S(x, x) \le \kappa^2. \tag{2}$$

(A2) **Bounded teacher outputs.** For some $C_T < \infty$,

$$|\hat{f}_T(x)| \le C_T \qquad Q\text{-almost surely.} \tag{3}$$

(A3) **Coverage of the target distribution.** For some $\rho < \infty$,

$$P_X \ll Q, \qquad \left\| \frac{dP_X}{dQ} \right\|_{L_\infty(Q)} \le \rho. \tag{4}$$

For a fixed radius $0 < B < \infty$, define the admissible student hypothesis class, its population distillation loss, and the corresponding approximation error by

$$\mathcal{F}_B := \{ f \in \mathcal{H}_S : \|f\|_{\mathcal{H}_S} \le B \}, \tag{5}$$

$$L_Q(f) := \mathbb{E}_{Z \sim Q} \left[\big(f(Z) - \hat{f}_T(Z) \big)^2 \right], \tag{6}$$

$$\epsilon^2_{\mathrm{app}}(B) := \inf_{f \in \mathcal{F}_B} L_Q(f). \tag{7}$$

No exact realizability assumption is imposed; in particular, $\hat{f}_T$ need not belong to $\mathcal{H}_S$.

Draw additional inputs $Z_1, \dots, Z_m \overset{\text{i.i.d.}}{\sim} Q$ and query the teacher for their targets. Let $\hat{f}_{S,m}$ be any measurable empirical minimizer satisfying

$$\hat{f}_{S,m} \in \arg\min_{f \in \mathcal{F}_B} \widehat{L}_m(f), \tag{8}$$

$$\widehat{L}_m(f) := \frac{1}{m} \sum_{j=1}^{m} \big(f(Z_j) - \hat{f}_T(Z_j) \big)^2. \tag{9}$$

Set $A := \kappa B + C_T$ and

$$\Delta_m(\delta) := \frac{4A\kappa B}{\sqrt{m}} + A^2 \sqrt{\frac{\log(2/\delta)}{2m}}, \tag{10}$$

$$U_m(\delta) := \rho \big[\epsilon^2_{\mathrm{app}}(B) + 2\Delta_m(\delta) \big]. \tag{11}$$

Theorem A.1 (Function Approximation with Synthetic Data)**.** *Under the setup and Assumptions* (A1)–(A3) *above, the infimum in equation 7 is attained by some*

$$f^\star_B \in \arg\min_{f \in \mathcal{F}_B} L_Q(f), \tag{12}$$

which is unique up to Q-almost-sure equality. For every $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the additional distillation inputs, the following inequalities hold simultaneously:

$$0 \le L_Q(\hat{f}_{S,m}) - \epsilon^2_{\mathrm{app}}(B) \le 2\Delta_m(\delta), \tag{13}$$

$$\|\hat{f}_{S,m} - f^\star_B\|^2_{L_2(P_X)} \le \rho \left[L_Q(\hat{f}_{S,m}) - \epsilon^2_{\mathrm{app}}(B) \right] \le 2\rho\Delta_m(\delta), \tag{14}$$

$$\|\hat{f}_{S,m} - \hat{f}_T\|^2_{L_2(P_X)} \le \rho\epsilon^2_{\mathrm{app}}(B) + 2\rho\Delta_m(\delta) = U_m(\delta), \tag{15}$$

The last bound separates the coverage-adjusted approximation upper bound $\rho\epsilon_{\text{app}}^2(B)$ from an $O(m^{-1/2})$ estimation term.

Proof. We first establish the existence and uniqueness of the population best approximation. By the reproducing property and Assumption (A1),

$$|f(x)| \le \|f\|_{\mathcal{H}_S}\sqrt{K_S(x,x)} \le \kappa\|f\|_{\mathcal{H}_S}, \qquad f \in \mathcal{H}_S. \tag{16}$$

Thus the natural embedding $J : \mathcal{H}_S \to L_2(Q)$ is bounded and linear. The closed ball $\mathcal{F}_B$ is weakly compact in $\mathcal{H}_S$, and $f \mapsto \|Jf - \hat{f}_T\|_{L_2(Q)}^2$ is weakly lower semicontinuous. Therefore, the minimum in equation 12 is attained. Since $\mathcal{F}_B$ is convex and the squared distance is strictly convex in the $L_2(Q)$ prediction, all minimizers agree Q-almost surely.

We next establish uniform convergence of the empirical distillation loss. For every $f \in \mathcal{F}_B$,

$$|f(x) - \hat{f}_T(x)| \le \kappa B + C_T = A \qquad Q\text{-almost surely.} \tag{17}$$

Consequently, the loss functions

$$\ell_f(z) := \big(f(z) - \hat{f}_T(z)\big)^2 \tag{18}$$

take values in $[0, A^2]$ almost surely.

Let $\sigma_1, \dots, \sigma_m$ be independent Rademacher random variables. Conditional on the inputs, the empirical Rademacher complexity of $\mathcal{F}_B$ satisfies

$$\begin{aligned}\widehat{\mathfrak{R}}_m(\mathcal{F}_B) &:= \mathbb{E}_\sigma\left[\sup_{f\in\mathcal{F}_B}\frac{1}{m}\sum_{j=1}^m \sigma_j f(Z_j)\right] \\ &= \frac{B}{m}\mathbb{E}_\sigma\left\|\sum_{j=1}^m \sigma_j K_S(Z_j,\cdot)\right\|_{\mathcal{H}_S} \\ &\le \frac{B}{m}\left(\sum_{j=1}^m K_S(Z_j,Z_j)\right)^{1/2} \le \frac{\kappa B}{\sqrt{m}}.\end{aligned} \tag{19}$$

For each z, the function

$$\phi_z(u) := \big(u - \hat{f}_T(z)\big)^2 \tag{20}$$

is $2A$-Lipschitz on $[-\kappa B, \kappa B]$. Apply the contraction inequality to $\phi_z(u) - \phi_z(0)$, which vanishes at zero. The centering term is independent of f and has zero expected Rademacher sum. Hence, for the loss class $\mathcal{L}_B := \{\ell_f : f \in \mathcal{F}_B\}$,

$$\widehat{\mathfrak{R}}_m(\mathcal{L}_B) \le 2A\,\widehat{\mathfrak{R}}_m(\mathcal{F}_B) \le \frac{2A\kappa B}{\sqrt{m}}. \tag{21}$$

It remains to convert the Rademacher bound into a high-probability uniform deviation. Define the one-sided empirical processes

$$\begin{aligned}\Phi &:= \sup_{f\in\mathcal{F}_B}\big(L_Q(f) - \widehat{L}_m(f)\big), \\ \Psi &:= \sup_{f\in\mathcal{F}_B}\big(\widehat{L}_m(f) - L_Q(f)\big),\end{aligned} \tag{22}$$

so that $\sup_{f\in\mathcal{F}_B}|L_Q(f) - \widehat{L}_m(f)| = \max(\Phi, \Psi)$. A standard symmetrization argument yields

$$\mathbb{E}[\Phi] \le 2\,\mathbb{E}\big[\widehat{\mathfrak{R}}_m(\mathcal{L}_B)\big] \le \frac{4A\kappa B}{\sqrt{m}}, \tag{23}$$

and the same bound holds for $\mathbb{E}[\Psi]$. The first factor of 2 is the usual comparison of the expected uniform deviation with the Rademacher complexity; the second factor of 2 is already present in equation 21.

Next, replacing any single sample Z_j changes $\widehat{L}_m(f)$ by at most A^2/m for every $f \in \mathcal{F}_B$, and therefore changes both Φ and Ψ by at most A^2/m. McDiarmid's bounded-differences inequality thus gives, for every $t > 0$,

$$\Pr\big(\Phi - \mathbb{E}[\Phi] \geq t\big) \leq \exp\left(-\frac{2mt^2}{A^4}\right),$$
$$\Pr\big(\Psi - \mathbb{E}[\Psi] \geq t\big) \leq \exp\left(-\frac{2mt^2}{A^4}\right). \tag{24}$$

Choosing $t = A^2\sqrt{\log(2/\delta)/(2m)}$ makes each tail at most $\delta/2$. Combining equation 23 and equation 24 and applying a union bound, we obtain that with probability at least $1 - \delta$,

$$\sup_{f \in \mathcal{F}_B} \left| L_Q(f) - \widehat{L}_m(f) \right| \leq \frac{4A\kappa B}{\sqrt{m}} + A^2\sqrt{\frac{\log(2/\delta)}{2m}} = \Delta_m(\delta). \tag{25}$$

On this event, empirical optimality gives

$$\begin{aligned} L_Q(\hat{f}_{S,m}) &\leq \widehat{L}_m(\hat{f}_{S,m}) + \Delta_m(\delta) \\ &\leq \widehat{L}_m(f_B^\star) + \Delta_m(\delta) \\ &\leq L_Q(f_B^\star) + 2\Delta_m(\delta) \\ &= \epsilon_{\mathrm{app}}^2(B) + 2\Delta_m(\delta). \end{aligned} \tag{26}$$

The reverse inequality $L_Q(\hat{f}_{S,m}) \geq \epsilon_{\mathrm{app}}^2(B)$ holds by definition, proving equation 13.

To control convergence to $f_B^\star$, use the first-order optimality condition over the convex set $\mathcal{F}_B$:

$$\left\langle f_B^\star - \hat{f}_T,\, f - f_B^\star \right\rangle_{L_2(Q)} \geq 0, \qquad f \in \mathcal{F}_B. \tag{27}$$

Expanding the population loss therefore yields

$$\begin{aligned} L_Q(f) - L_Q(f_B^\star) &= \|f - f_B^\star\|_{L_2(Q)}^2 + 2\left\langle f - f_B^\star,\, f_B^\star - \hat{f}_T \right\rangle_{L_2(Q)} \\ &\geq \|f - f_B^\star\|_{L_2(Q)}^2. \end{aligned} \tag{28}$$

In particular,

$$\|\hat{f}_{S,m} - f_B^\star\|_{L_2(Q)}^2 \leq L_Q(\hat{f}_{S,m}) - \epsilon_{\mathrm{app}}^2(B) \leq 2\Delta_m(\delta). \tag{29}$$

By Assumption (A3), every $h \in L_2(Q)$ satisfies

$$\|h\|_{L_2(P_X)}^2 = \int_{\mathcal{X}} h(x)^2 \frac{dP_X}{dQ}(x)\, dQ(x) \leq \rho \|h\|_{L_2(Q)}^2. \tag{30}$$

Applying equation 30 first to $h = \hat{f}_{S,m} - f_B^\star$ and then to $h = \hat{f}_{S,m} - \hat{f}_T$ proves equation 14 and equation 15, respectively. □

B Experimental Details

B.1 Full Experimental Setup

Datasets. Table 4 reports the split sizes of the five benchmarks we use. We train models exclusively on the training split, which we augment with a pool of class-conditional synthetic images. Real and synthetic images are concatenated into the single training set $\mathcal{D}_{\mathrm{aug}}$ of Section 4.1; the validation and test splits contain real images only.

Table 4: Details of fine-grained visual classification datasets used in the experiments. Datasets marked with * do not provide an official validation split; for those we partition the official training split into the training and validation sets reported here.

Dataset	#Train	#Val	#Test	#Class
FGVC-Aircraft Maji et al. (2013)	3334	3333	3333	100
Stanford Cars* Krause et al. (2013)	6509	1635	8041	196
Oxford Flowers-102 Nilsback & Zisserman (2008)	1020	1020	6149	102
CUB-200-2011* Wah et al. (2011)	4794	1200	5794	200
SUN397* Xiao et al. (2010)	17865	1985	19850	397

Synthetic data generation. We consider two latent diffusion models that differ in how they are conditioned: SDXL (Podell et al., 2024), a text-conditional model, and EDM2-S (Karras et al., 2024), a class-conditional model. For each target dataset, both generators are adapted on the training split alone with Low-Rank Adaptation (LoRA; Hu et al., 2022), keeping the publicly available pretrained backbone fixed and training only the LoRA parameters. Using the adapted generators, we synthesize class-balanced pools with an equal number of images per class, and we vary the pool size M to study its effect. All synthetic images for image classification are generated at 512×512 resolution and stored as JPEG files with quality 95 and chroma subsampling 2.

Models and training. We distill a fine-tuned ViT-Large teacher into ResNet-18 and ResNet-50 students with vanilla logit distillation (Hinton et al., 2015), using the objective of Section 4.1. Teacher and students alike start from ImageNet-1K pretrained weights; the teacher is then fine-tuned on the training split of each target dataset and frozen. All images are resized to 224×224 before being passed to the teacher or student. The loss-balancing weight α and the temperature τ are selected by a hyperparameter sweep performed separately for each dataset, choosing the configuration with the highest validation accuracy in the real-data-only setting; the selected (α, τ) is then held fixed across all synthetic-augmentation experiments. All students are trained for 100 epochs in bf16 mixed precision with SGD (momentum 0.9), a batch size of 128, an initial learning rate of 0.01, weight decay of 1×10^{-5}, and an epoch-wise cosine-annealing schedule. We report the top-1 accuracy of the final-epoch model on the real test split; no validation-based checkpoint selection is applied.

B.2 Computational Cost

To give a sense of cost, we time each stage of the pipeline on a single NVIDIA RTX PRO 6000 Blackwell Server Edition GPU. On this device, fine-tuning the SDXL LoRA takes 2–4 hours and fine-tuning the ViT-L teacher 3–4 hours, depending on the dataset; each is done once per dataset and reused across every pool size. Sampling is the most expensive stage: a 1M-image pool costs 89–92 GPU-hours. It parallelizes trivially, though, since the samples are independent, so we shard it across four GPUs and obtain a pool in under a day of wall-clock time. Training a student for 100 epochs at $M = 1$M takes 28–31 hours, which SynDrop at $E_{\text{drop}} = 50$ cuts to roughly half.

B.3 Scaling of the Test Loss

Section 4.2 states a scaling law for both the test loss and the test accuracy; the main text reports the accuracy, and Figure 7 reports the loss, fitted by the same nonlinear least squares procedure.

B.4 Data Leakage Check for Diffusion Models

We tested our generated images for test-set leakage and found it unlikely, so we attribute the performance gains we report to generalization rather than to memorized test content. The question is worth asking because SDXL is pretrained on a large web-scale corpus whose contents are publicly disclosed (Schuhmann et al., 2022). If such a corpus includes the test sets of the fine-grained benchmarks we study, a fine-tuned model may generate test-set-like images.

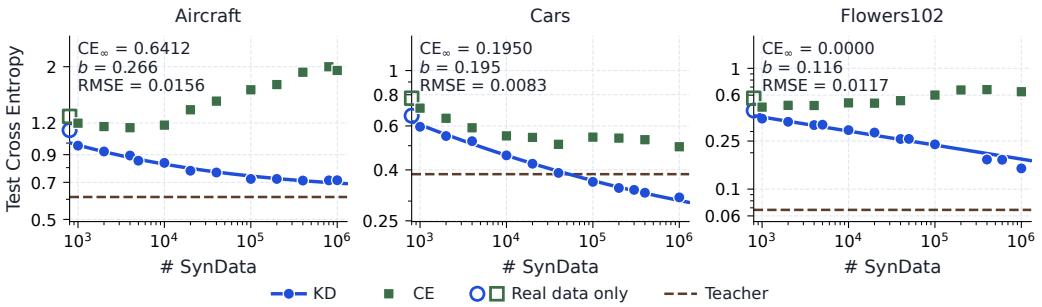

Figure 7: Test cross-entropy loss of the same runs as Figure 2. The loss follows $\mathcal{L}_{\text{test}}(M) \approx \mathcal{L}_\infty + aM^{-b}$ with the same qualitative separation between KD and cross-entropy.

We test for this with SSCD (Pizzi et al., 2022), a self-supervised descriptor trained to detect image copies under heavy augmentation. For every image in our full SDXL pools—5M images across the five datasets—we compute the maximum cosine similarity to any real test image, regardless of its class. As calibration we compute the same statistic for the real training split. Table 5 reports both. Only 5 images of generated SUN397 were above the conventional near-duplicate threshold of 0.7, and every pool stays below the corresponding train–test similarity. Since both the severity and the number of similar images are low, we conclude that our experiments are free of test-set leakage.

It is worth mentioning that the audit unintentionally revealed leakage within the real data itself. Stanford Cars and Oxford Flowers-102 both contain near-exact train–test pairs. This redundancy is a property of the benchmarks rather than of our pipeline, and it applies equally to every method trained on their real training splits.

Table 5: Maximum SSCD cosine similarity to the real test split. Almost no synthetic image crosses the 0.7 near-duplicate threshold, while the real train–test baseline includes near-exact matches.

	Synthetic → test		Real train → test		
Dataset	max	# ≥ 0.7	max	# ≥ 0.7	# = 1.0
FGVC-Aircraft	0.569	0	0.860	75	0
Stanford Cars	0.604	0	1.000	118	13
Oxford Flowers-102	0.634	0	1.000	54	1
CUB-200-2011	0.585	0	0.939	13	0
SUN397	0.740	5	0.982	117	0

B.5 SynDrop under cross-entropy training

Figure 8 repeats the sweep of Section 6.2 ($E_{\text{drop}} \in \{10, 20, \ldots, 90\}$ with $E = 100$, together with no-drop training) under the cross-entropy objective, i.e., the student is trained on the synthetic labels $\tilde{y}$ without a teacher. The pool size M varies over $\{1\text{K}, 10\text{K}, 100\text{K}, 200\text{K}, 400\text{K}, 800\text{K}, 1\text{M}\}$. SynDrop helps here as well, and by a larger margin: at $M \geq 100$K the best drop epoch exceeds no-drop training by $+6.6$ to $+12.0$ points on Aircraft and $+2.7$ to $+3.2$ points on CUB200. Two differences from the KD case stand out.

First, there is no window of drop epochs. The earliest drop, $E_{\text{drop}} = 10$, is best at every $M \geq 100$K on both datasets, accuracy decreases monotonically with later drops, and dropping within the final ten epochs falls below no-drop training. Second, SynDrop changes the scaling behavior itself. Under no-drop training, accuracy barely increases with the pool size: on Aircraft, no-drop training reaches 72.1% at 10K and 70.8% at 1M, and on CUB200 it increases only slowly. Along the best-drop chain, accuracy rises steadily with M on both datasets, to 82.9% on Aircraft and 78.8% on CUB200 at $M = 1$M.

Both differences follow from the labels. Cross-entropy treats the label the generator used as ground truth, so each image the generator assigns to the wrong class pushes the student toward that class, and the damage accumulates with every additional pass over the pool. The synthetic data are there-

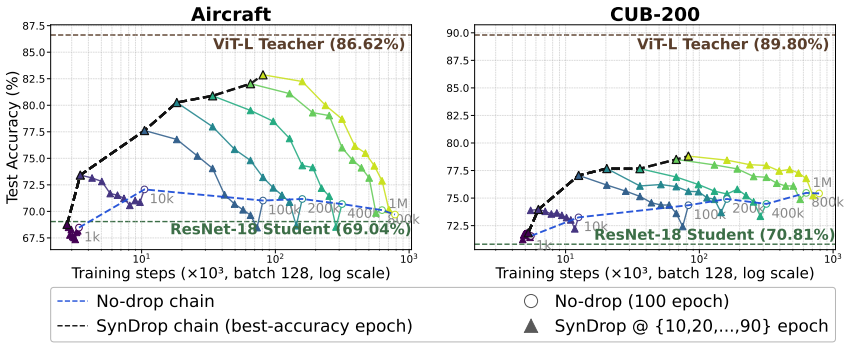

Figure 8: SynDrop under cross-entropy training on Aircraft (left) and CUB200 (right) with a ResNet-18 student, in the same layout as Figure 3: for each pool size M (colors), triangles show training steps and final test accuracy for $E_{\text{drop}} \in \{10, 20, \ldots, 90\}$ with $E = 100$, from left (early drop) to right, and the rightmost point (circle) is no-drop training. The black chain connects the best drop epoch for each pool. The teacher line is shown only for reference; no teacher is used. Unlike KD, the earliest drop is best at every $M \geq 100$K, and accuracy falls monotonically with later drops.

fore most useful in the first few epochs, where they shape the representation, and harmful afterwards. Under KD the teacher's soft targets are computed on the image actually generated, so mislabeled images cost little, and the student can afford to train on the pool for longer. This is the same mechanism that makes the optimal guidance scale lower under KD than under cross-entropy (Section 6), and it explains why the drop epoch is critical here but forgiving under KD.