ON-THE-FLY DATA AUGMENTATION VIA GRADIENT-GUIDED AND SAMPLE-AWARE INFLUENCE ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Data augmentation has been widely employed to improve the generalization of deep neural networks. Most existing methods apply fixed or random transformations. However, we find that sample difficulty evolves along with the model's generalization capabilities in dynamic training environments. As a result, applying uniform or stochastic augmentations, without accounting for such dynamics, can lead to a mismatch between augmented data and the model's evolving training needs, ultimately degrading training effectiveness. To address this, we introduce SADA, a Sample-Aware Dynamic Augmentation that performs on-the-fly adjustment of augmentation strengths based on each sample's evolving influence on model optimization. Specifically, we estimate each sample's influence by projecting its gradient onto the accumulated model update direction and computing the temporal variance within a local training window. Samples with low variance, indicating stable and consistent influence, are augmented more strongly to emphasize diversity, while unstable samples receive milder transformations to preserve semantic fidelity and stabilize learning. Our method is lightweight, which does not require auxiliary models or policy tuning. It can be seamlessly integrated into existing training pipelines as a plug-and-play module. Experiments across various benchmark datasets and model architectures show consistent improvements of SADA, including +7.3% on fine-grained tasks and +4.3% on long-tailed datasets, highlighting the method's effectiveness and practicality. Code will be made publicly available upon publication.

1 INTRODUCTION

Data augmentation has been widely adopted for improving the generalization performance of deep neural networks (Yang et al., 2022; Shorten & Khoshgoftaar, 2019; Iglesias et al., 2023). Despite its effectiveness, most existing DA approaches remain static, non-adaptive, and sample-agnostic: they apply either fixed or randomly sampled transformations to all data uniformly, regardless of the evolving difficulty of individual samples or the dynamic learning state of the model in a dynamic training environment (Müller & Hutter, 2021; Cubuk et al., 2019; 2020; Li et al., 2020). For instance, methods such as Cutout (DeVries & Taylor, 2017), AdvMask (Yang et al., 2023), and Mixup (Zhang et al., 2018) generate diverse training data by randomly sampling augmentation parameters. Automatic methods, such as AutoAugment (Cubuk et al., 2019), RandAugment (Cubuk et al., 2020), and DeepAA (Zheng et al., 2022), search for dataset-specific augmentation policy space before training begins and then apply these fixed policies during training. However, this design overlooks a crucial aspect of deep model training: the optimization landscape and the difficulty of individual samples evolving in dynamic training environments. Some samples become easy to fit early on and require increased diversity to avoid redundancy, while others remain hard or unstable and should be preserved in their semantic form to support model refinement. Applying uniform augmentations across these heterogeneous cases introduces a mismatch between augmentation strength and training needs, potentially resulting in noisy updates, degraded sample utility, and suboptimal convergence. Furthermore, many methods often require manual policy tuning or dataset-specific search, which limits scalability across different datasets and architectures (Cubuk et al., 2019; 2020; Yang et al., 2024b). Adaptive augmentation approaches have emerged, but they typically involve bi-level

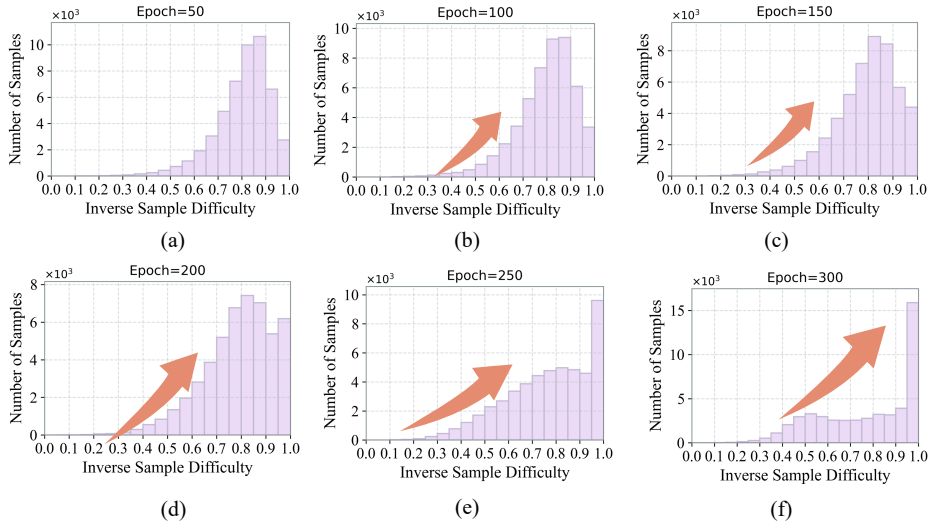


Figure 1: **Evolution of Sample Difficulty Across Training Epochs.** The distribution of sample difficulty evolves dynamically throughout training. A growing proportion of samples becomes easier (higher inverse sample difficulty values), particularly in later epochs. This dynamic trend highlights the necessity of dynamic and sample-aware augmentation strategies during training. Inverse sample difficulty: the reciprocal of sample difficulty.

optimization (Hou et al., 2023), auxiliary models (Suzuki, 2022; Yang et al., 2025), or large search spaces (Bekor et al., 2024), significantly increasing training complexity and resource demand. Thus, a pressing question emerges: *Can we develop an on-the-fly augmentation mechanism that dynamically adapts training data to a model's evolving learning dynamics without sacrificing scalability or efficiency.*

In this paper, we propose SADA, a Sample-Aware Dynamic Augmentation method that performs on-the-fly adjustment of augmentation strength based on each sample's evolving influence during training. Unlike many existing methods that optimize augmentation operations (Bekor et al., 2024; Cubuk et al., 2019), our method uses a unified dataset- and model-agnostic augmentation space (refer to Table 8) and directly modulates augmentation strength. This design offers three benefits: 1). reducing the complexity of the decision space and ensuring efficient online training, 2). providing a more interpretable and fine-grained control over the trade-off between semantic consistency and diversity (Yang et al., 2024a), and 3). eliminating the need for manually crafted or optimization-required dataset-specific augmentation policies and enhancing scalability. To quantify each sample's influence, we project its instantaneous gradient onto the direction of the accumulated model update, thereby capturing how much the sample contributes to the prevailing optimization trajectory. The gradients can be naturally obtained during the standard forward and backward passes, ensuring high efficiency. Furthermore, we compute the temporal variance of this projected influence within a local training window (e.g., 5 epochs), which serves as a proxy for the stability of a sample's learning dynamics. When a sample exhibits consistently low variance, indicating a stable contribution to learning, more substantial augmentation is assigned to promote diversity and avoid overfitting to redundant patterns. Conversely, samples with high variance, suggesting unstable or ambiguous influence, are augmented more conservatively to preserve semantic fidelity and support robust learning. In this way, our method dynamically tailors augmentation magnitudes for each sample based on its training-stage-aware influence. As illustrated in Figure 1, our gradient-guided influence estimation reveals that sample difficulty continuously evolves throughout training: while more samples gradually become easier to fit as the model learns, a small subset remains persistently challenging. By selectively increasing diversity for easier samples and preserving the core semantics of difficult ones, our framework improves generalization while mitigating the risk of introducing ambiguous or disruptive augmentations, highlighting the benefits of our sample-aware, dynamic augmentation.

Experiment results across a variety of benchmark datasets and deep architectures demonstrate consistent and robust performance improvements. On benchmark datasets such as CIFAR-10/100 (Krizhevsky et al., 2009), Tiny-ImageNet (Chrabaszcz et al., 2017), and ImageNet-

1k (Krizhevsky et al., 2017), our approach consistently outperforms existing data augmentation methods. Additionally, we demonstrate strong generalization across different model architectures, including ResNet-based (He et al., 2016) and Vision Transformer (ViT) (Dosovitskiy et al., 2020)-based backbones, etc. Thus, our method can be seamlessly integrated as a plug-and-play component without any modifications to model structures or training schedules. On more challenging long-tailed datasets such as ImageNet-LT and Places-LT (Liu et al., 2019), models trained with our method achieve substantial gains, improving top-1 accuracy by over 4.3% under the closed-set evaluation of ImageNet-LT, highlighting its robustness in imbalanced data scenarios.

Our main contributions can be summarized as follows: **(1)** We propose a lightweight, on-the-fly data augmentation framework that adjusts the augmented data based on the sample-aware evolving influence, without relying on auxiliary models or costly optimization procedures. **(2)** Our method explicitly captures the interplay between data and model by quantifying each sample's contribution to model optimization updates via gradient-guided influence estimation, aligning augmented data with the model's instantaneous learning dynamics. **(3)** Extensive experiments across diverse datasets and architectures demonstrate that our approach serves as a play-and-plug module, consistently improving generalization while maintaining training efficiency.

2 Related Work

Data augmentation has long been a fundamental technique for mitigating overfitting and improving the generalization capability of deep neural networks. DA methods have evolved from simple, hand-crafted transformations to more adaptive and automated strategies. It has evolved through multiple methodological paradigms. Early approaches primarily involved applying fundamental transformations, such as rotation, flipping, or cropping (Krizhevsky et al., 2012; Yang et al., 2022), to increase dataset diversity and model robustness. Subsequent advancements focus on developing more sophisticated transformation strategies tailored to specific data characteristics. DA methods can be broadly categorized into image deletion-based, image fusion-based, and automatic policy-based strategies (Müller & Hutter, 2021; Yang et al., 2024b).

Image Deletion-based Methods. Cutout (DeVries & Taylor, 2017) introduces regularization by randomly removing square regions from images. GridMask (Chen et al., 2020) generates resolution-matched masks for element-wise multiplication with images. Hide-and-Seek (HaS) (Singh & Lee, 2017) generalizes this idea by partitioning images into grids and stochastically masking subregions. Random Erasing (Zhong et al., 2020) further occludes rectangular areas without resizing. Moreover, AdvMask (Yang et al., 2023) generates learned or structure-aware masking to explicitly target semantic regions, encouraging the model to discover alternative discriminative cues.

Image Fusion-based Methods. Fusion-based augmentation synthesizes training samples by blending information across multiple instances. Mixup (Zhang et al., 2018) synthesizes samples via linear interpolation of pixel values and labels across image pairs. However, its indiscriminate blending may produce visually incoherent samples. CutMix (Yun et al., 2019) improves this by replacing rectangular regions between images, preserving spatial structure while introducing inter-sample variability. However, it may still obscure critical semantic content with irrelevant patches. Some improved variants, such as Attentive CutMix (Walawalkar et al., 2020) and PuzzleMix (Kim et al., 2020), incorporate saliency awareness. Despite their effectiveness, these methods typically rely on manually tuned parameters, with limited awareness of the model's evolving training dynamics, potentially limiting the adaptability and optimization efficiency.

Automated Augmentation Methods. Automated DA approaches define an augmentation operation space and search for optimal operations and magnitudes. During training, the augmentation operation and corresponding magnitudes are randomly sampled from the pre-defined space. AutoAugment (AA) (Goodfellow et al., 2015) uses reinforcement learning with an RNN controller to predict transformation sequences. Population-Based Augmentation (PBA) (Ho et al., 2019) integrates genetic algorithms with parallel network training, while Fast AutoAugment (Lim et al., 2019) employs Bayesian optimization to discover effective augmentation sequences across partitioned datasets. While powerful, these methods often incur high computational cost and are static once learned. RandAugment (Cubuk et al., 2020) and TrivialAugment (Müller & Hutter, 2021) simplify the parameter spaces through randomized policy selection. EntAugment (Yang et al., 2024b) uses entropy information derived from model snapshots to adjust the augmentation transformations. While effec-

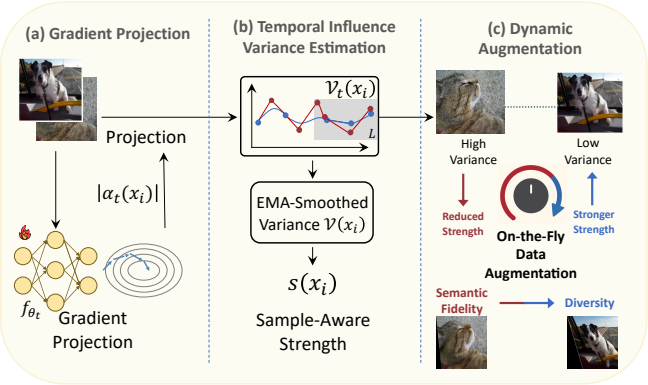


Figure 2: **Overview of Gradient-Guided On-the-Fly Data Augmentation.** At epoch t, we quantify the sample's influence on model optimization updates and estimate its stability. The augmentation strength is then adaptively adjusted based on this interplay between model training progress and sample difficulty.

tive, its decisions rely on entropy values extracted from instantaneous model snapshots, which can fluctuate due to the inherent instability of model training. Moreover, ParticleAugment (Tsaregorodtsev & Belagiannis, 2023) proposes a particle filtering scheme for the augmentation policy search. Gradient-based DAS approaches formulate differentiable search spaces, enabling optimization of augmentation strategies. MADAO (Hataya et al., 2020) optimizes models and data augmentation policies simultaneously with Neumann series approximation of the gradients. DADA (Li et al., 2020) formulates data augmentation policy search as a sampling problem and relaxes it into a differentiable framework via Gumbel-Softmax reparameterization. Adversarial variants such as Adversarial AutoAugment (Zhang et al., 2019) and TeachAugment (Suzuki, 2022) generate challenging transformations by maximizing training loss. DDAS (Liu et al., 2021) exploits meta-learning with one-step gradient update and continuous relaxation to the expected training loss for efficient search, without relying on approximations like Gumbel Softmax. In addition, DeepAA (Zheng et al., 2022) progressively constructs multi-layer augmentation pipelines. FreeAugment (Bekor et al., 2024) defines four free degrees of data augmentation and jointly optimizes them. MADAug (Hou et al., 2023), SelectAugment (Lin et al., 2023), SLACK (Marrie et al., 2023), and MetaAugment (Hataya et al., 2022) optimize or learn sample-wise augmentation policies using various techniques, e.g., training an auxiliary policy network. Despite these advances, most existing automated methods overlook the intrinsic heterogeneity of training data difficulty and fail to adapt augmentation intensities dynamically during online training. In contrast, our methodology introduces a lightweight, gradient-based mechanism that samples influence during training and adaptively adjusts augmentation magnitudes in real time, enabling fine-grained, instance-aware data augmentation.

3 Our Proposed Method

Overview. As illustrated in Fig. 2, we propose an on-the-fly data augmentation method that adjusts sample-aware augmentation strength based on each sample's evolving influence on the model's optimization trajectory. Specifically, we project the sample-wise gradient onto the accumulated gradient direction to quantify its contribution to parameter updates. To assess the consistency of this contribution, we compute the variance of the projected values within a local training window and apply EMA smoothing. In this way, the augmentation strengths are dynamically determined in proportion to the stability of the sample's training influence. Samples with low variance, indicating stable influence, are assigned stronger augmentations to improve generalization, while high-variance samples receive milder augmentations to maintain semantic fidelity and stabilize training. In essence, our approach adjusts augmentation strength based on the interaction between the training data and the model's evolving optimization dynamics, thereby achieving dynamic augmentation. During training, we randomly select one augmentation operation from the augmentation space for each sample per epoch and dynamically modulate its strength, which is uniformly applied to various datasets.

Let's denote the whole dataset as $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^D$, $y_i \in \mathbb{R}^{1\times K}$, and K is the number of classes. The model f_θ is trained via gradient descent, updating parameters θ at step t as:

$$\theta_{t+1} = \theta_t - \eta \sum_{n=1, x_i \in \mathcal{D}}^{N} g_t(x_i), \tag{1}$$

where $g_t(x_i)$ is the gradient of the loss with respect to sample (x_i, y_i), and η is the learning rate.

During training, each sample contributes to the update of the model parameters via its gradient. For samples that are easier to learn, the loss converges rapidly, and their gradient magnitudes tend to stabilize. In contrast, more difficult and ambiguous samples often induce slower loss decay and exhibit persistently fluctuating gradients (Toneva et al., 2019; Zhang et al., 2017; Swayamdipta et al., 2020). To quantify a sample's alignment with the model's current optimization trajectory, we compute the projection of its gradient onto the accumulated update direction. Specifically, we focus on the projection value of the gradient in the direction of parameter updates. The norm of the projected vector is calculated as follows:

$$|\alpha_t(x_i)| = |\left\langle g_t(x_i), \theta_{t-1} - \theta_t \right\rangle|. \tag{2}$$

The projected value reflects how much a sample's gradient contributes to the direction of the model's parameter update.

To maintain high efficiency, we approximate the sample-wise gradient projection using first-order Taylor expansion, transforming the gradient-based formulation into a loss-based difference (Zhang et al., 2024). Specifically, the projected influence $\alpha_t(x_i)$ can be approximated as:

$$\begin{aligned} |\alpha_t(x_i)| &= \frac{1}{\eta} \left| (\theta_{t-1} - \theta_t) \nabla_{\theta_{t-1}} \ell\left(f_{\theta_{t-1}}(x_i), y_i\right) \right| \\ &\approx \frac{1}{\eta} \left| \ell\left(f_{\theta_t}(x_i), y_i\right) - \ell\left(f_{\theta_{t-1}}(x_i), y_i\right) \right|, \end{aligned} \tag{3}$$

where $\ell(\cdot)$ denotes the loss function (e.g., cross-entropy). This approximation reduces the need to compute inner products between gradients and parameter updates. In the case of classification tasks with cross-entropy loss, the per-sample loss difference across consecutive steps is given by:

$$\begin{aligned} \Delta\ell_{t-1}^n &= \ell(f_{\theta_t}(x_i), y_i) - \ell(f_{\theta_{t-1}}(x_i), y_i) \\ &= y_i^\top \cdot \log \frac{f_{\theta_t}(x_i)}{f_{\theta_{t-1}}(x_i)}. \end{aligned} \tag{4}$$

To generalize this formulation and enable a fully differentiable approximation, we replace the one-hot label with the soft target $f_{\theta_t}(x_i)^\top$, yielding a KL divergence between the model outputs at two consecutive steps:

$$\Delta\ell_{t-1}^n = f_{\theta_t}(x_i)^\top \cdot \log \frac{f_{\theta_t}(x_i)}{f_{\theta_{t-1}}(x_i)}. \tag{5}$$

This formulation efficiently captures the alignment between a sample's prediction dynamics and model update direction without computing explicit gradients.

To maintain high efficiency during training, we avoid complete historical gradient information and instead approximate sample influence using local training dynamics. Specifically, we compute the variance of sample-wise loss differences over a fixed-size window of the past L epochs, which is:

$$\mathcal{V}_t(x_i) = \sum_{t-L+1}^{t} \left\| |\Delta\ell_t^n| - \overline{|\Delta\ell_t^n|} \right\|^2, \tag{6}$$

where $\overline{|\Delta\ell_t^n|}$ denotes the average of the absolute loss differences within the window. This formulation provides a local, memory-efficient measure of influence variability and mitigates instability from single-step snapshot assessments. To smooth short-term fluctuations and emphasize recent training dynamics, we update the influence estimate using an exponential moving average:

$$\mathcal{V}(x_i) = \beta\mathcal{V}_t(x_i) + (1-\beta)\mathcal{V}(x_i), \tag{7}$$

where β is the decay coefficient, and both β and L are set as constants. In this way, the resulting influence score $\mathcal{V}(x_i)$ shows a proportional relationship with the sample difficulty. To scale the values

Table 1: Image classification accuracy (%) on CIFAR-10/100. * means results reported in the original paper (Müller & Hutter, 2021; Yang et al., 2024b).

Method	ResNet-44	ResNet-50	WRN-28-10	SS-26-32	ResNet-44	ResNet-50	WRN-28-10	SS-26-32
	CIFAR-10				CIFAR-100			
baseline	94.10±.40	95.66±.08	95.52±.11	94.90±.07*	74.80±.38*	77.41±.27*	78.96±.25*	76.65±.14*
RE	94.87±.16*	95.82±.17	96.92±.09	96.46±.13*	75.71±.25*	77.79±.32	80.57±.15	77.30±.18
RA	94.38±.22	96.25±.06	96.94±.13*	97.05±.15	76.30±.16	80.95±.22	82.90±.29*	80.00±.29
EA	95.76±.09	97.09±.09	97.47±.10	97.46±.11	76.40±.18	81.56±.21	83.09±.22	81.60±.13
TA	95.00±.10	97.13±.08	97.18±.11	97.30±.10	76.57±.14	81.34±.18	82.75±.26	82.14±.16
AA	95.01±.11	96.59±.04*	96.99±.06	97.30±.11	76.36±.22	81.34±.29	82.21±.17	81.19±.19
FAA	93.80±.12	96.69±.16	97.30±.24	96.42±.12	76.04±.28	79.08±.12	79.95±.12	81.39±.16
HaS	94.97±.27	95.60±.15	96.94±.08	96.89±.10*	75.82±.32	78.76±.24	80.22±.16	76.89±.33
DADA	93.96±.38	95.61±.14	97.30±.13*	97.30±.14*	74.37±.47	80.25±.28	82.50±.26*	80.98±.15
Cutout	94.78±.35	95.81±.17	96.92±.09	96.96±.09*	74.84±.56	78.62±.25	79.84±.14	77.37±.28
CutMix	95.28±.16	96.81±.10*	96.93±.10*	96.47±.07	76.09±.15	81.24±.14	82.67±.22	79.57±.10
GridMask	95.02±.26	96.15±.19	96.92±.09	96.91±.12	76.07±.18	78.38±.22	80.40±.20	77.28±.38
AdvMask	95.49±.17*	96.69±.10*	97.02±.05*	97.03±.12*	76.44±.18*	78.99±.31*	80.70±.25*	79.96±.27*
TeachA	95.05±.21	96.40±.14	97.50±.16	97.29±.11	76.18±.31	80.54±.25	82.81±.26	81.30±.18
MADAug	95.25±.18	97.12±.17	97.48±.15	97.37±.11	76.49±.21	81.40±.18	83.01±.23	81.67±.19
SoftAug	94.51±.20	96.99±.14	97.15±.16	97.22±.19	76.41±.33	80.94±.33	82.61±.24	80.33±.20
Ours	**95.87**±.21	**97.21**±.10	**97.66**±.06	**97.51**±.07	**80.81**±.41	**81.75**±.28	**83.17**±.19	**82.73**±.15

of $\mathcal{V}(x_i)$ into the range $[0, 1]$, consistent with the allowable augmentation strength range m_{max}, we apply a min-max normalization on it and derive the applied augmentation strengths as $s(x_i) \cdot m_{max}$. When $s(x_i) \to 1$, the augmented samples present a greater variability, and conversely, minor transformations occur as $s(x_i) \to 0$. Importantly, $s(x_i)$ evolves dynamically throughout training, reflecting the model's changing perception of each sample's role in the optimization process. Due to the limited space, we provide the details of the augmentation space and algorithm in Appendix A.

Theoretical Analysis. We provide a theoretical analysis to better understand why SADA works. In particular, we show that SADA reduces the empirical Rademacher complexity, thereby tightening the generalization error bound. Formally, the generalization gap is upper-bounded by a term of the form $\mathcal{O}(\frac{1}{n}\sqrt{\sum_i \alpha_i s_i^2})$, where α_i measures sample sensitivity to augmentation and s_i denotes the applied augmentation strength. Optimizing this bound yields a simple allocation rule: augment stable samples more, and unstable samples less. This aligns precisely with the SADA strategy. Therefore, SADA improves generalization from data-centric learning. The complete theoretical derivation is provided in Appendix B.

Complexity Analysis. We provide a theoretical analysis showing that SADA introduces negligible computational overhead compared to vanilla training. Specifically, the computational complexity of SADA is $\mathcal{O}(K \times N \times L)$, where K is the total number of classes, N is the number of samples, and L denotes the window length.

4 EXPERIMENT

Datasets and network architectures. Following prior works (Müller & Hutter, 2021; Yang et al., 2024b; Cubuk et al., 2019), we evaluate our work on a diverse set of benchmark datasets, including CIFAR-10/100 (Krizhevsky et al., 2009), Tiny-ImageNet (Chrabaszcz et al., 2017), and ImageNet-1k (Krizhevsky et al., 2017). To assess its effectiveness in fine-grained recognition tasks, we additionally conduct experiments on Oxford Flowers (Nilsback & Zisserman, 2008), Oxford-IIIT Pets (Parkhi et al., 2012), FGVC-Aircraft (Maji et al., 2013), and Stanford Cars (Krause et al., 2013). Moreover, for evaluating performance under class imbalance, we also conduct experiments using long-tailed datasets, such as ImageNet-LT and Places-LT (Liu et al., 2019). Due to the limited space, more experimental settings are provided in Appendix C.

Comparison with State-of-the-arts. We compare our method with a wide range of representative and commonly used methods, including: 1). Cutout (DeVries & Taylor, 2017), 2). HaS (Singh & Lee, 2017), 3). CutMix (Yun et al., 2019), 4) GridMask (Chen et al., 2020), 5). AdvMask (Yang et al., 2023), 6). RandomErasing (Zhong et al., 2020), 7). AutoAugment (AA) (Cubuk et al., 2019), 8). Fast-AutoAugment (FAA) (Lim et al., 2019), 9). RandAugment (RA) (Cubuk et al., 2020), 10). DADA (Li et al., 2020), 11). TeachAugment (TeachA) (Suzuki, 2022), 12). MADAug (Hou et al., 2023), 13). SoftAug (Liu et al., 2023), and 14). TrivialAugment (TA) (Müller & Hutter, 2021).

Table 2: Image classification accuracy (%) on Tiny-ImageNet across various deep models.

Method	ResNet-18	ResNet-50	WRN-50-2	ResNext-50
baseline	61.38±0.99	73.61±0.43	81.55±1.24	79.76±1.89
HaS	63.51±0.58	75.32±0.59	81.77±1.16	80.52±1.88
FAA	68.15±0.70	75.11±2.70	82.90±0.92	81.04±1.92
DADA	70.03±0.10	78.61±0.34	83.03±0.18	81.15±0.34
Cutout	68.67±1.06	77.45±0.42	82.27±1.55	81.16±0.78
CutMix	64.09±0.30	76.41±0.27	82.32±0.46	81.31±1.00
AdvMask	65.29±0.20	78.84±0.28	82.87±0.55	81.38±1.54
GridMask	62.72±0.91	77.88±2.50	82.25±1.47	81.05±1.33
AutoAugment	67.28±1.40	75.29±2.40	79.99±2.20	81.28±0.33
RandAugment	65.67±1.10	75.87±1.76	82.25±1.02	80.36±0.62
EntAugment	70.16±1.01	79.06±1.20	83.92±0.97	81.90±1.51
TeachAugment	70.05±0.57	70.56±0.44	82.95±0.13	81.39±0.97
TrivialAugment	69.97±0.96	78.41±0.39	82.16±0.32	80.91±2.26
RandomErasing	64.00±0.37	75.33±1.58	81.89±1.40	81.52±1.68
Ours	**71.15**±0.60	**79.66**±0.52	**84.15**±0.35	**82.16** ±0.20

Table 3: Top-1 accuracy (%) on ImageNet-1k dataset with ResNet-50.

HaS	GM	Cutout	CutMix	Mixup	AA	EA	FAA	RA	MA	SA	DADA	TA	TeachA	Ours
77.2±0.2	77.9±0.2	77.1±0.3	77.2±0.2	77.0±0.2	77.6±0.2	78.2±0.2	77.6±0.2	77.6±0.2	**78.5**±0.1	78.0±0.1	77.5±0.1	77.9±0.3	77.8±0.2	78.4±0.1

4.1 Performance Comparison

Table 1 compares our method and several widely adopted state-of-the-art baselines on the CIFAR-10 and CIFAR-100 datasets across various deep architectures. While the accuracy margins on these small-scale benchmarks are generally narrow, our method consistently achieves the highest performance across architectures. For example, using WideResNet-28-10 on CIFAR-10, our approach improves accuracy by 2.14% over the best-performing baseline. Similarly, with ResNet-44 on CIFAR-100, we observe a notable performance gain of 7.01%.

To assess scalability, we further evaluate our method on the large-scale Tiny-ImageNet dataset in Table 2. Across different architectures, our method consistently outperforms existing baselines. For instance, on ResNeXt-50, it surpasses the next-best method by over 0.64%, without introducing noticeable training overhead compared to standard training routines. These gains can be attributed to our method's adaptive augmentation mechanism, which dynamically adjusts the augmentation strength based on each sample's influence stability. This design enables a better balance between evolving models and training data, thereby enhancing generalization across models and datasets.

4.2 Generalization on Large-scale ImageNet-1k

We further evaluate the generalization performance of our method on the large-scale ImageNet-1k dataset. Specifically, following experiment settings (Müller & Hutter, 2021), we train ResNet-50 models using different DA methods. As shown in Table 3, our method achieves a competitive performance compared to other baselines. While the accuracy gap between our method and MADAug is marginal, our approach is significantly more efficient, achieving over 2x faster training than MADAug and over 4x faster than TeachAugment, without relying on auxiliary models or bi-level optimization. These results demonstrate that our method offers a compelling trade-off between accuracy and efficiency for large-scale model training.

4.3 Data Augmentation Improves Transfer Learning

Beyond evaluations on benchmark datasets, we assess model generalization through transfer learning, which tests a model's ability to extract transferable and robust features across domains (Yosinski et al., 2014; Kornblith et al., 2019; Raghu et al., 2019). In this setup, we pretrain ResNet-50 models on CIFAR-100 and Tiny-ImageNet using various data augmentation methods, and then fine-tune them on CIFAR-10.

Table 4: Transferred test accuracy (%) on CIFAR-10 of various DA methods. The pretrained ResNet-50 model is trained on CIFAR-100 (upper row) and Tiny-ImageNet (bottom row).

baseline	HaS	FAA	DADA	Cutout	CutMix	MADAug	GridMask	AA	EA	RA	TeachAug	TA	RE	Ours
91.53±.03	92.51±.24	92.28±.13	92.58±.09	92.42±.20	92.81±.47	92.84±.10	91.49±.10	92.82±.04	92.89±.19	92.78±.23	92.83±.18	92.80±.16	92.55±.05	**93.11**±.25
64.02±.05	66.84±.06	70.32±.63	69.04±.43	65.54±.75	69.29±.09	72.82±.32	64.88±.43	69.53±.53	72.68±.73	64.68±.23	69.98±.17	71.53±.35	64.56±.27	**77.26**±.12

Table 5: Top-1 classification accuracy (%) on ImageNet-LT and Places-LT. * means results reported in the original paper.

Dataset	Methods	closed-set setting				open-set setting			
		Many-shot	**Medium-shot**	**Few-shot**	**Overall**	**Many-shot**	**Medium-shot**	**Few-shot**	**F-measure**
ImageNet-LT	OLTR	43.2±0.1*	35.1±0.2*	18.5±0.1*	35.6±0.1*	41.9±0.1*	33.9±0.1*	17.4±0.2*	44.6±0.2*
	OLTR+**Ours**	**46.9**±0.1	**37.0**±0.2	**21.6**±0.2	**36.9**±0.1	**45.2**±0.1	**35.6**±0.2	**20.6**±0.1	**45.5**±0.1
Places-LT	OLTR	**44.7**±0.1*	37.0±0.2*	25.3±0.1*	35.9±0.1*	**44.6**±0.1*	36.8±0.1*	25.2±0.2*	46.4±0.1*
	OLTR+**Ours**	44.3±0.1	**40.8**±0.2	**28.9**±0.2	**38.5**±0.1	44.1±0.1	**40.6**±0.2	**28.6**±0.1	**50.4**±0.2

This evaluation is motivated by the fact that stronger data augmentation strategies can lead to more generalizable feature representations. As shown in Table 4, it can be observed that our method achieves consistently higher accuracy after transfer compared to baseline augmentation approaches, regardless of the pertaining dataset. These results indicate that models trained with our dynamic augmentation strategy learn more transferable and semantically meaningful features, further validating the generalization benefits of our approach.

4.4 Results on Fine-Grained Datasets

To further assess the versatility of our method, we evaluate its performance on several fine-grained classification benchmarks, including Oxford Flowers (Nilsback & Zisserman, 2008), Oxford-IIIT Pets (Parkhi et al., 2012), FGVC-Aircraft (Maji et al., 2013), and Stanford Cars (Krause et al., 2013). These datasets are characterized by subtle inter-class differences, making them particularly challenging for standard data augmentation strategies.

As shown in Table 6, incorporating our method into the standard training process can significantly enhance model performance. Notably, on the Oxford Flower dataset, it achieves over 8% absolute improvement compared to baseline learning. These results highlight the effectiveness of our sample-aware augmentation approach in fine-grained scenarios.

4.5 Results on Long-Tailed Datasets

While most existing DA methods are not evaluated on long-tailed datasets, we further evaluate the robustness of our method on more challenging long-tailed benchmarks, i.e., ImageNet-LT and Places-LT (Liu et al., 2019), which exhibit significant class imbalance. We closely follow the experimental setting in OLTR (Liu et al., 2019), using the same network backbone and evaluation metrics, except utilizing our augmentation method. As shown in Table 5, our method achieves consistent performance improvements across both closed-set and open-set evaluation settings. On ImageNet-LT, we improve the overall top-1 accuracy by 1.3% in the closed-set scenario. On Places-LT, our method increases the F-measure by 4% in the open-set setting. These results highlight the ability of our adaptive augmentation strategy to improve generalization under severe data imbalance, without requiring explicit rebalancing techniques or auxiliary supervision.

4.6 Cross-Architecture Generalization

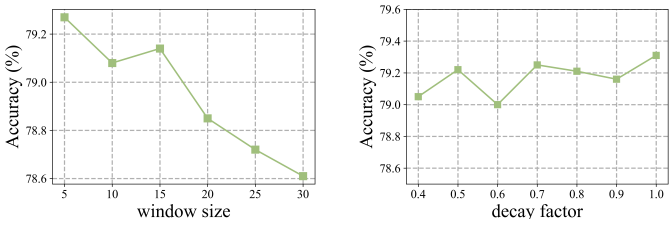
In Table 1 and Table 2, we demonstrate the effectiveness of our method across various CNN-based architectures. To further evaluate its generalizability, we extend our experiments to Vision Transformer-based models using the ImageNet-1k dataset. As shown in Table 7, our method yields consistent performance gains for both ViT variants, improving the performance of ViT-Base/Large/Huge on ImageNet-1k. Importantly, these gains are achieved without introducing large additional training overheads, highlighting the efficiency of our method. Consequently, these results confirm that our method is architecture-agnostic and can be seamlessly integrated into training pipelines as a plug-and-play module to improve performance.

Table 6: Test accuracy (%) on fine-grained datasets with ResNet-50.

Dataset	baseline	Ours
Oxford Flowers	89.47±0.08	**98.04**±0.09
Oxford-IIIT Pets	89.73±0.18	**92.53**±0.12
FGVC-Aircraft	77.25±0.09	**80.76**±0.12
Stanford Cars	82.13±0.03	**91.89**±0.07

Table 7: Test accuracy (%) on ImageNet-1k with ViT-Base/Large/Huge.

Model	baseline	Ours
ViT-B	82.30	**83.38**↑1.08
ViT-L	84.47	**85.01**↑0.54
ViT-H	85.91	**86.88**↑0.97



(a) (b)

Figure 3: The stability of our method on the two parameters, i.e., the window size and the decay factor, with CIFAR-100 using ResNet-18.

4.7 Efficiency Comparison

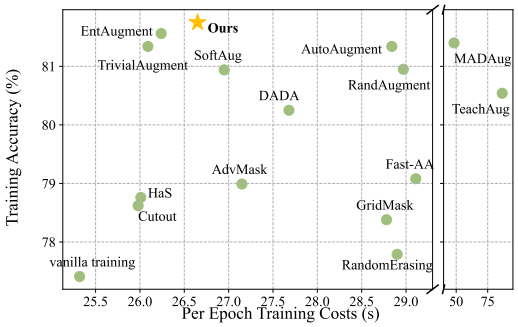


Figure 4: Comparison in the effectiveness-efficiency tradeoff. We report the average per-epoch training costs using a 2-NVIDIA-RTX2080TI-GPUs server.

We compare the training costs of our method with other baselines. As illustrated in Figure 4, in the efficiency-effectiveness plane, our method achieves a favorable trade-off between training cost and performance. Consistent with the complexity analyses in Section 3, our approach introduces negligible additional overhead compared to standard training. This is primarily because the required gradient information can be directly obtained during standard forward and backward passes, without relying on auxiliary networks or a complex optimization process. While our method incurs slightly higher training costs than baselines such as Cutout, HaS, and TrivialAugment, the difference is minimal. Importantly, our method consistently delivers better performance, achieving a better balance between efficiency and accuracy.

4.8 Ablation Study

We conduct an ablation study to investigate the effect of two hyperparameters in our method: the window size L in Eq. equation 6 and decay factor in Eq. equation 7. As shown in Figure 3(a), increasing the window size L leads to a consistent drop in classification accuracy. This is because larger windows oversmooth the instantaneous dynamics of sample influence, thereby delaying the dynamic augmentation's responsiveness to model training dynamics. As a result, maintaining a small window size not only better captures the local importance of each sample but also reduces the memory costs. Figure 3(b) shows the effect of varying the decay factor β. The model performance remains generally stable across different β values, indicating that our method is robust to it.

5 Conclusion

This paper proposes a novel on-the-fly data augmentation method that performs sample-aware augmentation by modeling the evolving interplay between data and the model during training. Unlike existing approaches, our proposed method leverages a dynamic augmentation mechanism, mitigating overfitting for stable samples by increasing their diversity while promoting generalization for uncertain ones by preserving semantic fidelity. We hope our work inspires further research on train-dynamic-aware data augmentation from an on-the-fly perspective and believe our method will serve as a promising plug-and-play tool for the community, enabling enhanced deep model training.

REFERENCES

Tom Bekor, Niv Nayman, and Lihi Zelnik-Manor. Freeaugment: Data augmentation search across all degrees of freedom. In *European Conference on Computer Vision*, pp. 36–53. Springer, 2024.

Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Gridmask data augmentation. *arXiv preprint arXiv:2001.04086*, Jan. 2020.

Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, Aug 2017.

Ekin D. Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 113–123, 2019.

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Proc. Adv. Neural Inf. Process. Syst.*, volume 33, pp. 18613–18624, 2020.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, Nov 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Xavier Gastaldi. Shake-shake regularization. *arXiv preprint arXiv:1705.07485*, May 2017.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Ryuichiro Hataya, Jan Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. Meta approach to data augmentation optimization, 2020. URL https://arxiv.org/abs/2006.07965.

Ryuichiro Hataya, Jan Zdenek, Kazuki Yoshizoe, and Hideki Nakayama. Meta approach to data augmentation optimization. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2574–2583, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 770–778, 2016.

Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In *International conference on machine learning*, pp. 2731–2741. PMLR, 2019.

Chengkai Hou, Jieyu Zhang, and Tianyi Zhou. When to learn what: Model-adaptive data augmentation curriculum. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1717–1728, 2023.

Guillermo Iglesias, Edgar Talavera, Ángel González-Prieto, Alberto Mozo, and Sandra Gómez-Canaval. Data augmentation techniques in time series domain: a survey and taxonomy. *Neural Computing and Applications*, 35(14):10123–10145, 2023.

Jang-Hyun Kim, Wonho Choo, and Hyun Oh Song. Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *International conference on machine learning*, pp. 5275–5285. PMLR, 2020.

Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2661–2671, 2019.

Jonathan Krause, Jia Deng, Michael Stark, and Li Fei-Fei. Collecting a large-scale dataset of fine-grained cars. 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Yonggang Li, Guosheng Hu, Yongtao Wang, Timothy M. Hospedales, Neil Martin Robertson, and Yongxin Yang. DADA: differentiable automatic data augmentation. In *European Conference on Computer Vision*, pp. 580–595. Springer, 2020.

Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/6add07cf50424b14fdf649da87843d01-Paper.pdf.

Shiqi Lin, Zhizheng Zhang, Xin Li, and Zhibo Chen. Selectaugment: hierarchical deterministic sample selection for data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 1604–1612, 2023.

Aoming Liu, Zehao Huang, Zhiwu Huang, and Naiyan Wang. Direct differentiable augmentation search, 2021. URL https://arxiv.org/abs/2104.04282.

Yang Liu, Shen Yan, Laura Leal-Taixé, James Hays, and Deva Ramanan. Soft augmentation for image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16241–16250, 2023.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2537–2546, 2019.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Juliette Marrie, Michael Arbel, Diane Larlus, and Julien Mairal. Slack: Stable learning of augmentations with cold-start and kl regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24306–24314, 2023.

Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 774–782, 2021.

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. IEEE, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging, 2019. URL https://arxiv.org/abs/1902.07208.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 3544–3553. IEEE, Oct 2017.

Teppei Suzuki. Teachaugment: Data augmentation optimization using teacher knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10904–10914, 2022.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics, 2020. URL https://arxiv.org/abs/2009.10795.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning, 2019. URL https://arxiv.org/abs/1812.05159.

Alexander Tsaregorodtsev and Vasileios Belagiannis. Particleaugment: Sampling-based data augmentation. *Computer Vision and Image Understanding*, 228:103633, 2023.

Devesh Walawalkar, Zhiqiang Shen, Zechun Liu, and Marios Savvides. Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification. *arXiv preprint arXiv:2003.13048*, 2020.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks, 2017. URL https://arxiv.org/abs/1611.05431.

Suorong Yang, Weikang Xiao, Mengcheng Zhang, Suhan Guo, Jian Zhao, and Furao Shen. Image data augmentation for deep learning: A survey. *arXiv preprint arXiv:2204.08610*, 2022.

Suorong Yang, Jinqiao Li, Tianyue Zhang, Jian Zhao, and Furao Shen. Advmask: A sparse adversarial attack-based data augmentation method for image classification. *Pattern Recognition*, 144: 109847, 2023.

Suorong Yang, Suhan Guo, Jian Zhao, and Furao Shen. Investigating the effectiveness of data augmentation from similarity and diversity: An empirical study. *Pattern Recognition*, 148:110204, 2024a. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2023.110204.

Suorong Yang, Furao Shen, and Jian Zhao. Entaugment: Entropy-driven adaptive data augmentation framework for image classification. In *European Conference on Computer Vision*, pp. 197–214. Springer, 2024b.

Suorong Yang, Peijia Li, Xin Xiong, Furao Shen, and Jian Zhao. Adaaugment: A tuning-free and adaptive approach to enhance data augmentation. *IEEE Transactions on Image Processing*, 2025.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization, 2017. URL https://arxiv.org/abs/1611.03530.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1Ddp1-Rb.

Xin Zhang, Jiawei Du, Yunsong Li, Weiying Xie, and Joey Tianyi Zhou. Spanning training progress: Temporal dual-depth scoring (tdds) for enhanced dataset pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26223–26232, 2024.

Xinyu Zhang, Qiang Wang, Jian Zhang, and Zhao Zhong. Adversarial autoaugment. *arXiv preprint arXiv:1912.11188*, 2019.

Yu Zheng, Zhi Zhang, Shen Yan, and Mi Zhang. Deep autoaugmentation. In *ICLR*, 2022.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proc. AAAI*, volume 34, pp. 13001–13008, 2020.

A More Details of the Method

Table 8: Our employed augmentation operations with corresponding magnitude ranges across different datasets (Müller & Hutter, 2021; Yang et al., 2024b), only including lightweight image transformations.

Transformation	Max allowable magnitude
identity	-
auto contrast	-
equalize	-
color	$+1.9$
contrast	$+1.9$
brightness	$+1.9$
sharpness	$+1.9$
rotation	$\pm 30°$
translate$_x$	± 10
translate$_y$	± 10
shear$_x$	± 0.3
shear$_y$	± 0.3
solarize	$+256$
posterize	$+4$

Algorithm: Detailed algorithm pipeline of our method

Require: Training dataset $\mathcal{D}$, network f_θ with weights θ, decay coefficient β

for each training step $t = 0, 1, \ldots$ **do**
  Sample a mini-batch $\{\boldsymbol{x}_i, y_i\}_{i=1}^{B}$ from $\mathcal{D}$
  Compute predicted probabilities $f_\theta(x_i)$ for each $\boldsymbol{x}_i$
  Update model weights according to Eq. 1
  Compute $\Delta\ell_t^n$ for each $\boldsymbol{x}_i$ (Eq. 5)
  Compute $\mathcal{V}_t(x_i)$ in one window (Eq. 6)
  Update $\mathcal{V}_t(x_i)$ with EMA (Eq. 7)
  Compute $s(x_i)$
  Augment samples with $s(x_i)$ in next epoch
end for

B Theoretical Justification

We provide a sketch of theoretical justification showing why our sample-adaptive augmentation (SADA) strategy—assigning stronger augmentation to stable (low-variance) samples and weaker augmentation to unstable (high-variance) samples—can improve generalization.

Setup. Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ be the training set, with feature map $\phi : \mathcal{X} \to \mathbb{R}^d$ satisfying $\|\phi(x)\| \le R$. The hypothesis class is $f_\theta(x) = \langle \theta, \phi(x) \rangle$ with $\|\theta\| \le B$. For each sample, an augmentation operator $\mathcal{A}_{s_i}$ with magnitude $s_i \in [0, s_{\max}]$ generates

$$\phi(\tilde{x}_i) = \mu_i(s_i) + \Delta_i(s_i), \quad \mathbb{E}[\Delta_i(s_i) \mid x_i] = 0.$$

We assume (i) the loss ℓ is L_ℓ-Lipschitz in its prediction, and (ii) the fluctuation satisfies $\mathbb{E}\|\Delta_i(s_i)\|^2 \le \alpha_i s_i^2$, where α_i encodes the sample's sensitivity to augmentation.

Rademacher complexity with augmentation. The empirical Rademacher complexity of the augmented class is

$$\mathfrak{R}_n(\mathcal{F}_s) = \frac{1}{n}\,\mathbb{E}_{\varepsilon,\tilde{x}}\Big[\sup_{\|\theta\|\le B} \sum_{i=1}^{n} \varepsilon_i \langle \theta, \phi(\tilde{x}_i)\rangle\Big].$$

Decomposing into mean and fluctuation terms and applying Khintchine–Kahane inequality yields

$$\mathfrak{R}_n(\mathcal{F}_s) \;\le\; \frac{B}{n}\Big(\sqrt{\sum_{i=1}^{n} \|\mu_i(s_i)\|^2} + \sqrt{\sum_{i=1}^{n} \alpha_i s_i^2}\Big).$$

By the contraction lemma, the loss class satisfies

$$\mathfrak{R}_n(\ell \circ \mathcal{F}_s) \;\le\; \frac{L_\ell B}{n}\Big(\sqrt{\sum_{i=1}^{n} \|\mu_i(s_i)\|^2} + \sqrt{\sum_{i=1}^{n} \alpha_i s_i^2}\Big).$$

Thus, the generalization gap is controlled by a complexity term $\frac{L_\ell B}{n}\sqrt{\sum_i \alpha_i s_i^2}$.

Optimal allocation. Suppose we require $\sum_i s_i \geq S$. Minimizing $\sum_i \alpha_i s_i^2$ under this constraint gives the strictly convex problem

$$\min_{0 \leq s_i \leq s_{\max}} \sum_{i=1}^{n} \alpha_i s_i^2 \quad \text{s.t.} \quad \sum_{i=1}^{n} s_i \geq S.$$

The KKT conditions yield a water-filling solution:

$$s_i^* = \min\left\{s_{\max}, \frac{\lambda}{2\alpha_i}\right\}, \quad \sum_i s_i^* = S.$$

Therefore, the optimal strategy assigns *larger augmentation strength to samples with smaller* α_i (i.e., lower sensitivity), and smaller strength to those with larger α_i (higher sensitivity).

Connection to variance measure. In our method, α_i is bounded by a constant multiple of the variance measure $\mathcal{V}(x_i)$ computed from the gradient dynamics, i.e., $\alpha_i \leq c\,\mathcal{V}(x_i)$. Hence, the optimal allocation s_i^* is monotone decreasing in $\mathcal{V}(x_i)$, which aligns exactly with our SADA rule: *low-variance samples receive stronger augmentation, while high-variance samples receive weaker augmentation.*

C Implementation Details

Our experiments are conducted across a wide range of network architectures, including ResNet-based models, e.g., ResNet-18/50 and Wide ResNet, ViT-based models, e.g., ViT-Base/Large/Huge, and architectures with advanced regularization such as Shake-Shake (Gastaldi, 2017) and ResNeXt (Xie et al., 2017). This setup allows us to comprehensively evaluate the generalization and scalability of our method across different data domains and architectural families. Some results for baseline methods are taken from the original publications Yang et al. (2024b); Müller & Hutter (2021); Cubuk et al. (2019).

Our experimental setup follows standard practices established in prior works (DeVries & Taylor, 2017; Yang et al., 2023; 2024b; Chen et al., 2020; Müller & Hutter, 2021). Specifically, during online training, only augmented data is used for model optimization, without incorporating original data. Unless otherwise specified, we train all models for 300 epochs using a batch size of 256, an initial learning rate of 0.1, SGD with momentum 0.9, weight decay of $5e-4$, and a cosine annealing learning rate decay strategy. Input images undergo standard preprocessing with random cropping and horizontal flipping, consistent with the augmentation setup used for the baseline methods. For experiments involving the Shake-Shake model, we follow the established protocol (Gastaldi, 2017) and train for 1800 epochs using SGD with Nesterov Momentum, weight decay of $1e-3$, and cosine learning rate decay. The augmentation operation space used is consistent with prior works (Müller & Hutter, 2021; Yang et al., 2024b). Unless otherwise stated, we use ResNet-50 as the default architecture. We consistently set the window size as 10 and the decay factor as 0.9 across all the tasks and datasets without any dataset- or architecture-specific tuning. The consistent improvements across settings demonstrate that SADA is robust and not sensitive to these hyperparameters in practice. For all experiments, we report the average and standard deviation of test accuracy over three independent runs. Note that because of the huge calculation consumption on ImageNet-1k, the experiment in each case is performed once.

D Performance under Controlled Randomness of the Augmentation Operations

Table 9: Performance under different numbers of augmentation operations in our augmentation space on CIFAR-100 using ResNet-50.

# of operations	4	6	8	10	12	14
Acc. (%)	81.6	81.5	81.6	81.8	81.9	81.8

In this section, we evaluate the performance of our method under different controlled randomness. As shown in Table 9, it can be observed that reduced randomness in augmentation operations brings

minimal influence on our method. SADA remains highly stable across different levels of operation randomness. Therefore, we validate that the superior effectiveness of SADA stems from the adaptive adjustment of augmentation strengths, rather than from the random selection of operations.

E Training Costs Analysis

Table 10: Wall-clock time (h) of baseline vs. SADA on ImageNet-1k using a 4-A100-GPU server.

	ResNet-50	ViT-B	ViT-L
Baseline	22.1	149.1	363.2
SADA	22.5	150.8	366.4
Increased costs	+1.8%	+1.1%	0.8%

In this section, we further analyze SADA's actual training costs. As shown in Table 10, it can be observed that SADA incurs no noticeable additional training cost compared to standard training. This is because we adopt a first-order Taylor expansion to convert the gradient-projection term into a loss-difference formulation (Eq. 6), which can be obtained directly from the forward pass. This avoids any additional gradient calculation beyond standard training, and thus the resulting computational overhead introduced by SADA is minimal.

F Comparison with EntAugment

Recently, adaptive data augmentation methods have shown strong effectiveness, and both SADA and EntAugment fall within this broader family of approaches that adjust augmentation strength based on per-sample behavior during training. While we empirically compare SADA and EntAugment across various evaluation settings, here we outline their methodological differences to provide a clearer understanding. 1). Different signals. EntAugment uses classification entropy from model snapshots, while SADA instead uses gradient-based influence projection to measure how each sample directly contributes to the optimization trajectory. 2). Different stability mechanisms. EntAugment can fluctuate across training, while SADA incorporates the temporal variance of sample influence over a local window, providing a more stable and reliable indicator of the learning effect. 3). Different training-stage awareness: EntAugment's entropy does not explicitly capture how a sample's effect evolves over time. SADA naturally reflects evolving sample dynamics via gradient influence and its temporal consistency. 4). Different optimization basis. EntAugment relies on a heuristic uncertainty signal. SADA is grounded in optimization theory, using gradients and accumulated updates to modulate augmentation in a way that is directly aligned with the learning process.

In summary, while the two methods share similarities, SADA adopts a fundamentally different mechanism that is more stable, more training-aware, and more closely aligned with underlying optimization dynamics. Thus, while EntAugment provides promising performance, SADA achieves stronger effectiveness.

G More Comparison with the Published Results of TrivialAugment

Table 11: Comparison with the published results of TrivialAugment (TA) using the experimental setting from Müller & Hutter (2021) on CIFAR-10/100.

Dataset	Method	Baseline	TA	Ours
CIFAR-10	WRN-28-10	97.0	97.5	**97.9**
	SS-26-96	97.5	98.2	**98.4**
CIFAR-100	WRN-28-10	82.2	84.3	**84.6**
	SS-26-96	83.3	86.2	**86.7**

In addition to the comparisons with TrivialAugment (TA) in Section 4, in this section, we compare with TA's published results using its training configurations. As shown in Table 11, under the identical settings, SADA consistently surpasses TA across deep models and datasets.

H Societal Impact Statement

This work focuses on improving the generalization and training efficiency of deep learning models through a sample-aware data augmentation framework, SADA. The potential positive societal impacts include reducing the reliance on large-scale, manually curated datasets by enabling more effective use of limited or imbalanced data, which can lower data collection costs and broaden access to machine learning in resource-constrained settings. In particular, the method's plug-and-play nature and computational efficiency may benefit applications in healthcare, environmental monitoring, or education, where robust generalization under limited data is critical.

I Discussion and Future Work

In this section, we discuss some potential limitations and future work for our method.

Since our method computes the variance of gradient-based influence signals to determine sample-wise augmentation strengths, it requires maintaining a local history of these values within a sliding window and introduces two parameters: window size L and decay factor β. In all our experiments across datasets and architectures, we adopt the same default hyperparameter configuration ($L = 10$ and $\beta = 0.9$) without any dataset-specific or model-specific tuning. To ensure the responsiveness of augmentation strength to recent training dynamics, our framework favors small window sizes, thus capturing meaningful local variations. Meanwhile, our ablation studies confirm that the decay factor is highly stable. These findings suggest that our framework is robust to hyperparameter choices. To provide clearer parameter setting suggestions in practice, based on our ablation study results, we summarize these insights: using $L = 5, 10$ with $\beta = 0.9$, without large-scale tuning.

Currently, our method is designed and evaluated primarily for supervised image classification tasks. While the sample-aware augmentation principle is general, its application to other domains, such as object detection, semantic segmentation, or image generation, remains underexplored. These tasks involve fundamentally different training objectives and model behaviors, and investigating how gradient-guided influence estimation interacts with task-specific objectives and model architectures will be an important direction for future work.

J AI Assistant usage statement

During the preparation of this paper, we made only moderate use of large language models for text polishing.

K Reproducibility

Implementation will be made publicly available.