Dataset Distillation via Dynamic Difficulty Alignment

Anonymous Authors[1]

Abstract

Dataset distillation has emerged as a prominent area of research, aiming to compress large, original datasets into compact, lightweight synthetic datasets. Although existing works have shown their effectiveness on small scale datasets, they struggle to maintain performance on large scale datasets. Our analysis reveals that conventional dataset distillation methods have a bias towards generating synthetic images that mainly capture the easy samples of the real dataset. Although this bias is actually beneficial for small scale datasets, larger datasets require synthetic images to be diverse, capturing not only the easy samples, but also the hard samples of the real dataset. To this end, we propose a novel dataset distillation framework that generates synthetic images in multiple small segments, where each segment is distilled by matching a small portion of the whole dataset. Each portion is obtained by filtering out images prior to distillation, and each synthetic segment is distilled using a different set of images . This method facilitates the generation of diverse synthetic datasets that captures both easy and hard samples of the real dataset. Our experiments show that our method incorporated into existing frameworks show SOTA performance across a variety of large scale datasets, paving the way to make dataset distillation more scalable to larger datasets.

1. Introduction

The increasing scale of datasets poses significant challenges for training machine learning models, including high storage requirements, computational demands, and extensive training times. Dataset distillation addresses these challenges by synthesizing a smaller synthetic dataset from the original dataset, such that models trained on the synthetic dataset achieves performance comparable to those trained on the original dataset. This technique has practical applications in various domains. For example, in edge computing, where devices such as smartphones and IoT sensors have limited storage and computational power, dataset distillation enables efficient on-device training. Similarly, in federated learning, it reduces the amount of data transmitted between devices and servers, thereby improving communication efficiency.

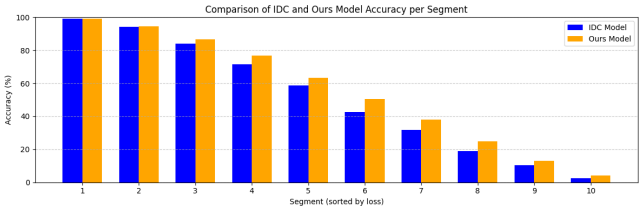

Figure 1. Classification accuracy of a model trained with synthetic datasets generated using IDC and our method. The original dataset is divided into 10 subsets, then the subsets are ordered by loss. Subsets on the left represent easy samples, while subsets on the right represent hard samples. Performance quickly saturates for hard samples with IDC, while our method maintains high accuracy even for hard samples.

Various approaches to dataset distillation have been proposed. For instance, gradient matching methods optimize synthetic datasets to replicate the gradients of the original dataset. Trajectory matching extends this concept by aligning the full training trajectories of the original data. Other methods, such as distribution matching, aim to generate synthetic images that produce feature maps similar to those of real images. These methods have shown effectiveness in generating synthetic datasets with a small images-per-class (IPC). However, their performance degrades when generating synthetic datasets with high IPC, even falling behind performance of random selection. The primary reason for this limitation is that conventional methods are biased toward generating synthetic datasets that primarily capture easy samples of the original dataset. As illustrated in Fig. 1, both IDC and MTT, the standard gradient matching and trajectory matching frameworks, achieve high classification accuracy for easy samples but exhibit significant perfor-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

mance degradation for harder samples. Recent methods, such as DATM and SelMatch, have attempted to address the easy-sample bias by embedding hard features into synthetic datasets. DATM achieves this by matching specific parts of training trajectories, whereas SelMatch relies on heuristic scores to select samples of appropriate difficulty for distillation. However, these methods rely heavily on heuristics, requiring manual inspection and extensive trial-and-error processes. This poses a challenge as datasets vary in their characteristics, necessitating separate adjustments for each dataset. Moreover, even within the same dataset, additional tuning is often required for different IPC settings.

To address these challenges, we propose D4A (Dataset Distillation via Dynamic Difficulty Alignment), a novel framework that dynamically incorporates hard samples into synthetic datasets without relying on heuristic intervention. Instead of manual inspection, D4A employs a model-driven approach to identify parts of the original dataset that should be embedded into the synthetic dataset. The process begins by generating a small subset of synthetic images. A model trained on these synthetic images is then used to classify the original dataset. Images that are correctly classified are identified as easy samples and discarded, while misclassified images are retained as hard samples for embedding into the synthetic dataset. This iterative filtering process ensures that subsequent synthetic images emphasize harder samples. However, naively filtering out easy samples can lead to performance degradation due to the forgetting phenomenon (), where adding new synthetic images to the existing synthetic dataset causes the trained model to misclassify images it previously classified correctly. To mitigate forgetting, we design a regularization loss derived from previously generated synthetic images. This loss is applied during the training of subsequent synthetic images, reducing forgetting and improving overall performance. Our main contributions can be summarized as follows:

- We propose D4A, a novel dataset distillation framework that dynamically incorporates hard samples into synthetic datasets without heuristic intervention.
- We introduce a regularization loss to mitigate the forgetting phenomenon, reducing misclassifications caused by the addition of new synthetic segments.
- Our method achieves state-of-the-art performance across diverse datasets. For example, in the Tiny-ImageNet 50 IPC setting, D4A achieves 44.3% accuracy, outperforming the previous best method by a margin of 4.6%.

2. Related Works

Dataset distillation compresses real images into synthetic images by optimizing the distance between the features of real and synthetic images. Current research primarily focuses on two strategies: gradient/trajectory matching and distribution matching.

Gradient/Trajectory Matching: The core principle of these methods is that synthetic images should produce back-propagation signals similar to those of real images when training models. DC () proposes to match one-step gradients, while MTT () advances this concept by matching entire training trajectories. Although trajectory matching methods are often regarded as the current SOTA framework for dataset distillation, our experiments reveal that gradient matching methods outperform them on large-scale datasets and high-IPC settings. Furthermore, gradient matching methods demonstrate superior generalization across multiple architectures, which is why we mainly implement our method to gradient matching frameworks.

Distribution Matching: An alternative approach involves aligning the feature maps of real and synthetic data. Early methods such as CAFE and DM employed this strategy but showed inferior performance compared to gradient or trajectory matching frameworks. Recent advances in this domain propose aligning the batch normalization statistics of real and synthetic images to generate synthetic data. However, these methods are highly memory-intensive, requiring roughly 40 times more memory to store soft labels than synthetic images themselves. Additionally, the synthetic images generated using this approach often perform worse than randomly selected images. Given these limitations, we conclude that distribution matching methods are not well-suited for dataset distillation. Therefore, our work focuses on enhancing traditional gradient/trajectory matching approaches.

3. Preliminary

Dataset distillation is a task of generating a batch of synthetic images $\mathcal{S}$ such that networks trained with $\mathcal{S}$ perform similar to networks that are trained with the original training set $\mathcal{R}$. However, directly optimizing with respect to the generalization performance is very expensive, so surrogate objectives are solved instead to minimize distance between the real and synthetic dataset.

$$R : \text{Real Dataset} \tag{1}$$

$$S : \text{Synthetic Dataset} \tag{2}$$

$$S^* = \arg\min_S \mathcal{D}(S, R) \tag{3}$$

The distance between the datasets can be measured in term of feature maps, gradients, or training trajectories.

The DASM framework can be implemented into all three surrogate objective methods, but as stated earlier, we work

on gradient matching methods due to its superior performance and cross architecture generalization. In particular, we build upon IDC, where gradient matching is performed with downsampled synthetic images.

$$S^* = \arg\min_S \mathcal{D}\left(\nabla_\theta \mathcal{L}(\theta(\mathcal{A}^*(S))), \nabla_\theta \mathcal{L}(\theta(\mathcal{A}(R)))\right) \quad (4)$$

$\mathcal{A}$ represents differential siamese augmentation, and $\mathcal{A}^*$ is upsampling followed by $\mathcal{A}$.

References

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

A. You *can* have an appendix here.

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one.

The `\onecolumn` command above can be kept in place if you prefer a one-column appendix, or can be removed if you prefer a two-column appendix. Apart from this possible change, the style (font size, spacing, margins, page numbering, etc.) should be kept the same as the main body.