Dataset Distillation by Learning Exclusive Information

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

Dataset distillation aims to generate a small synthetic dataset such that models trained on it achieve performance comparable to those trained on the real dataset. Existing methods () generate synthetic datasets by updating synthetic images to match training properties between synthetic and real datasets, such as gradients, feature maps, or training trajectories generated when training models. The synthetic dataset can then be used instead of the real dataset for various tasks such as continual learning (), privacy preservation (), or neural architecture search ().

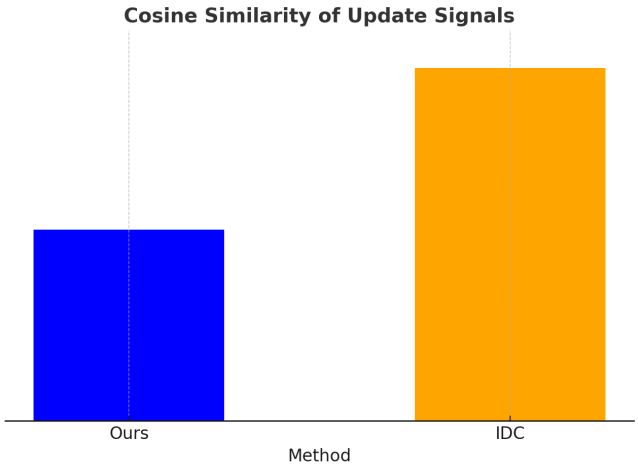
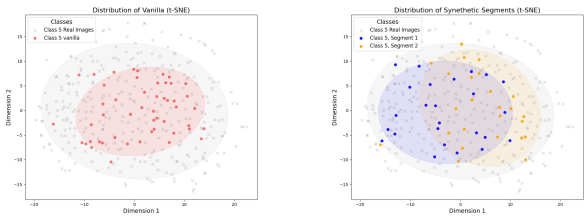


Figure 1. Cosine similarity of back-propagation signals of OUR and IDC method. Back-propagation signals are obtained by subtracting initialized images from the generated synthetic images, where both OUR and IDC are initialized with the same set of images. The cosine similarity is averaged for all classes.

Although existing dataset distillation methods () demonstrate strong performance when generating synthetic datasets with few images per class (IPC), they fail to scale effectively to higher IPC settings. This limitation arises because synthetic images are updated together in a single batch, which causes the individual synthetic images to become similar. The synthetic images are simultaneously updated to minimize the same loss function, resulting in similar back-propagation signals across the batch of synthetic images. Consequently, redundant information is embedded in the synthetic images. In contrast, our method divides the synthetic images into exclusive subsets. Then each subset is updated using a different loss, which allows the synthetic images to be updated with diverse back-propagation signals. This is illustrated in Fig. 1, where the average cosine similarity of back-propagation signals are plotted for IDC () and our method. The back-propagation signals

[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.



(a) IDC (b) OURS

Figure 2. T-SNE distribution of synthetic images generated from using our and IDC method. IDC generates synthetic images that are clustered around the center, while our method generates synthetic images that cover a wide distribution. Note that images generated in different segments tend to cover different parts of the distribution.

are measured by subtracting the initialized image from the generated synthetic image. High cosine similarity of IDC indicates that the synthetic images are being updated in a similar direction, while low cosine similarity of our method shows that synthetic images are updated towards different directions. Furthermore, Fig. 2 presents T-SNE () distribution of synthetic dataset generated by IDC and our method. IDC generates synthetic images that are close in distribution, which indicate the generated synthetic images are similar to each other. On the other hand, our method generates synthetic images that cover a wide range of distribution, demonstrating that it captures diverse aspects of the real dataset.

In this paper, we propose **EDD** (Exclusive Dataset Distillation), a novel framework that can be orthogonally integrated into many existing works (). To address the issue of shared loss, synthetic images are updated in small exclusive subsets, which refer to as segments. Each segment is then trained by matching different subsets of the real dataset, encouraging the segments to learn different parts of the real dataset. The process begins by generating the first segment of synthetic images. A model is trained using this segment, and the trained model is then applied to classify images from the real dataset. Among the classified images, a portion of those that are correctly predicted by the model are filtered out, while the misclassified images are retained. These retained, images form a filtered subset of the real dataset, which is then used to train the next segment of synthetic images. This process of training a segment, classifying the real dataset, discarding images, and using the filtered real dataset for the next segment is repeated iteratively. This cycle continues until all segments are produced, ensuring that each segment focuses on learning from parts of the real dataset that previous segments struggled with. However, naively training subsequent segments with the filtered dataset results in performance degradation compared to baseline methods. This is mainly due the 'forgetting' phenomenon () which will be explained in more detail in section 4. To state briefly, adding new segments to existing ones causes the model to misclassify images it previously classified correctly. To mitigate forgetting, we design a regularization loss that is derived from the earlier segments. By applying this loss during the training of new segments, we significantly reduce forgetting and achieve better overall performance. When integrated into existing approaches, our method demonstrates competitive performance across many baseline methods, especially for datasets with many classes in high IPC regions. We summarize our contributions as follows:

- We propose EDD, a novel framework that allows embedding of exclusive information into synthetic images by training segments with different subsets of the real dataset.
- To address the forgetting phenomenon, we add a regularization loss, which reduces the number of forgotten images when adding new segments.
- Our method shows competitive performance across a variety of datasets, and especially performs well on datasets with many classes. On the tiny-imagenet 50 ipc setting, our method achieves 44.3% accuracy, surpassing the current SOTA by a 4.6% margin.

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. DC emphasizes matching 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: Another approach focuses on aligning the feature maps of real and synthetic data. Early methods such as CAFE and DM adopted this strategy but exhibited inferior performance compared to gradient or trajectory matching frameworks. More recent works propose a novel take on distribution matching, generating synthetic images by aligning the batch normalization statistics of real images.

However, these methods are highly memory-intensive, requiring approximately 40 times more memory to store soft labels than is needed for synthetic images. Moreover, synthetic images generated using this approach often perform worse than randomly selected images. Based on these findings, we conclude that this line of work is not well-suited for dataset distillation. Consequently, our focus remains on traditional gradient/trajectory matching methods.

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.

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

$\mathcal{A}$ 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.