Feature extractor comparison for distribution matching framework in dataset distillation

Muyang Li[1,2], Zeheng He[3], Yi Qu[2,4✉], and Yong Shi[2,4]

[1] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100190, China
[2] Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences, Beijing 100190, China
[3] Faculty of Information Technology, Monash University, Melbourne Victoria 3168, Australia
[4] School of Economics and Management, University of Chinese Academy of Sciences, Beijing 100190, China
quyi@ucas.ac.cn

Abstract. Dataset distillation is a technique to generate compact synthetic datasets which enable efficient model training and knowledge transfer. It relies on two critical procedures in distribution matching frameworks: feature extraction and distribution alignment. Previous studies have less attention on systematical investigations about how different feature extractors influence the performance of distilled datasets. To enrich research in this filed, this paper conducts comprehensive comparison of four feature extractors including convolutional neural network (CNN), ResNet-18, multilayer perceptron (MLP) and lightweight Vision Transformer (ViT), and further analyzes the impact of dynamic or fixed feature extractors. Experimental results indicates the optimal performance of ConvNet and finding that slight pre-training of feature extractors using image classification tasks can promote the performance of distilled datasets. This work provides empirical guidance for appropriate feature extractors selection in distribution matching frameworks of dataset distillation.

Keywords: Dataset distillation · Distribution matching · Feature extractor selection · Representation Learning

1 Introduction

As a thriving research field in computer vision, dataset distillation has been attracting increasingly significant research interests. The main challenge of dataset distillation lies in how to transfer crucial information from the original dataset $\mathcal{T}$ to the distilled one $\mathcal{S}$. Several frameworks have been proposed for dataset distillation problem, including bi-level optimization framework [12], kernel-based optimization framework [9], parameter matching framework [1,15] and distribution matching framework [14,16].

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

By considering dataset $\mathcal{X}$ as a sample of the distribution $\mathcal{D}_{\mathcal{X}}(\mathcal{X} \in \{\mathcal{S}, \mathcal{T}\})$, the workflow of distribution matching framework (Figure 1) can be described as follows [14]:

Step 1: Extracts representation vectors via feature extractors from datasets $\mathcal{T}$ and $\mathcal{S}$ respectively.
Step 2: Construct the dataset $\mathcal{S}$ by matching these two distributions $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{D}_{\mathcal{S}}$.

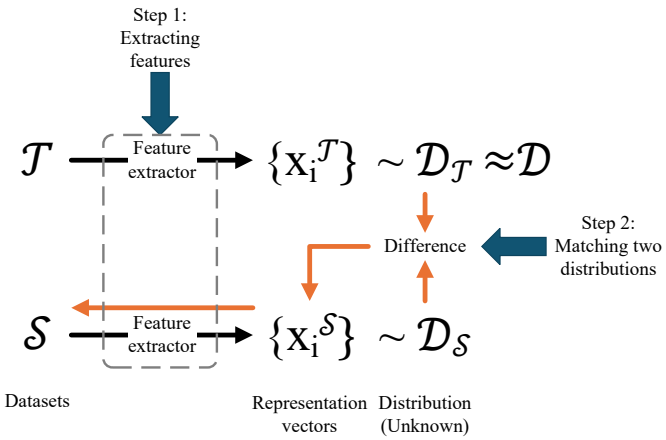


Fig. 1: Workflow of distribution matching framework. *Black arrows indicate the forward propagation process, orange arrows represent the backward propagation process.*

The procedure of training a neural network can be regarded as the network learning the distribution of the dataset [3]. Thus the distribution matching framework can be considered as extracting the distribution of the dataset $\mathcal{T}$ and reconstructing the distribution in the dataset $\mathcal{S}$.

Existing works have mainly focused on the design of matching target while ignoring the importance of feature extraction [11,13,14]. In [16], the author provides a brief comparison regarding the performance of distilled dataset with respect to the training epoch iteration.

In this paper, we propose that different feature extractor impact the performance of distilled dataset. To verify the validity of this opinion, we carefully compare the performance of distilled dataset under different feature extractors. We select MLP, ConvNet, ResNet-18 and lightweight ViT as candidates.

The remainder of this paper is organized as follows: In section 2, we will review some existing works on distribution matching approach. Section 3 will

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

introduce the experimental setup and details. The experimental results and relevant discussions will be presented in section 4. Finally, section 5 will conclude the work of this paper.

2 Related Work

2.1 Dataset Distillation

As illustrated in section 1, dataset distillation refers to the task of constructing another dataset $\mathcal{S}$ for a given dataset $\mathcal{T}$, such that $\mathcal{S}$ serves as an alternative to $\mathcal{T}$ [12]. Here alternative means that a neural network trained on $\mathcal{S}$ can achieve comparable performance to one trained on $\mathcal{T}$. This process involves distilling the essential knowledge from the original dataset $\mathcal{T}$ into a smaller synthetic dataset $\mathcal{S}$, which retains the critical statistical properties and representational power required to train models with equivalent efficacy.

Mathematically, the goal of dataset distillation can be expressed as:

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}[\text{loss}(f_{\mathcal{S}}(x), y)] \approx \mathbb{E}_{(x,y)\sim\mathcal{D}}[\text{loss}(f_{\mathcal{T}}(x), y)] \tag{1}$$

Here $\mathcal{D}$ is the real distribution of dataset $\mathcal{T}$, $f_{\mathcal{X}}$ describes the neural network trained on dataset $\mathcal{X}$ ($\mathcal{X} = \mathcal{T}/\mathcal{S}$), thus $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\text{loss}(f(x), y)]$ represents the expectation risk of model f.

Distribution matching (DM) framework aims to achieve dataset distillation by aligning the distributions of the synthesized dataset $\mathcal{S}$ and the original dataset $\mathcal{T}$.

2.2 Distribution matching

The vanilla DM method synthesizes the distilled dataset $\mathcal{S}$ based on the optimization of the Maximum Mean Discrepancy (MMD) [4], minimizing the MSE between the mean of the data representation distributions in $\mathcal{T}$ and $\mathcal{S}$. The MMD between two distributions is defined by:

$$\text{MMD} = \sup_{\|\psi\|_{\mathcal{H}}\leq 1} (\mathbb{E}_{\mathcal{T}}[\psi(x)]) - (\mathbb{E}_{\mathcal{S}}[\psi(x)]) \tag{2}$$

Here $\psi(\cdot)$ is the feature extractor, $\mathbb{E}_{\mathcal{X}}(\mathcal{X} = \mathcal{T}/\mathcal{S})$ refers to the expectation calculated based on dataset $\mathcal{X}$. In practical applications, the empirical estimate of the MMD is used [14].

$$S^* = \arg\min_{\mathcal{S}} \mathbb{E}\| \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\psi(x_i) - \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|}\psi(x_i)\|^2 \tag{3}$$

Furthermore, Improvements to vanilla DM mainly focus on two categories. One is extracting more features: CAFE [11] aligns all feature maps in the same feature extractor between samples in $\mathcal{T}$ and $\mathcal{S}$; Improved distribution matching (IDM) [16] enhances image minibatches and increases feature extractors,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

adding crossentropy loss to alignment features. The other is using better alignment methods: Inspired by transfer learning, M^{3}D maps the distribution to a reproducing kernel Hilbert space to measure distribution distance [13]. In existing research, the Wasserstein distance is also used for better alignment result [8].

3 Experiments

3.1 Experiment settings

All experiments are conducted on dataset CIFAR-10 [6], which consists of 10 classes, each class contains 5,000 32×32 images.

In [14], the author utilizes the first $n-1$ layers of the ConvNet as the feature extractor and treats the last fully connected layer as a classifier. In this paper, following a similar approach, we also regard the first $n-1$ layers of the neural network as the feature extractor, and take their output as the representation of the input data.

We select ConvNet [7], ResNet-18 [5], MLP [10] and lightweight ViT as feature extractors. The ConvNet consists of three identical convolution blocks, each has a 3×3 convolution layer, an instance normalization layer, a ReLU activation operation and an average pooling operation sequentially (see Figure 2). The MLP in our experiment consists of 2 hidden layers, both have 128 hidden dimensions with activation function ReLU. The ResNet-18 is the standard original form. The lightweight ViT network takes input images of size 32×32, with each patch sized 4×4. The output dimension of the embedding layer in this network is 192. It consists of 12 transformer blocks, where the dimension of the feed-forward network within each transformer block is 768. For the multi-head attention mechanism, the network incorporates 12 attention heads(Tab. 1).

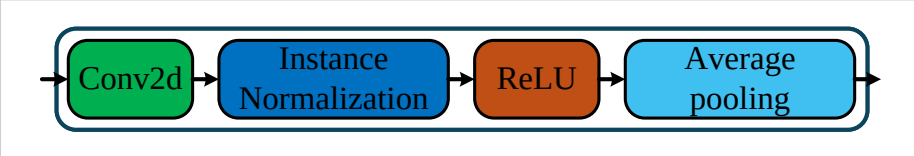


Fig. 2: One convolution block of ConvNet

In the experiments of this paper, two kinds of feature extractors are used: the fixed feature extractors and the dynamic ones. The fixed feature extractors refer to those whose parameters remain frozen throughout the entire experiment. Specifically, they are the embedding layers of corresponding pre-training neural networks. In contrast, when a dynamic feature extractor is used for feature extraction, it always starts with a specified neural network that is randomly initialized. After training the network for the given number of epochs, its embedding layers are employed as the feature extractor.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

Hyperparameters	values
input size	32×32
patch size	4×4
transformer blocks	12
feed-forward dimension	768
embedding dimension	192
number of heads	12

Table 1: Hyperparameters of lightweight ViT

3.2 Experimental content

In this section, we explain the specific experimental details. This study comprises three sets of experiments, as outlined below:

The first set of experiments compares the performance differences of distilled datasets under various combinations of feature extractors and networks. For networks, we select ConvNet and ResNet-18. The feature extractors are dynamic ConvNet and ResNet-18, both of which are trained on the CIFAR-10 dataset for 0, 1, 2, and 3 epochs respectively.

The second set of experiments focuses on the performance of the distilled dataset generated from using dynamic and fixed feature extractor. The performance evaluation is conducted by training ConvNet. The feature extractors in this part include ConvNet, ResNet-18, and MLP, all of which are trained on the CIFAR-10 dataset for 0, 1, 2, and 3 epochs respectively.

The third set of experiments compares the influence of different pre-training epochs of feature extractors on the performance of the distilled dataset, with the performance evaluation implemented by training ConvNet. The feature extractors include ConvNet, ResNet-18, MLP, and lightweight ViT. For fixed feature extractor, they are trained on the CIFAR-10 dataset for 0-5, 10, 50, and 100 epochs respectively. For dynamic feature extractors, they are trained on the CIFAR-10 dataset for 0, 1, 2, and 3 epochs respectively. Due to constraints on training time, lightweight ViT is not included in the dynamic feature extractors in this experiment set.

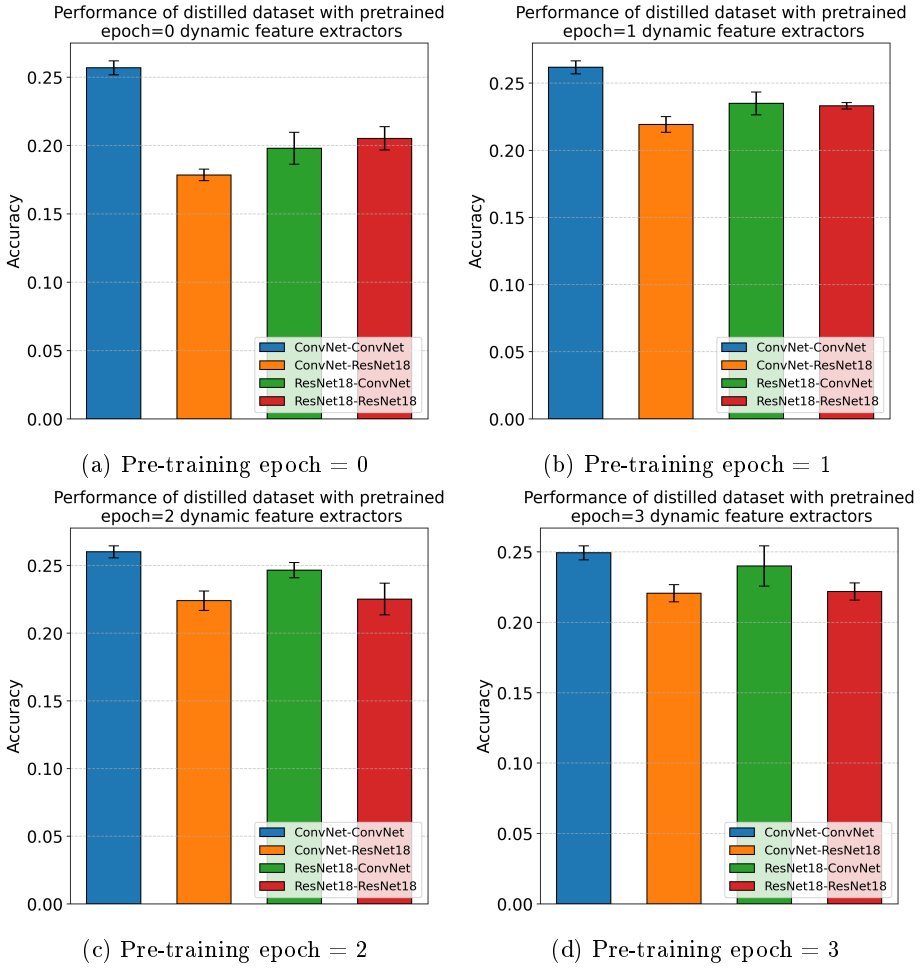
A summary of the above experimental contents is provided in Tab. 2.

4 Results and discussion

4.1 Comparison between different network-feature extractor combinations

Figure 3 demonstrates the performance of distilled dataset in experiment set 1. The figure shows that using ConvNet as the feature extractor and training the ConvNet (the blue bar) exhibits the optimal performance across all scenarios where the number of pre-training epochs is 0, 1, 2, and 3. The performance of the ConvNet-ConvNet combination does not show a significant increase with

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8



(a) Pre-training epoch = 0

(b) Pre-training epoch = 1

(c) Pre-training epoch = 2

(d) Pre-training epoch = 3

Fig. 3: Performance of dynamic feature extractor with different pre-training epochs. *The legends have form "network-feature extractor"*

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

Experiment set	Network	Feature extractor	Fixed/Dynamic	Epochs
Architecture	ConvNet	ConvNet ResNet-18	Dynamic	0,1,2,3
	ResNet-18	ConvNet ResNet-18		
Dynamic/Fixed	ConvNet	ConvNet	Dynamic Fixed	
		ResNet-18	Dynamic Fixed	
		MLP	Dynamic Fixed	
Epoch	ConvNet	ConvNet ResNet-18 MLP lightweight ViT	Fixed	0,1,2,3,4,5 10,50,100

Table 2: Summary of experiments

the growth of the pre-training epochs of the feature extractor. However, the performance of the other three combinations is relatively poor when the pre-training epochs of the feature extractor are 0 ($\leq 20\%$), but there is a notable improvement ($\geq 20\%$) once the pre-training epochs exceed 0. We argue that this indicates that a slight pre-training of the feature extractor can effectively enhance the performance of the distilled dataset.

4.2 Comparison between dynamic and fixed feature extractor

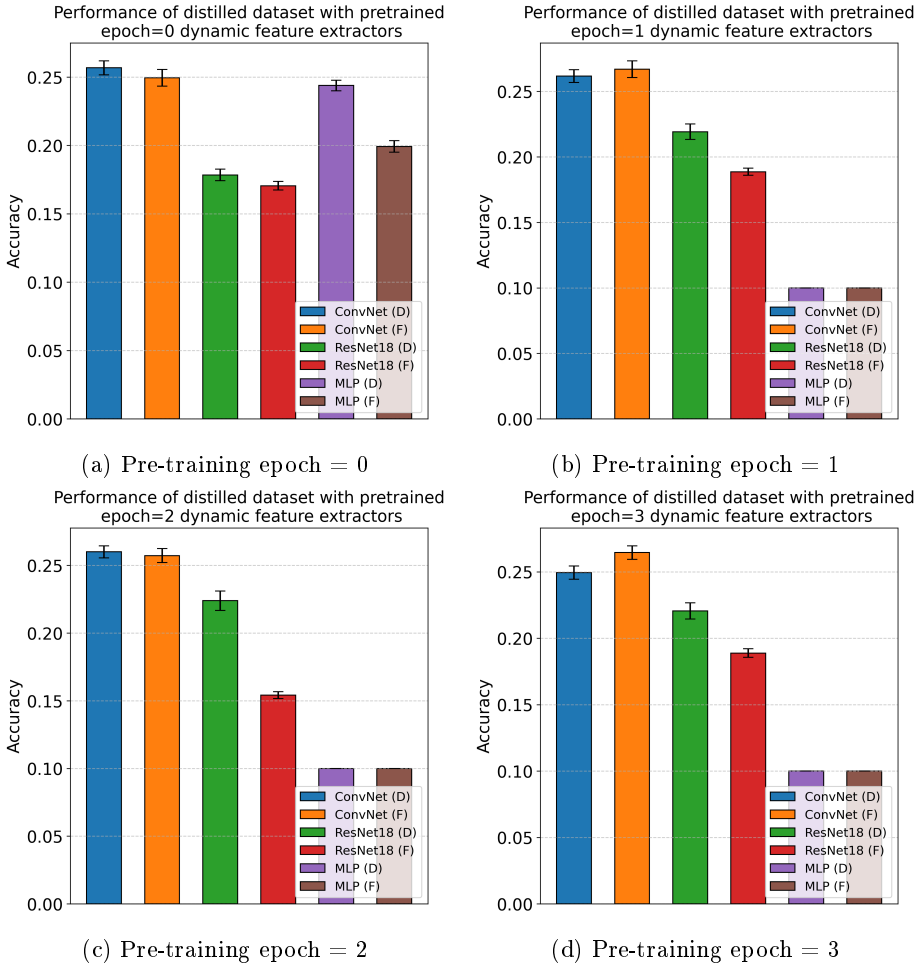
Figure 4 demonstrates the performance of distilled dataset in experiment set 2.

As shown in the figure, when the feature extractor is ConvNet, there is no significant difference in the performance of the distilled dataset between the dynamic ConvNet and the fixed ConvNet. However, both consistently outperform the distilled datasets generated by other feature extractors.

When the feature extractor is ResNet-18, increasing the pre-training epochs from 0 to 3 consistently improves the performance of the distilled dataset, and the performance improvement of the dynamic ResNet-18 is significantly greater than that of the fixed ResNet-18.

When the feature extractor is MLP, the dynamic MLP yields a well-performing distilled dataset only when the pre-training epoch is 0, with such performance even approaching that of the best-performing ConvNet feature extractor. In contrast, the fixed MLP results in relatively poor performance of the distilled dataset but still better than that with feature extractor ResNet-18. It is also worth noting that due to the strong fitting capacity of MLP, even though the small-scale MLP is used, the use of pre-trained MLP embedding layers as feature extractors ultimately led to gradient explosion in all experiments, as detailed in Figure

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8



(a) Pre-training epoch = 0

(b) Pre-training epoch = 1

(c) Pre-training epoch = 2

(d) Pre-training epoch = 3

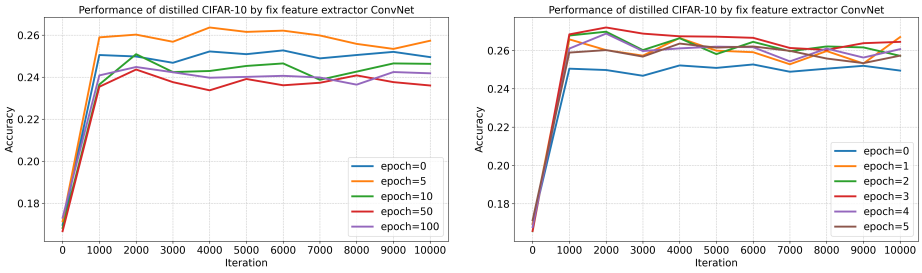
Fig. 4: Comparison between dynamic and fixed feature extractor with network ConvNet. *The legends has form "feature extractor(Dynamic/Fixed)"*

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

4b-4d. Therefore, the possibility of using the embedding layers of pre-training MLPs as feature extractors is not discussed further in this paper.

4.3 Comparison between different training epoch of fixed feature extractor

Figure 5-7 illustrate the performance of dataset distillation tasks using the embedding layers of fixed neural networks trained for different epochs as feature extractors. In this section, we will discuss performance of each feature extractors, respectively.

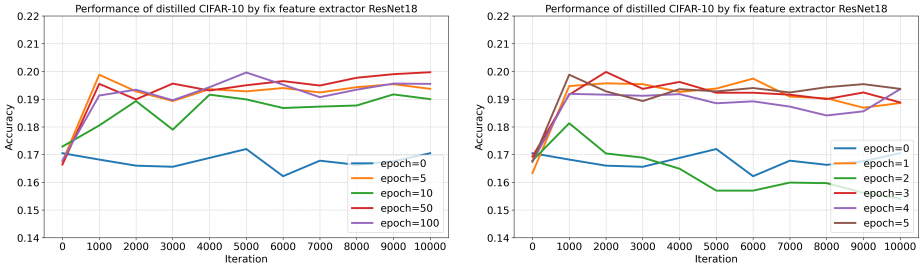


(a) (b)

Fig. 5: Performance of fixed trained feature extractor ConvNet

For ConvNet, Figure 5a shows that when a fixed ConvNet is employed as the feature extractor, the best distillation performance is achieved at 5 training epochs, followed by that at 0 epochs. When the number of epochs exceeds 5, the performance of the dataset distillation task is even worse than that of the random ConvNet at 0 epochs; moreover, a larger number of training epochs leads to additional time cost. We attribute this phenomenon to overfitting caused by large training epochs and the downstream task of "classification", which drives data from different classes to be as divergent as possible in the representation space. To further investigate, we conducted a more detailed comparison of ConvNet feature extractors with pre-training epochs ranging from 0 to 5, and the results are presented in Figure 5b. The results indicate that the feature extractor with 3 pre-training epochs outperforms those with 1, 2, 4, and 5 pre-training epochs in the dataset distillation task, while the latter perform better than the random feature extractor with 0 pre-training epochs. Therefore, we conclude that when using ConvNet as the feature extractor, all produce relatively optimal hyperparameters.

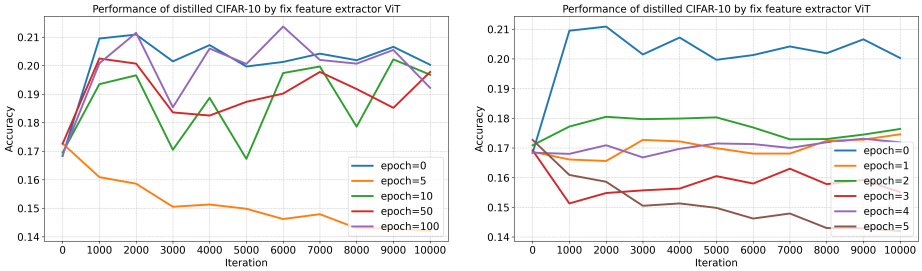
ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8



(a) (b)

Fig. 6: Performance of fixed trained feature extractor ResNet-18

For ResNet, Figure 6b demonstrates that the random feature extractor with 0 pre-training epochs exhibits significantly inferior performance in the dataset distillation task compared to pre-training feature extractors, and the performance of the dataset distillation task shows an upward trend as the number of pre-training epochs increases.



(a) (b)

Fig. 7: Performance of fixed trained feature extractor ViT

For lightweight ViT, Figure 7b reveals that it seems that there is no significant correlation between the performance of the dataset distillation task and the number of pre-training epochs for lightweight ViT as a feature extractor. We hypothesize that this phenomenon is directly associated with the fact that the lightweight ViT model can only exhibit strong representational capabilities when pre-training on a large volume of data [2].

5 Conclusion and future work

This paper conducts a comparative study on the impact of different feature extractors on the performance of distilled datasets obtained through the distribu-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

tion matching framework for dataset distillation tasks. The comparison reveals that ConvNet achieves the optimal performance among ConvNet, ResNet-18, MLP, and lightweight ViT. Meanwhile, slight pre-training (1-3 epochs) of feature extractors using image classification tasks can significantly enhance the performance of distilled datasets.

The comparison of feature extractors in this paper has certain limitations. This work lacks the exploration of precision structures in feature extractors, and the downstream tasks for pre-training feature extractors only contain image classification. In future research, we aim to conduct the precision structures of feature extractors to investigate how different micro-structures influence their performance. Additionally, we will leverage autoregressive encoder architectures to add image reconstruction and other generative tasks into downstream tasks, thereby examining the impact of diverse downstream tasks on the efficacy of pre-training feature extractors.

Acknowledgment

This study has been funded by Key Projects of National Natural Science Foundation of China (#72231010, #71932008).

References

1. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Dataset distillation by matching training trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 4750–4759 (June 2022)
2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2021), https://arxiv.org/abs/2010.11929
3. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org
4. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A.: A kernel two-sample test. J. Mach. Learn. Res. **13**(null), 723–773 (Mar 2012)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
6. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images.(2009) (2009)
7. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (2002)
8. Li, M., Xue, J., Shi, Y.: Dataset distillation via kantorovich-rubinstein dual of wasserstein distance. In: Paszynski, M., Barnard, A.S., Zhang, Y.J. (eds.) Computational Science – ICCS 2025 Workshops. pp. 293–306. Springer Nature Switzerland, Cham (2025)
9. Nguyen, T., Chen, Z., Lee, J.: Dataset meta-learning from kernel ridge-regression. In: International Conference on Learning Representations (2021)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8

10. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review **65**(6), 386 (1958)
11. Wang, K., Zhao, B., Peng, X., Zhu, Z., Yang, S., Wang, S., Huang, G., Bilen, H., Wang, X., You, Y.: Cafe: Learning to condense dataset by aligning features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12196–12205 (June 2022)
12. Wang, T., Zhu, J.Y., Torralba, A., Efros, A.A.: Dataset distillation. arXiv preprint arXiv:1811.10959 (2018)
13. Zhang, H., Li, S., Wang, P., Zeng, D., Ge, S.: M3d: Dataset condensation by minimizing maximum mean discrepancy. Proceedings of the AAAI Conference on Artificial Intelligence **38**(8), 9314–9322 (Mar 2024). https://doi.org/10.1609/aaai.v38i8.28784, https://ojs.aaai.org/index.php/AAAI/article/view/28784
14. Zhao, B., Bilen, H.: Dataset condensation with distribution matching. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 6514–6523 (January 2023)
15. Zhao, B., Mopuri, K.R., Bilen, H.: Dataset condensation with gradient matching. arXiv preprint arXiv:2006.05929 (2020)
16. Zhao, G., Li, G., Qin, Y., Yu, Y.: Improved distribution matching for dataset condensation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7856–7865 (June 2023)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_8