Dataset Distillation via Kantorovich-Rubinstein Dual of Wasserstein Distance

Muyang Li[1,3,4], Jiayu Xue[1,3,4], and Yong Shi[2,3,4]

[1] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, 100190, China
[2] School of Economics and Management, University of Chinese Academy of Sciences, Beijing, 100190, China
[3] Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences, Beijing, 100190, China
[4] Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing, 100190, China
limuyang23@mails.ucas.ac.cn

Abstract. The exponential scaling of dataset volumes in contemporary deep learning imposes great computational and storage burdens across learning paradigms, which emphasizes the importance of intelligent dataset compression methods. Dataset distillation(DD) emerges as a promising solution for dataset size reduction. This study focus on the distribution matching framework for DD, introducing a novel methodology that quantifies the inter-distribution difference between source and distilled datasets via optimal transport theory, where Wasserstein metric W_1 serves as the discrepancy measurement. We implement this metric via the Kantorovich-Rubinstein(KR) dual $\sup_{f\in \mathrm{Lip}(\Omega)} \mathbb{E}_{\mu_\mathcal{T}}[f] - \mathbb{E}_{\mu_\mathcal{S}}[f]$. According to Universal Approximation Theorem, a single-hidden-layer multilayer perceptron(MLP) with non-polynomial activation function can approximate continuous functions with arbitrary precise, thus a single-hidden-layer MLP is selected to approximate the function f in the expression of KR dual while maintaining its Lipschitz continuity through a parameter truncation technique. Empirical evaluations demonstrate that our approach achieves performance comparable to the mainstream benchmarks. The empirical findings of this study validates the operational feasibility of employing Wasserstein distance and KR dual in DD problem. Related code is available at `https://github.com/muyangli17/DD-with-KR-dual`.

Keywords: dataset distillation · distribution matching · Optimal Transport · Wasserstein distance · Kantorovich-Rubinstein Dual

1 Introduction

In recent years, the exponential growth of data required for deep learning has directly resulted in increasing storage costs within data centers. From ImageNet (14M samples) [3] to LAION-5B (5.8 billion samples) [17], the dataset scale has

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

expanded by 410 times over past 13 years. This has pushed dataset compression techniques to the forefront of academic research.

Dataset compression aims to identify a smaller data set $\mathcal{S}$ that can effectively replace an original large-scale dataset $\mathcal{T}$ ($\|\mathcal{S}\| \ll \|\mathcal{T}\|$), where this replacement specifically requires that models trained on $\mathcal{S}$ maintain comparable generalization performance to their counterparts trained on $\mathcal{T}$.

Current mainstream approaches of dataset lightweighting focus on two categories: coreset selection [6,9] and dataset distillation (DD) [20]. The former involves identifying a representative subset of $\mathcal{T}$, while the latter synthesizes new artificial samples through feature recombination. The coreset selection problem, inherently NP-hard in nature, poses significant scalability challenges for large-scale datasets [5], while its empirical performance typically underperforms dataset distillation approaches under equivalent conditions[10]. Table 1 shows the accuracy of the state-of-the-art coreset selection method and the dataset distillation method on image classification problems. Concurrently, dataset distillation has emerged as a rapidly growing research frontier, garnering substantial attention in recent years.

Table 1. Comparison of the accuracy of the state-of-the-art coreset selection method and the dataset distillation method on image classification problems

Datsaset	IPC	coreset selection	dataset distillation	Whole
MNIST	1	$89.2\%_{\pm 1.6}$	$98.7\%_{\pm 0.7}$	$99.6\%_{\pm 0.0}$
	10	$95.1\%_{\pm 0.9}$	$99.3\%_{\pm 0.5}$	
	50	$97.9\%_{\pm 0.2}$	$99.4\%_{\pm 0.4}$	
FashionMNIST	1	$67.0\%_{\pm 1.9}$	$88.5\%_{\pm 0.1}$	$93.5\%_{\pm 0.1}$
	10	$71.1\%_{\pm 0.7}$	$90.0\%_{\pm 0.7}$	
	50	$71.9\%_{\pm 0.8}$	$91.2\%_{\pm 0.3}$	
SVCH	1	$20.9\%_{\pm 1.3}$	$87.3\%_{\pm 0.1}$	$95.4\%_{\pm 0.1}$
	10	$50.5\%_{\pm 3.3}$	$91.4\%_{\pm 0.2}$	
	50	$72.6\%_{\pm 0.8}$	$92.2\%_{\pm 0.1}$	
CIFAR-10	1	$21.5\%_{\pm 1.2}$	$66.4\%_{\pm 0.4}$	$84.8\%_{\pm 0.1}$
	10	$31.6\%_{\pm 0.7}$	$72.0\%_{\pm 0.3}$	
	50	$43.4\%_{\pm 1.0}$	$75.0\%_{\pm 0.2}$	
CIFAR-100	1	$8.4\%_{\pm 0.3}$	$40.0\%_{\pm 0.5}$	$56.2\%_{\pm 0.3}$
	10	$17.3\%_{\pm 0.3}$	$50.6\%_{\pm 0.2}$	
	50	$33.7\%_{\pm 0.5}$	$47.0\%_{\pm 0.2}$	

Building upon optimal transport theory, we approximate the distributional difference between source dataset $\mathcal{T}$ and synthetic dataset $\mathcal{S}$ through the Kantorovich-Rubinstein duality formulation of Wasserstein distance, establishing a novel DD method grounded in this geometric metric.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

The Wasserstein distance is given by:

$$W_p(\mu,\nu) = \inf_{\gamma\in\Gamma}\left(\int_{x,y\sim\gamma}\gamma(x,y)d(x,y)^p dxdy\right)^{\frac{1}{p}}.$$

Here μ and ν are two distributions, Γ is the collection of all binary distribution with marginal distribution μ and ν respectively.

For $p = 1$ case of Wasserstein distance, W_1 distance can be written as Kantorovich-Rubinstein dual form[18]:

$$W_1 = \sup_{f\in \mathrm{Lip}(\Omega)}\left(\mathbb{E}_\mu(f) - \mathbb{E}_\nu(f)\right).$$

According to Universal Approximation Theorem[2], MLPs can approximate any continuous function on a compact set. This inspires us to parameterize a Lipschitz function f with a MLP(see section 3 for details). Compared to traditional linear programming solutions, the Kantorovich-Rubinstein dual(KRdual) form constructs functions in the Lipschitz function space through deep neural networks and weight truncation methods. The Wasserstein distance calculation based on deep neural networks can be seamlessly embedded into the original dataset distillation deep learning framework, and end-to-end training is realized through the automatic differentiation mechanism, supporting multi-GPU data parallelism, which is an advantage that traditional linear programming solutions and Sinkhorn approximations cannot achieve. Although the computational complexity of Wasserstein distance is relatively high, the actual training time is still within an acceptable range through GPU parallel algorithms and a relatively simple MLP structure. The flowchart of the method proposed in this paper is shown in Figure 1.

Experiments show that method proposed in this paper achieves the comparable accuracy as the mainstream DD method on multiple datasets for image classification problems. The full code can be found at: `https://github.com/muyangli17/DD-with-KR-dual`, and the specific training hyperparameters are detailed in section 4.

The remainder of this paper is organized as follows: In section 2, we will briefly review several mainstream research frameworks in the field of DD. In section 3, we will detail our specific method. The relevant experimental results and discussions will be presented in section 4. Finally, in section 5, we will briefly summarize our work.

2 Related work

2.1 Dataset distillation

The primary methods for DD currently include meta-learning approaches, parameter matching methods, and distribution matching methods[15].

For the meta-learning approach, according to the different inner model, methods are mainly divided into two categories: the backpropagation through time

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

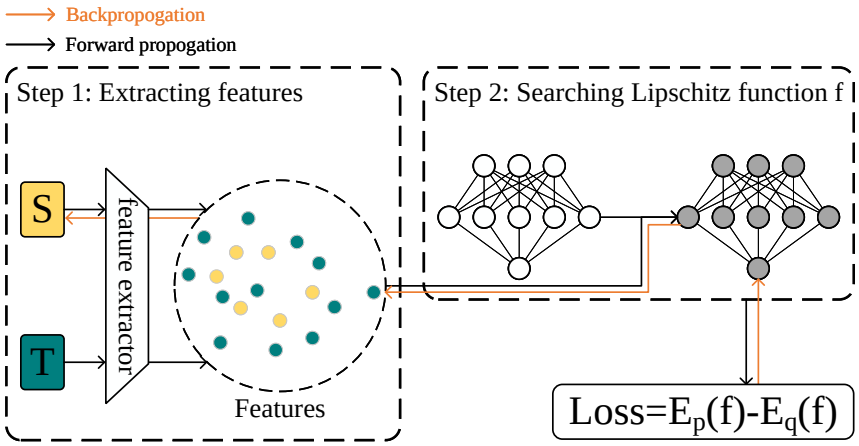

Fig. 1. Main flow of our method

(**BPTT**) and the kernal-based optimization. BPTT contains the vanilla **DD**[20] and **RaT-BPTT**[4] method that uses a small neural network as a classifier to realize the inner loop. The kernal-based optimization[13,14] methods use kernel ridge regression as the inner loop to realize the classification.

For the parameter matching approach, **DC**[25] focus on the matching of gradient and **MTT**[1] focus on the matching of training trajectories.

The distribution matching approach synthesizes the distilled dataset $\mathcal{S}$ by aligning the distributions between the original dataset $\mathcal{T}$ and $\mathcal{S}$, the approach under which the methodology proposed in this paper falls.

2.2 Distribution matching

Based on the optimization of the Maximum Mean Difference (MMD), the vanilla distribution matching approach(**DM**) synthesizes the distilled dataset $\mathcal{S}$ by minimizing the MSE between the mean of the distribution of the data representation in the source dataset $\mathcal{T}$ and $\mathcal{S}$[24].

Most of the improvements to the vanilla DM have focused on increasing the total amount of features that can be aligned. These include increasing the richness of feature extractors in the **IDM**[26] and increasing the number of alignable feature layers in **CAFE**[19].

In addition to the refinement of the vanilla DM method, the **M^{3}D** starts directly from the definition of MMD and maps the distribution to its corresponding reproducing kernel hilbert space (RKHS) by appling the reproducing kernel. Aligning the two distributions efficiently by measuring the distance in the Hilbert space[22].

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

3 Methodology

According to Figure 1, our method contains two stages: feature extraction and wasserstein distance computation.

3.1 Extracting feature

In the feature extraction stage, previous work has pointed out that randomly initialized neural networks can generate high-quality representations for images[16]. Therefore, this paper selects a randomly initialized convolutional neural network as the feature extractor to map image data into a 2048-dimensional vector space. Let the original dataset and the synthetic dataset follow the distributions $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$, respectively, then the vectors $\{x_i\}$ and $\{y_i\}$ mapped from the original dataset and the synthetic dataset into the representation space can be regarded as a sample from $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$.

3.2 Constructing loss function based on Wasserstein distance

Given distributions $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$, a *transportation* from $\mu_{\mathcal{T}}$ to $\mu_{\mathcal{S}}$ is defined as a binary probability distribution $\gamma(x, y)$ satisfying

$$\int_y \gamma(x,y) = \mu_{\mathcal{T}}, \int_x \gamma(x,y) = \mu_{\mathcal{S}}$$

The definition of Wasserstein distance can be expressed as follow:

$$W_p(\mu_{\mathcal{T}}, \mu_{\mathcal{S}}) = \inf_{\gamma \in \Gamma} \left(\int_{x,y\sim\gamma} \gamma(x,y) d(x,y)^p dxdy \right)^{\frac{1}{p}} \tag{1}$$

Here Γ is the set of all *transportation* from $\mu_{\mathcal{T}}$ to $\mu_{\mathcal{S}}$. $d(x, y)$ describes the cost of transporting unit mass from x to y.

We select Wasserstein distance under $p = 1$ between distributions $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$ as the loss function ,i.e.

$$\mathcal{L} = W_p(\mu_{\mathcal{T}}, \mu_{\mathcal{S}}) \tag{2}$$

The KR dual of W_1 is

$$\text{KR dual} = \sup_{f \in \text{Lip}(\Omega)} \mathbb{E}_{\mu_{\mathcal{T}}}[f] - \mathbb{E}_{\mu_{\mathcal{S}}}[f] \tag{3}$$

Here $\text{Lip}(\Omega)$ is the space of all Lipschitz continue function.

It can be proved that the KR dual is a strong dual form of W_1. Therefore, minimizing W_1 is equivalent to maximizing its KR dual form. Thus the loss function can be expressed as:

$$\mathcal{L} = \sup_{f \in \text{Lip}(\Omega)} \mathbb{E}_{\mu_{\mathcal{T}}}[f] - \mathbb{E}_{\mu_{\mathcal{S}}}[f] \tag{4}$$

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

3.3 Calculating the KR dual

According to the Universal Approximation Theorem[2], a two-layer fully connected neural network is capable of approximating any continuous function almost everywhere. Thus, to solve the maximized KR dual form, we consider using a two-layer fully connected neural network to fit the Lipschitz function f in Equation 4.

Given the intractability of distributions $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$, we consider the original distribution as a weighted sum of *Dirac delta functions*, that is $\mu_{\mathcal{T}} = \sum_i \alpha_i \delta_{x_i}$, $\mu_{\mathcal{S}} = \sum_i \beta_i \delta_{y_i}$.

$\delta(x)$ is the *Dirac delta function* satisfying:

$$\begin{aligned} \int_{\mathbb{R}} \delta(x)dx = 1 \\ \delta(x) = 0, \forall x \neq 0 \end{aligned} \tag{5}$$

and delta function with position parameter $\delta_{x_0} = \delta(x - x_0)$

In the absence of prior knowledge regarding $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$, we treat $\mu_{\mathcal{T}}$ and $\mu_{\mathcal{S}}$ as the equally weighted summation of delta functions, i.e. $\mu_{\mathcal{T}} = \frac{1}{|n_{\mathcal{T}}|} \sum_i \delta_{x_i}$ and $\mu_{\mathcal{S}} = \frac{1}{|n_{\mathcal{S}}|} \sum_i \delta_{y_i}$

Here denominator $n_{\mathcal{T}/\mathcal{S}}$ is the size of each batch in training procedure. Specifically, $n_{\mathcal{T}}$ is the BatchSize for origianl dataset $\mathcal{T}$ and $n_{\mathcal{S}}$ is $\frac{1}{\text{ipc}}$, where ipc (image per class) is the number of images per class in the synthetic dataset. Based on the preceding analysis, $\mathbb{E}_{\mu_{\mathcal{T}/\mathcal{S}}}[f]$ has form:

$$\mathbb{E}_{\mu_{\mathcal{T}/\mathcal{S}}}[f] = \sum_{X \sim \mu_{\mathcal{T}/\mathcal{S}}} \mu_{\mathcal{T}/\mathcal{S}}(X) f(X) = \frac{1}{|n_{\mathcal{T}/\mathcal{S}}|} \sum_{x_i} f(x_i) \tag{6}$$

Thus, the KR dual form can be written as:

$$\mathbb{E}_{\mu_{\mathcal{T}}}[f] - \mathbb{E}_{\mu_{\mathcal{S}}}[f] = \frac{1}{|n_{\mathcal{T}}|} \sum f(y_i) - \frac{1}{|n_{\mathcal{S}}|} \sum f(x_i) \tag{7}$$

3.4 Ensuring the Lipschitz continuity of f

In Equation4, f is restricted in Lipschitz space. It is not difficult to see from Equation 3 that if the gradient of f is unbounded, the KR dual does not converge.

In this paper, we use the weight truncation method to ensure a Lipschitz f. That is, after each iteration of updating f, the parameters in f is limited to $[-b, b]$, which ensures that the Lipschitz constant is globally consistent throughout the training procedure (see lemma1).

The pseudocode of this method is shown in algorithm1

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

Algorithm 1 Main Algorithm

Input: Original dataset $\mathcal{T}$, Initialized distilled dataset $\mathcal{S}$,
randomly initialized neural network ψ,
differentiable augmentation $\mathcal{A}$,
class number C, truncation threshold b,
training epoch K_1 for updating $\mathcal{S}$, learning rate η_1 for updating $\mathcal{S}$,
training epoch K_2 for calculating KR dual, learning rate η_2 for calculating KR dual.

Output: $\mathcal{S}$

for $i = 1, 2, \cdots, K_1$ **do**
  for $c = 1, 2, \cdots, C$ **do**
    # Extracting features
    Sample mini-batch $\mathcal{B}_c^{\mathcal{T}}$ from $\mathcal{T}$ with class c
    Select $\mathcal{S}_c$ from $\mathcal{S}$ with class c
    Extracting feature $\{x_i\} = \psi(\mathcal{B}_c^{\mathcal{T}})$, $\{y_i\} = \psi(\mathcal{S}_c)$

    # Computing KR dual
    Initialize f_θ
    for $j = 1, 2, \cdots, K_2$ **do**
      Calculate $\mathbb{E}_{\mu_\mathcal{T}}[f] - \mathbb{E}_{\mu_\mathcal{S}}[f] = \frac{1}{|n_\mathcal{T}|}\sum f(y_i) - \frac{1}{|n_\mathcal{S}|}\sum f(x_i)$
      Update $\theta = \theta - (-\eta_2 \nabla_\theta(\mathbb{E}_{\mu_\mathcal{T}}[f] - \mathbb{E}_{\mu_\mathcal{S}}[f]))$ # Maximum $\mathbb{E}_{\mu_\mathcal{T}}[f] - \mathbb{E}_{\mu_\mathcal{S}}[f]$

      # Truncating parameters
      if $\theta > b$ **then**
        $\theta = b$
      end if
      if $\theta < -b$ **then**
        $\theta = -b$
      end if
    end for

    # Updating distillated dataset $\mathcal{S}$
    Calculate $\mathcal{L} = \mathbb{E}_{\mu_\mathcal{T}}[f] - \mathbb{E}_{\mu_\mathcal{S}}[f]$ with fully trained f
    Update $\mathcal{S} = \mathcal{S} - \eta_1 \nabla_\mathcal{S} \mathcal{L}$
  end for
end for

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

4 Experiments

4.1 Experiments setup

Datasets We evaluate our method on five datasets: MNIST[8], FashionMNIST[21], SVHN[12], CIFAR-10 and CIFAR-100[7]. These datasets are all designed for image classification tasks. The MNIST dataset comprises 60,000 28×28 grayscale images across 10 classes, while FashionMNIST contains 70,000 28×28 grayscale images in 10 categories. The SVHN dataset consists of 60,000 32×32 RGB images spanning 10 classes. The CIFAR-10 and CIFAR-100 benchmarks both include 50,000 32×32 RGB images, with CIFAR-10 containing 10 object categories and CIFAR-100 having 100 distinct classes.

Experiments setup We initially synthesized the distilled dataset $\mathcal{S}$ following the specifications in algorithm 1, utilizing the training set of the original dataset $\mathcal{S}$ and the hyper-parameter ipc. Subsequently, a neural network was trained on $\mathcal{S}$, evaluated on the test set of $\mathcal{T}$, and the corresponding performance metrics were reported.

For feature extraction, we employed a convolutional neural network (CNN) without its last classifier layer. To assess the effectiveness of $\mathcal{S}$, a CNN was implemented as the evaluation network.

All experimental trials were executed exclusively on NVIDIA RTX3090 GPUs, with fixed random seeds explicitly specified in the accompanying codebase to ensure full reproducibility of reported results.

Hyper-parameters Our method has three hyper-parameters: learning rate η_1 for synthesizing distillaed dataset $\mathcal{S}$, learning rate η_2 and training epoch K_2 for training potential function f in computing KR dual of Wasserstein distance. All experiments are conducted under $\eta_1 = 1.0$, $\eta_2 = 1e-3$ and $K_2 = 100$. Our empirical observations confirm that the training of function f achieves sufficient convergence under the hyperparameter configuration $\eta_2 = 1e-3$ and $K_2 = 100$, as illustrated in Figure 2. Thus we mainly focus on η_1 for this work. Across all experimental configurations, η_1 is fixed at 1.0, maintaining alignment with the parameter initialization strategy employed in vanilla Distribution Matching[24].

4.2 Comparison with other methods

We compared our approach to the four mainstream ones: Random selection from coreset selection, DC[25], DSA[23], CAFE[19] and DM[24]. As demonstrated in Table2, our approach achieves comparable result with several mainstream benchmark methods.

As demonstrated in Table 2, the distilled dataset synthesized through our methodology achieves classification accuracy comparable to mainstream baseline approaches when training image classifiers. This effectively demonstrates the feasibility of constructing a distillation dataset via calculating the Wasserstein distance by KR duality from the perspective of optimal transportation.

A comparison of the relevant data is shown in Figure3-7

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

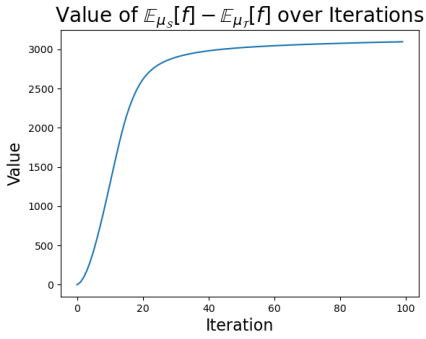

Fig. 2. The effectiveness of the neural network in converging to the solution governed by Equation 3 with $\eta_2 = 1e-3$ and $K_2 = 100$

Table 2. Comparison between our method with others'

Dataset	IPC	Random	DC	DSA	CAFE	DM	Ours	Whole
MNIST	1	$64.9\%_{\pm 3.5}$	$91.7\%_{\pm 0.5}$	$88.7\%_{\pm 0.6}$	$93.1\%_{\pm 0.3}$	$89.7\%_{\pm 0.6}$	$88.1\%_{\pm 0.6}$	$99.6\%_{\pm 0.0}$
	10	$95.1\%_{\pm 0.9}$	$97.4\%_{\pm 0.2}$	$97.8\%_{\pm 0.1}$	$97.2\%_{\pm 0.2}$	$97.5\%_{\pm 0.1}$	$96.8\%_{\pm 0.1}$	
	50	$97.9\%_{\pm 0.2}$	$98.8\%_{\pm 0.2}$	$99.2\%_{\pm 0.1}$	$98.6\%_{\pm 0.2}$	$98.6\%_{\pm 0.1}$	$98.4\%_{\pm 0.1}$	
F-MNIST	1	$51.4\%_{\pm 3.8}$	$70.5\%_{\pm 0.6}$	$70.6\%_{\pm 0.6}$	$77.1\%_{\pm 0.9}$	$70.7\%_{\pm 0.6}$	$72.2\%_{\pm 0.6}$	$93.5\%_{\pm 0.1}$
	10	$73.8\%_{\pm 0.7}$	$82.3\%_{\pm 0.4}$	$84.6\%_{\pm 0.3}$	$83.0\%_{\pm 0.4}$	$83.5\%_{\pm 0.3}$	$83.3\%_{\pm 0.3}$	
	50	$82.5\%_{\pm 0.7}$	$83.6\%_{\pm 0.4}$	$88.7\%_{\pm 0.2}$	$84.8\%_{\pm 0.4}$	$88.1\%_{\pm 0.6}$	$88.6\%_{\pm 0.1}$	
SVHN	1	$14.6\%_{\pm 1.6}$	$31.2\%_{\pm 1.4}$	$27.5\%_{\pm 1.4}$	$42.6\%_{\pm 3.3}$	$30.3\%_{\pm 0.1}$	$21.6\%_{\pm 1.1}$	$95.4\%_{\pm 0.1}$
	10	$35.1\%_{\pm 4.1}$	$76.1\%_{\pm 0.6}$	$79.2\%_{\pm 0.5}$	$75.9\%_{\pm 0.6}$	$73.5\%_{\pm 0.5}$	$72.7\%_{\pm 0.3}$	
	50	$70.9\%_{\pm 0.9}$	$82.3\%_{\pm 0.3}$	$84.4\%_{\pm 0.4}$	$81.3\%_{\pm 0.3}$	$82.0\%_{\pm 0.2}$	$80.4\%_{\pm 0.1}$	
CIFAR-10	1	$14.4\%_{\pm 2.0}$	$28.3\%_{\pm 0.5}$	$28.8\%_{\pm 0.7}$	$30.3\%_{\pm 1.1}$	$26.0\%_{\pm 0.8}$	$26.2\%_{\pm 0.6}$	$84.8\%_{\pm 0.1}$
	10	$26.0\%_{\pm 1.2}$	$44.9\%_{\pm 0.5}$	$52.1\%_{\pm 0.5}$	$46.3\%_{\pm 0.6}$	$48.9\%_{\pm 0.6}$	$48.5\%_{\pm 0.5}$	
	50	$43.4\%_{\pm 1.0}$	$53.9\%_{\pm 0.5}$	$60.6\%_{\pm 0.5}$	$55.5\%_{\pm 0.6}$	$63.0\%_{\pm 0.4}$	$60.9\%_{\pm 0.2}$	
CIFAR-100	1	$4.2\%_{\pm 0.3}$	$12.8\%_{\pm 0.3}$	$13.9\%_{\pm 0.3}$	$12.9\%_{\pm 0.3}$	$11.4\%_{\pm 0.3}$	$11.3\%_{\pm 0.2}$	$56.2\%_{\pm 0.3}$
	10	$14.6\%_{\pm 0.5}$	$25.2\%_{\pm 0.3}$	$32.3\%_{\pm 0.3}$	$27.8\%_{\pm 0.3}$	$29.7\%_{\pm 0.1}$	$29.0\%_{\pm 0.3}$	
	50	$30.0\%_{\pm 0.4}$	$53.9\%_{\pm 0.5}$	$42.8\%_{\pm 0.4}$	$37.9\%_{\pm 0.3}$	$43.6\%_{\pm 0.4}$	$40.3\%_{\pm 0.4}$	

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

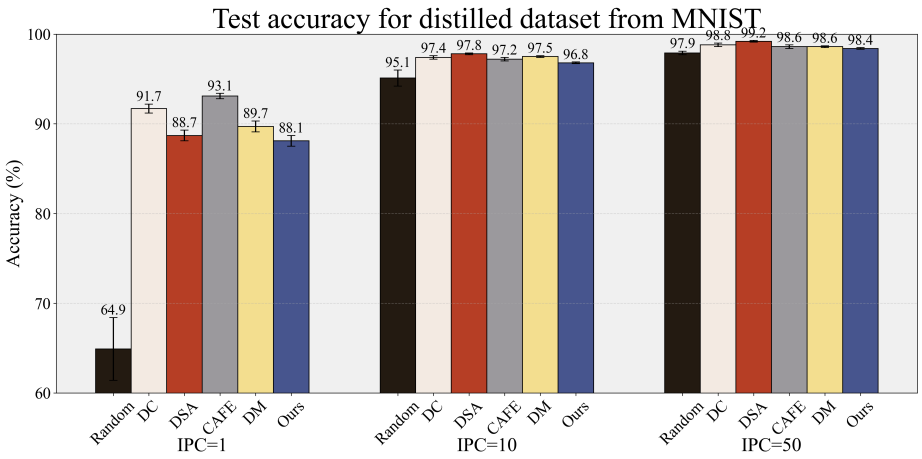

Fig. 3. Comparison between benchmarks on MNIST

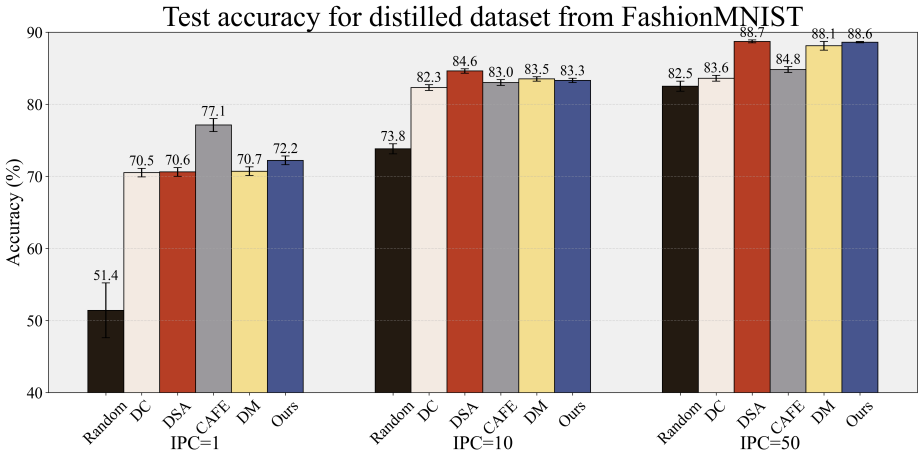

Fig. 4. Comparison between benchmarks on FashionMNIST

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

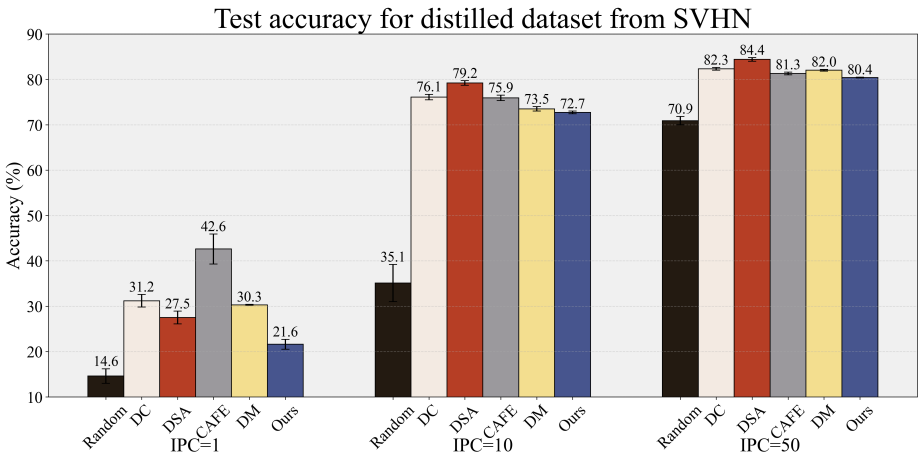

Fig. 5. Comparison between benchmarks on SVHN

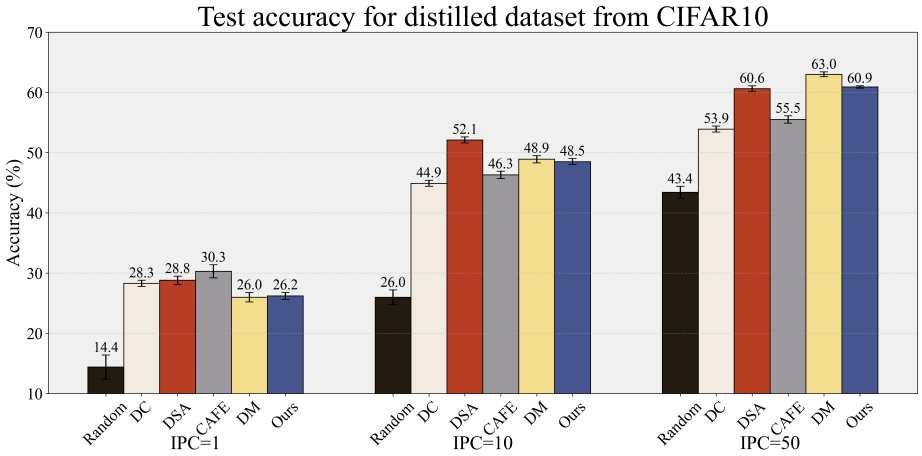

Fig. 6. Comparison between benchmarks on CIFAR-10

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

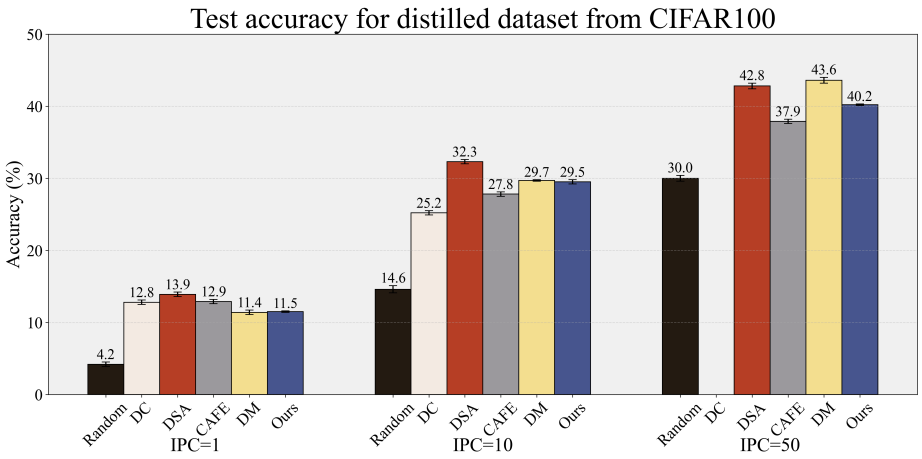

Fig. 7. Comparison between benchmarks on CIFAR-100

5 Conclusion

This paper presents a novel dataset distillation method grounded in optimal transport theory, wherein we formulate the dataset condensation process through Wasserstein metric optimization. The proposed methodology employs deep neural network and parameter truncation to approximate Lipschitz-continuous functions that compute the Wasserstein distance between source and synthesized datasets via Kantorovich-Rubinstein duality, establishing this metric as the optimization objective. Empirical validation with multiple benchmarks method confirms the efficiency of our approach.

Notwithstanding the current computational inefficiency of our neural network-based Wasserstein approximator compared to conventional linear programming-based solver[11], the parallelization capabilities of deep neural networks partially mitigate this limitation through parallel processing. Future research directions will focus on algorithmic acceleration techniques and enhanced performance refinement.

Appendix A. Proof of Lemma 1

Lemma 1 (Lipschitz constant of full-connected neural network with bounded parameters). *Let* $f(x) = W^{[2]}\sigma(W^{[1]}x + b^{[1]}) + b^{[2]}$ *be a fully-connected network with one hidden layer where:*

- $\sigma : \mathbb{R}^d \to \mathbb{R}^d$ *is an element-wise K-Lipschitz activation function*
- *Weight matrices* $W^{[s]} \in \mathbb{R}^{m_i \times n_i}$ *satisfy* $|W_{ij}^{[s]}| \leq c$ *for* $s = 1, 2$
- *Biases satisfy* $|b^{[s]}| \leq c$ *for* $s = 1, 2$

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

Then f is Lipschitz continuous with constant:

$$L \leq Kc^2\sqrt{m_1 n_1 m_2 n_2}$$

where $\mathcal{X}$ is the input domain.

Proof. Denote $z = W^{[1]}x + b^{[1]}, a = \sigma(z)$.

For two different input x_1, x_2,

$$\begin{aligned} f(x_i) &= W^{[2]}a_i + b^{[2]} \\ a_i &= \sigma(z_i) \\ z_i &= W^{[1]}x_i + b^{[1]} \end{aligned}$$

Take common 2-norm $\|\cdot\|_2$ as metric for input x and output $f(x)$. Since Frobenius norm $\|\cdot\|_F$ is compatible, we can obtain

$$\begin{aligned} \|f(x_1) - f(x_2)\|_F &= \|W^{[2]}(a_1 - a_2)\|_F && (8) \\ &\leq \|W^{[2]}\|_F \|a_1 - a_2\|_F && (9) \\ &= \|W^{[2]}\|_F \|(\sigma(z_1) - \sigma(z_2)\|_F && (10) \\ &\leq K\|W^{[2]}\|_F \|z_1 - z_2\|_F && (11) \\ &\leq K\|W^{[2]}\|_F \|W^{[1]}(x_1 - x_2)\|_F && (12) \\ &= K\|W^{[2]}\|_F \|W^{[1]}\|_F \|(x_1 - x_2)\|_F && (13) \end{aligned}$$

Thus the Lipschitz constant of f is no more than $K\|W^{[2]}\|_F\|W^{[1]}\|_F$.

Since $|W_{ij}^{[s]}|$ is bounded by b, $\|W_{ij}^{[s]}\|_F = \sqrt{\sum_{i,j} w_{ij}^2} \leq \sqrt{mnc^2} = c\sqrt{mn}$, thus the Lipschitz constant of f is given by $K\|W^{[2]}\|_F\|W^{[1]}\|_F = Kb^2\sqrt{m_1 n_1 m_2 n_2}$

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

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

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. Cybenko, G.: Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems **2**(4), 303–314 (1989)
3. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

4. Feng, Y., Vedantam, R., Kempe, J.: Embarrassingly simple dataset distillation. In: 12th International Conference on Learning Representations, ICLR 2024 (2024)
5. Geng, J., Chen, Z., Wang, Y., Woisetschlaeger, H., Schimmler, S., Mayer, R., Zhao, Z., Rong, C.: A survey on dataset distillation: Approaches, applications and future directions (2023), https://arxiv.org/abs/2305.01975
6. Guo, Chengcheng zhao, B., Bai, Y.: Deepcore: A comprehensive library for coreset selection in deep learning. In: Strauss, C., Cuzzocrea, A., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) Database and Expert Systems Applications. pp. 181–195. Springer International Publishing, Cham (2022)
7. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
8. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791
9. Lee, H., Kim, S., Lee, J., Yoo, J., Kwak, N.: Coreset selection for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 7682–7691 (June 2024)
10. Lei, S., Tao, D.: A comprehensive survey of dataset distillation. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(1), 17–32 (2024). https://doi.org/10.1109/TPAMI.2023.3322540
11. Liu, H., Li, Y., Xing, T., Dalal, V., Li, L., He, J., Wang, H.: Dataset distillation via the wasserstein metric. arXiv preprint arXiv:2311.18531 (2023)
12. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y., et al.: Reading digits in natural images with unsupervised feature learning. In: NIPS workshop on deep learning and unsupervised feature learning. vol. 2011, p. 4. Granada (2011)
13. Nguyen, T., Chen, Z., Lee, J.: Dataset meta-learning from kernel ridge-regression. In: International Conference on Learning Representations (2021)
14. Nguyen, T., Novak, R., Xiao, L., Lee, J.: Dataset distillation with infinitely wide convolutional networks. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 5186–5198. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/299a23a2291e2126b91d54f3601ec162-Paper.pdf
15. Sachdeva, N., McAuley, J.: Data distillation: A survey (2023), https://arxiv.org/abs/2301.04272
16. Saxe, A.M., Koh, P.W., Chen, Z., Bhand, M., Suresh, B., Ng, A.Y.: On random weights and unsupervised feature learning. In: Proceedings of the 28th International Conference on International Conference on Machine Learning. p. 1089–1096. ICML'11, Omnipress, Madison, WI, USA (2011)
17. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 25278–25294. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/a1859debfb3b59d094f3504d5ebb6c25-Paper-Datasets_and_Benchmarks.pdf
18. Villani, C., et al.: Optimal transport: old and new, vol. 338. Springer (2008)
19. 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)

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23

20. Wang, T., Zhu, J.Y., Torralba, A., Efros, A.A.: Dataset distillation (2020), https://arxiv.org/abs/1811.10959
21. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
22. 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
23. Zhao, B., Bilen, H.: Dataset condensation with differentiable siamese augmentation. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 12674–12685. PMLR (18–24 Jul 2021), https://proceedings.mlr.press/v139/zhao21a.html
24. 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)
25. Zhao, B., Mopuri, K.R., Bilen, H.: Dataset condensation with gradient matching (2021), https://arxiv.org/abs/2006.05929
26. 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 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97564-6_23