Dataset Distillation via the Wasserstein Metric

Haoyang Liu[1], Yijiang Li[2], Tiancheng Xing[3], Peiran Wang[4], Vibhu Dalal[5], Luwei Li[1], Jingrui He[1], Haohan Wang[1]
[1]UIUC [2]UC San Diego [3]NUS [4]UCLA [5]SAICE

Scan Here for Website:

ICCV HONOLULU HAWAII OCT 19-23, 2025

Dataset Distillation Challenge

(a) (b) KL divergence (c) MMD distance (d) Wasserstein distance

- Core Goal: **Dataset distillation** creates a small synthetic dataset that maintains the performance of models trained on the full large dataset, improving computational efficiency
- Current Limitations:
  - Bi-level optimization methods typically require expensive computation of second-order derivatives
  - Distribution matching methods using MMD does not directly capture geometric properties, leading to suboptimal performance

Advantage of Wasserstein metric

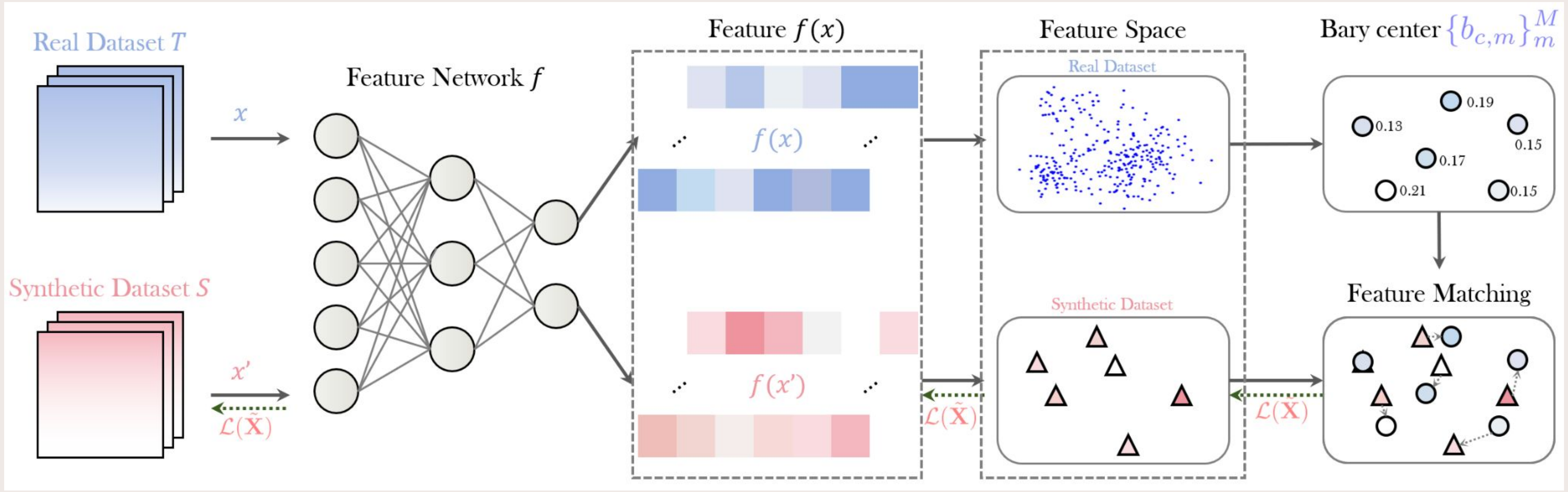
- Geometric Insight: **Wasserstein distance** measures minimal transport cost between distributions, which naturally captures distribution geometry
- Barycenter Property: Represents distribution centroid under certain constraints (e.g. sample size), preserving essential characteristics

Synthetic Data Quality

Our synthetic images of the class Hay (classId: 958)

- Representativeness: Our synthetic images capture essential class features aligned with human perception (shown in the upper row)
- Distribution Preservation: Better maintains intra-class variations from the training data distribution (shown in the lower row)

WMDD Method



- **Core idea:** First computes class-wise Wasserstein barycenter in the feature space; then learns synthetic images that match these points
- **Efficient barycenter computation:** Alternating optimization
  - Weight optimization: Solve optimal transport with fixed positions
  - Position optimization: Newton step updates using transport weights
- **Leveraging deep model prior:**
  - Use features from a pretrained classifier for high-dimensional image data
  - Propose Per-Class BatchNorm (PCBN) regularization to match BN statistics in each class separately, modeling intra-class distribution

Implementation Details

Our loss function is designed as follows:

- Match the features of the synthetic images with the corresponding data points in the learned barycenter:

$$\mathcal{L}_{\text{feature}}(\tilde{\mathbf{X}}) = \sum_{k=1}^{g}\sum_{j=1}^{m_k} \|f_e(\tilde{\mathbf{x}}_{k,j}) - \mathbf{b}_{k,j}\|_2^2$$

- Match the BN statistics of the synthetic data feature map with the real data, with synthetic samples weighted by the learned barycenter weight:

$$\mathcal{L}_{\text{BN}}(\tilde{\mathbf{X}}) = \sum_{k=1}^{g}\sum_{l=1}^{L}\Big(\|\mathcal{A}_{\text{mean}}(\{f_l(\tilde{\mathbf{x}}_{k,j})\}_{j=1}^{m_k}, \{w_{k,j}\}_{j=1}^{m_k}) - \text{BN}_{k,l}^{\text{mean}}\|_2^2 + \|\mathcal{A}_{\text{var}}(\{f_l(\tilde{\mathbf{x}}_{k,j})\}_{j=1}^{m_k}, \{w_{k,j}\}_{j=1}^{m_k}) - \text{BN}_{k,l}^{\text{var}}\|_2^2\Big)$$

- Combined loss:

$$\mathcal{L}(\tilde{\mathbf{X}}) = \mathcal{L}_{\text{feature}}(\tilde{\mathbf{X}}) + \lambda\mathcal{L}_{\text{BN}}(\tilde{\mathbf{X}})$$

Performance Results

Methods	ImageNette				Tiny ImageNet				ImageNet-1K			
	1	10	50	100	1	10	50	100	1	10	50	100
Random [60]	23.5 ± 4.8	47.7 ± 2.4	-	-	1.5 ± 0.1	6.0 ± 0.8	16.8 ± 1.8	-	0.5 ± 0.1	3.6 ± 0.1	15.3 ± 2.3	-
DM [60]	32.8 ± 0.5	58.1 ± 0.3	-	-	3.9 ± 0.2	12.9 ± 0.4	24.1 ± 0.3	-	1.5 ± 0.1	-	-	-
MTT [3]	**47.7** ± 0.9	63.0 ± 1.3	-	-	**8.8** ± 0.3	23.2 ± 0.2	28.0 ± 0.3	-	-	-	-	-
DataDAM [35]	34.7 ± 0.9	59.4 ± 0.4	-	-	8.3 ± 0.4	18.7 ± 0.3	28.7 ± 0.3	-	2.0 ± 0.1	6.3 ± 0.0	15.5 ± 0.2	-
SRe^{2}L [53]	20.6† ± 0.3	54.2† ± 0.4	80.4† ± 0.4	85.9† ± 0.2	-	-	41.1 ± 0.4	49.7 ± 0.3	-	21.3 ± 0.6	46.8 ± 0.2	52.8 ± 0.4
CDA‡ [52]	-	-	-	-	-	-	48.7	53.2	-	-	53.5	58.0
G-VBSM [36]	-	-	-	-	-	-	47.6 ± 0.3	51.0 ± 0.4	-	31.4 ± 0.5	51.8 ± 0.4	55.7 ± 0.4
SCDD [63]	-	-	-	-	-	31.6 ± 0.1	45.9 ± 0.2	-	-	32.1 ± 0.2	53.1 ± 0.1	57.9 ± 0.1
WMDD	40.2 ± 0.6	**64.8** ± 0.4	**83.5** ± 0.3	**87.1** ± 0.3	7.6 ± 0.2	**41.8** ± 0.1	**59.4** ± 0.5	**61.0** ± 0.3	**3.2** ± 0.3	**38.2** ± 0.2	**57.6** ± 0.5	**60.7** ± 0.2

- **State-of-the-Art:** Consistent improvements across all datasets
  - Large performance gains over prior art in >10 IPC settings
  - 100 IPC results *approaching full dataset performance* with the same teacher model (ResNet 18)
- **Cross-architecture generalization**
  - Performance increasing with the model capacity in ResNet family
  - Lower performance but still surpassing prior methods for ViTs

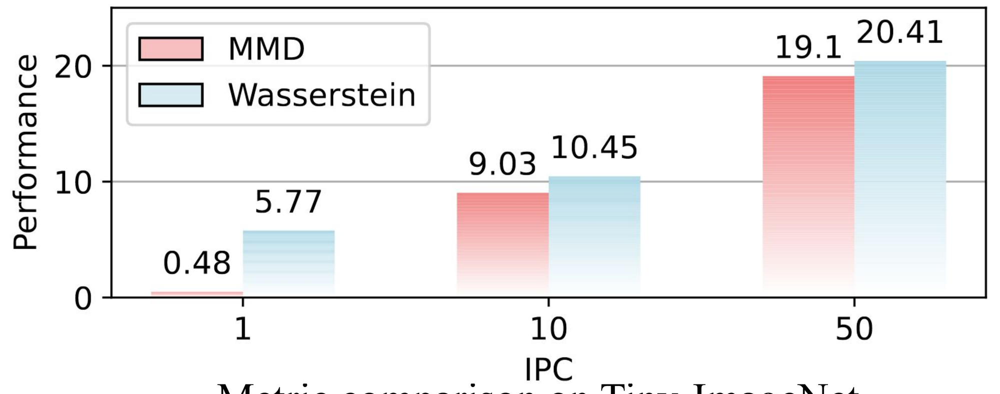
Method	Res18	Res50	Res101	ViT-T	ViT-S
SRe^{2}L	48.02	55.61	60.86	16.56	15.75
CDA	54.43	60.79	61.74	31.22	32.97
G-VBSM	52.28	59.08	59.30	30.30	30.83
WMDD (Ours)	**57.83**	**61.22**	**62.57**	**34.25**	**34.87**

Results with different evaluation models on ImageNet-1K in 50 IPC setting

Ablation Studies

$\mathcal{L}_{\text{feature}}$	$\mathcal{L}_{\text{reg}}$	ImageNette	Tiny ImageNet	ImageNet-1K
Wass.	**PCBN**	**64.7** ± 0.2	**41.8** ± 0.1	**38.1** ± 0.1
CE	PCBN	63.5 ± 0.1	41.0 ± 0.2	36.4 ± 0.2
Wass.	BN	60.7 ± 0.2	36.6 ± 0.1	26.8 ± 0.3
CE	BN	54.2 ± 0.1	38.0 ± 0.3	35.9 ± 0.2

Ablation results on loss function components in 10 IPC setting



Metric comparison on Tiny-ImageNet

- **Ablation Studies:** Combining Wasserstein metric with our PCBN regularization achieves best results across datasets; using Wasserstein metric alone leads to mixed results
- **Wasserstein vs MMD**: Significant performance advantage over MMD. This largely stems from the intractable trade-off between approximation errors and computational feasibility of the empirical MMD loss.