Video Dataset Distillation: Enhancing Diversity and Realism in Video Data for Efficient Training

Ruohong Jiang[a], Yifan Chen[a], and Haoran Wang[a]

[a]Shanghai Jiao Tong University

ABSTRACT

Training large neural networks on video datasets presents substantial computational challenges, especially when dealing with high-resolution and large-scale data. Inspired by recent advances in dataset distillation, we propose a novel approach to video data distillation, aiming to compress and optimize video datasets for efficient training while preserving critical characteristics such as realism, diversity, and efficiency. Our method demonstrates significant improvements in both accuracy and computational efficiency, achieving comparable performance to models trained on full datasets, but with drastically reduced computational cost. We conduct extensive empirical evaluations across various video datasets and model architectures, showing that our approach can effectively distill large video datasets into compact yet high-fidelity subsets, maintaining key properties crucial for real-world video applications.

Keywords: dataset distillation, video analysis, computer vision

1. INTRODUCTION

The rapid advancement of deep learning has been fueled by the availability of large-scale datasets and the development of powerful neural architectures. However, training deep models on massive video datasets still presents significant challenges due to the high computational requirements. Dataset distillation, which aims to compress large datasets into smaller, more manageable subsets while preserving performance, has emerged as a promising solution. Despite its potential, existing distillation methods often struggle with video data, particularly in terms of maintaining both high efficiency and the quality of the distilled data.

In this paper, we introduce a novel approach to video dataset distillation. Our goal is to enhance the training efficiency and effectiveness of deep learning models on video data by distilling large video datasets into compact subsets that retain the essential characteristics needed for accurate model training. We focus on two key improvements: first, the distillation process significantly reduces training time compared to using the full video dataset; second, our method outperforms existing distillation techniques in terms of both training efficiency and the quality of the distilled data. Our approach is non-optimization-based, which not only improves the scalability of the distillation process but also ensures that the distilled video data remains realistic and diverse.

Through extensive experiments, we demonstrate that our method achieves comparable performance to training on the full video dataset, but with a drastic reduction in training time.Moreover, when compared to other dataset distillation methods, our approach consistently shows superior results in both the efficiency of training and the overall performance. These findings highlight the potential of video dataset distillation as an effective tool for large-scale, high-resolution video learning tasks.

2. RELATED WORK

Dataset distillation[1] (or dataset condensation) is a technique designed to reduce the size of large datasets while retaining similar performance during training. Various approaches have been proposed, and they can be grouped into the following categories:

Further author information: (Send correspondence to Ruohong Jiang)
Ruohong Jiang: E-mail: jiangruohong@sjtu.edu.cn, Telephone: +8619946096126

2.1 Dataset Distillation

Matching Performance Metrics The first category of methods focuses on matching the performance of models trained on the distilled and original datasets. Bi-level optimization is commonly used in these approaches. Some methods, such as KIP,[2,3] leverage the Neural Tangents Kernel (NTK), while RFAD[4] uses the Empirical Neural Network Gaussian Process (NNGP) to reduce computational complexity. FRePo[5] introduces a different strategy by decomposing a neural network into a feature extractor and a linear classifier, allowing the optimization to be performed more efficiently.

Gradient and Parameter Matching Another approach aligns the gradients or parameters of the synthetic and real datasets. For example, DC[6] aligns the gradients at a single step, while DSA[7] improves this by applying symmetrical image augmentation. IDC[8] enhances the process by using lower-resolution synthetic data, and MTT[9] applies multi-step gradient matching. TESLA[10] reduces memory usage by utilizing learnable soft labels for parameter matching.

Aligning Data Distributions Methods in this category focus on aligning the distributions between real and synthetic data. DM[11] directly aligns feature distributions across both datasets, while CAFE[12] ensures that the statistical properties of the features remain consistent across all layers of the network, except for the final one.

Factorization Approaches Factorization techniques decompose the dataset into two components: a base and a hallucinator. HaBa[13] uses task-input-like bases alongside ConvNet hallucinators, while LinBa[14] integrates a linear hallucinator with provided predictands. These methods inspired video distillation techniques, where the static and dynamic elements of video data are separated to minimize temporal redundancy and reduce storage costs.

Other Methods and Extensions In addition to the above categories, there are other approaches that blend ideas from multiple areas or offer novel extensions. For instance, the approach proposed by optimizes based on training trajectory matching, showing impressive performance with low IPC. Additionally, methods like GLaD integrate generative priors to improve generalization across different distillation methods, addressing architecture transferability challenges. These approaches show promise in further reducing the computational complexity while enhancing the performance of the distilled datasets.

2.2 Video Recognition

Video recognition refers to classifying videos into predefined categories, such as human actions or scenes. Deep learning has led to several dominant approaches in this field:

One straightforward method is to treat each video frame as an individual image and process them separately using 2D convolutions. Temporal information is then aggregated through techniques like pooling, LSTMs, or GRUs to generate video-level features for classification.

To directly capture temporal dynamics, 3D convolutional networks extend 2D convolutions by adding a temporal dimension. Tran et al.[15] introduce C3D, while Carreira and Zisserman adapt pre-trained 2D models to 3D to address the high parameter count of 3D convolutions. Other methods, like R(2+1)D,[16] use a combination of spatial 2D convolutions and temporal 1D convolutions to reduce complexity while maintaining performance.

Inspired by the success of attention mechanisms in NLP, transformer-based models have been applied to video recognition to capture long-range dependencies. These models[17–19] use self-attention to effectively model temporal relationships across video frames, offering an alternative to traditional convolutional approaches.

3. LIMITS OF DATASET DISTILLATION

We begin by establishing a clear understanding of dataset distillation and subsequently outline the main challenges faced in this area.

3.1 Background

The objective of dataset condensation is to create a smaller synthetic dataset, denoted as $\mathcal{D} = (A, B) = \{\mathbf{a}_k, b_k\}_{k=1}^{|\mathcal{D}|}$, that retains the critical properties of a larger dataset $\mathcal{L} = (\tilde{A}, \tilde{B}) = \{\tilde{\mathbf{a}}_l, \tilde{b}_l\}_{l=1}^{|\mathcal{L}|}$. The condensed set $\mathcal{D}$ is generated through a transformation $\mathcal{F}$, such that $\mathcal{D} \in \mathcal{F}(\mathcal{L})$, where $|\mathcal{D}|$ is substantially smaller than $|\mathcal{L}|$ (i.e., $|\mathcal{D}|\mathbf{l}|\mathcal{L}|$). Each label $b_k \in B$ corresponds to a synthetic label for the sample $\mathbf{a}_k \in A$, and a similar association holds for $(\tilde{\mathbf{a}}_l, \tilde{b}_l) \in \mathcal{L}$.

The main goal of dataset condensation is to create $\mathcal{D}$ such that models trained on $\mathcal{D}$ perform similarly to models trained on $\mathcal{L}$, with performance deviation bounded by a small margin δ. Formally, this is expressed as:

$$\sup \left\{ \left| \mathcal{L}_{\phi_{\theta_{\mathcal{L}}}}(\mathbf{a}) - \mathcal{L}_{\phi_{\theta_{\mathcal{D}}}}(\mathbf{a}) \right| \right\}_{(\mathbf{a},b)\in\mathcal{L}} \leq \delta, \tag{1}$$

where $\theta_{\mathcal{L}}$ represents the parameters of the neural network ϕ optimized on $\mathcal{L}$:

$$\theta_{\mathcal{L}} = \arg\min_{\theta} \mathbb{E}_{(\mathbf{a},b)\in\mathcal{L}} \left[\mathcal{L}_{\phi_\theta}(\mathbf{a}, b) \right], \tag{2}$$

and a similar formulation is applied to $\theta_{\mathcal{D}}$ for the distilled dataset.

Thus, the objective is to construct $\mathcal{D}$ such that the models trained on it exhibit performance that closely approximates those trained on $\mathcal{L}$, within an error tolerance of δ.

Key Characteristics of Video Dataset Distillation. The process of dataset distillation hinges on certain defining properties that ensure the synthesized datasets are both useful and efficient. These characteristics, outlined in Definition below, are fundamental for crafting distilled datasets that not only serve as efficient training resources but also enhance the model's ability to generalize across various architectures.

Definition(Fundamental Properties). Let $\mathcal{V}$ denote a collection of observer models, including diverse types such as human models ϕ_{h} and pre-trained neural networks $\phi_{\boldsymbol{\theta}_{\mathcal{T}}}$. In this context, ϕ_{h} acts as a simplified abstraction of human predictive capacity.

The core properties of a distilled video dataset $\mathcal{S} = (X, Y)$, where X represents video samples and Y denotes corresponding labels, are listed below:

1. **Comprehensive Coverage:** To ensure robust learning and broad generalization, a distilled dataset must encompass a wide variety of samples X and labels Y. This allows the model to learn from diverse situations and contexts, thus improving its ability to perform across unseen data.

2. **Realistic Representation:** For the distilled data to facilitate cross-architecture transfer, the synthesized samples X and labels Y must reflect real-world scenarios and be applicable to a range of observer models in $\mathcal{V}$. The goal is to avoid data that is overly tailored to a specific model's characteristics, ensuring broad applicability.

3. **Scalability and Efficiency:** The practical utility of dataset distillation is also determined by its scalability. Addressing computational and memory constraints is critical, especially when adapting the distillation process to work with larger datasets. Effective distillation should not compromise performance when scaling up.

3.2 Shortcomings of Existing Dataset Distillation Methods

Traditional dataset distillation methods primarily focus on optimizing image datasets to extract representative subsets. Specifically, many approaches, such as bi-level optimization-based distillation, aim to distill high-resolution images by iteratively optimizing them. However, these methods often produce noise-like *non-realistic* patterns and suffer from overfitting to the specific model architecture used during training, which limits their ability to generalize to other architectures. Moreover, these methods suffer from a high computational burden due to the bi-level optimization structure, making them inefficient for large-scale datasets. Another approach

introduces prior regularization to mitigate architecture overfitting, thereby enhancing the realism of the synthetic images and their cross-architecture generalization. However, these methods still inherit the inefficiency of bi-level optimization-based distillation and thus struggle to scale to large datasets.

Uni-level optimization-based distillation offers a solution to some of these challenges, improving both efficiency and realism while enabling distillation of large-scale, high-resolution datasets (e.g., ImageNet). Yet, they still face a significant limitation in terms of *diversity*, as they rely on knowledge extracted from pre-trained models that contain only partial information about the original dataset. CoreSet selection methods, which aim to efficiently distill datasets by selecting a CoreSet of realistic images, also suffer from the drawback of limited data diversity, leading to catastrophic performance degradation in some cases.

While these methods show progress in image dataset distillation, they face significant challenges when applied to video datasets. Video data not only involve much larger scales but also contain temporal dependencies that complicate the distillation process. Many methods that work well for images struggle to handle the temporal dimension of video data, which requires optimization across both spatial and temporal domains. As a result, these traditional distillation methods are often inefficient and limited in their ability to handle the complexity of video datasets, necessitating more specialized approaches for video distillation.

In response to these challenges, we propose a novel method that effectively compresses video datasets while maintaining both **Diversity** and **Realism**.[20] Our approach ensures that the distillation process is capable of handling the inherent complexity of video data, and we demonstrate that it achieves promising results in terms of both accuracy and compression efficiency. This offers a more scalable and effective solution for video dataset distillation.

4. METHODOLOGY

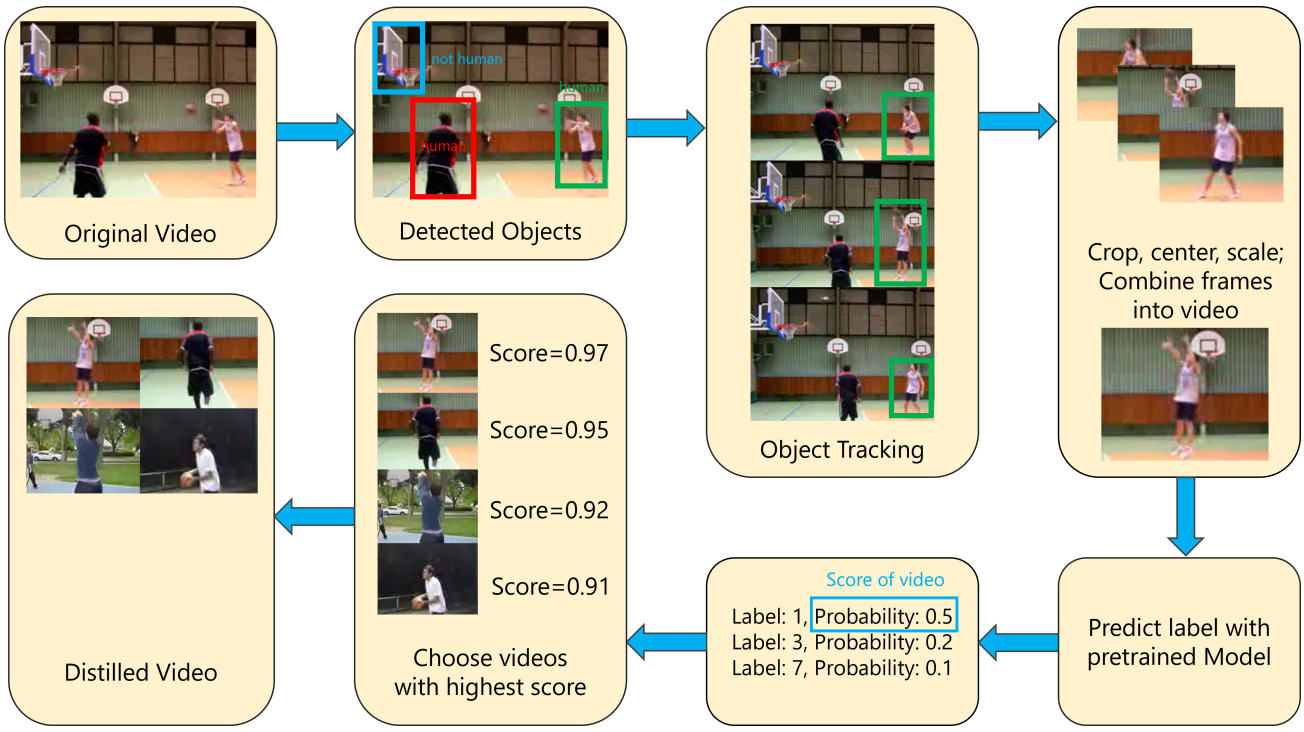

Figure 1. Caption of the image

To address the challenge of distilling large-scale, high-resolution video datasets, we propose a novel data condensation paradigm, `OptDistill`, which emphasizes both *variability* and *authenticity* while maintaining *computational efficiency*.

4.1 Fostering Variability and Authenticity in the Condensed Dataset

Inspired by key principles in artificial intelligence, specifically *minimalism* and *consistency*, our goal is to ensure that models trained on the condensed dataset adhere to these concepts. To this end, we aim to create a representation Z from the original dataset D that is both *streamlined* (*minimalism*) and rich in content (*consistency*).

This leads us to redefine the objective of dataset condensation, as described in equation (3), with a focus on both the structured simplicity and the inherent richness of the original large dataset $\mathcal{T}$:

$$\mathcal{M} = \underset{(A,Z)\in\mathcal{B}(\mathcal{T})}{\arg\max}\ \mathbb{I}_{\mathcal{U}}(A \to Z)\,, \tag{3}$$

where $\mathbb{I}_{\mathcal{U}}$ denotes the predictive $\mathcal{U}$-information from input data A to the condensed data Z. This can be elaborated as:

$$\mathbb{I}_{\mathcal{U}}(A \to Z) = \underbrace{H_{\mathcal{U}}(Z|\emptyset)}_{\text{variability}} - \underbrace{H_{\mathcal{U}}(Z|A)}_{\text{authenticity}}\,, \tag{4}$$

where $H_{\mathcal{U}}(Z|A)$ represents the conditional $\mathcal{U}$-entropy with the observed data A, and $H_{\mathcal{U}}(Z|\emptyset)$ is the entropy when no additional information is provided (i.e., no conditioning).

Understanding the Components: Variability vs Authenticity In the above formulation:

- **Variability** (captured by $H_{\mathcal{U}}(Z|\emptyset)$): This term quantifies the diversity of the distilled data Z by measuring how much uncertainty exists in Z when no side information is available. The goal is to retain the full range of possibilities present in the original dataset.
- **Authenticity** (captured by $H_{\mathcal{U}}(Z|A)$): This term quantifies the degree of realism in the distilled data Z by measuring how much the distilled set retains the true characteristics of the original dataset A. The goal here is to ensure that the condensed data is as close as possible to the true data distribution of A.

4.2 Efficient Condensation Process

The goal of `OptDistill` is to generate a smaller dataset that captures both variability and authenticity while being computationally efficient. This is achieved by optimizing the $\mathcal{U}$-information objective, where both the diversity and the fidelity of the distilled data are balanced through a systematic selection process, ultimately minimizing computational cost while maximizing the preservation of key features of the original dataset.

4.3 Extracting Key Segments from Original Dataset

To extract explicit key information from the original dataset, we focus on capturing high-realism video segments at both the pixel and sample space levels. For each video, we select the most realistic segment, denoted as $\sigma_{i,\star}$, from a set of candidate segments $\{\sigma_{i,k}\}$ extracted from the video $\hat{\mathbf{v}}_i$. The realism score $r_{i,k}$ for each segment is calculated using the loss function $r_{i,k} := -\ell(\phi_{\boldsymbol{\theta}_\mathcal{T}}(\sigma_{i,k}), y_i)$, where y_i is the human-annotated label. We then form a set $\mathcal{Q}_c$ consisting of the top-ranked segments for each class c.

To mitigate redundancy, we apply a sample space selection strategy to prune the most impactful segments based on a threshold $\bar{r}_\star$, ensuring only the highest-scoring segments are kept. This process is repeated for all classes in the dataset. Due to computational overhead and potential distribution bias from score-based selection, we propose using random uniform selection to form a pre-selected subset $\mathcal{V}_c' \subset \mathcal{V}_c$ for further processing.

4.4 Information Reconstruction of Segments

To preserve the extracted key information in the distilled dataset, we propose reconstructing video segments. For each class c, we randomly select M segments from the set $\mathcal{Q}_c'$ and concatenate them to form a final distilled video $\mathbf{v}_j$. Given the issue of single-label annotations, which may misalign with the ground truth, we use a soft labeling approach to generate region-level labels $y_{j,m}$ for each segment in the reconstructed video, which are then used for model training.

The student model $\phi_{\boldsymbol{\theta}_\mathcal{S}}$ is trained on the distilled data using the objective function:

$$\mathcal{L} = -\sum_j \sum_m y_{j,m} \log \phi_{\boldsymbol{\theta}_\mathcal{S}}(\mathbf{v}_{j,m})\,. \tag{5}$$

5. EXPERIMENT

5.1 Distillation Process

The frame rate of the output video, the number of parts the video is divided into after merging (from $N \times N$ into one), and the width and height W, H can be modified. These values can be adjusted to any desired value as needed.

1. For the first frame of the video, use YOLOv5 for object detection. Select all bounding boxes labeled with the required label, and randomly choose 3 bounding boxes from them.
2. For each selected bounding box, perform object tracking with the following steps:
   - Use the CSRT tracker from the OpenCV library.
   - Process each frame of the video individually. For each frame, the tracker will return a bounding box, say with size $w \times h$. We then use a frame size of either $w \times \frac{wH}{W}$ or $\frac{hW}{H} \times h$ (whichever fully covers the bounding box), center the bounding box, crop the area inside, and perform linear scaling to $\frac{W}{N} \times \frac{H}{N}$.
   - Process each frame separately and reassemble them to create a cropped video.
3. Convert the generated cropped video into a tensor and use a video classification model (ResNet-18-3D pre-trained on Kinetics-400) to predict its label. For each video, assign a score based on the model's confidence. In practice, the "probability of the highest confidence label" is used as the score.
4. For each label, sort all cropped videos by their score, select the top TOP_K videos, and then sort them by length. For each selection, choose $N \times N$ videos, speed up the slower videos (those with more frames) so that all videos are the same speed, and then concatenate them together.

5.2 Datasets and Metrics

In our experiments, we utilize a mix of small and large-scale video datasets. These include UCF101,[21] HMDB51,[22] Kinetics,[23] and SomethingSomething V2 (SSv2).[24] The UCF101 dataset[21] contains 13,320 video clips across 101 action categories, while HMDB51[22] consists of 6,849 video clips in 51 action classes. The Kinetics dataset[23] comprises video clips covering 400, 600, or 700 human action categories, depending on the subset, and SSv2[24] includes 174 motion-heavy categories.

To evaluate distillation algorithms at various scales, and considering both the experiment efficiency and clarity of model comparisons, we build a miniaturized version of UCF101, referred to as MiniUCF. This version includes the 50 most common classes from UCF101. Such miniaturization allows for rapid testing of our methods while also providing clearer insights into performance changes. We report the top-1 accuracy for MiniUCF and HMDB51, and top-5 accuracy for Kinetics400 and SSv2.

5.3 Baselines

The baseline methods we compare include: (1) coreset selection techniques, such as random selection, Herding,[25] and K-Center, following the implementation for image distillation in DC.[6] (2) Direct adaptation of common image distillation approaches, such as DM,[11] MTT,[9] and FRePo,[5] for the video distillation task. (3) The image distillation method (DC)[6] with frame duplication for "boring videos," which we introduce as "Static-DC."

5.4 Implementation Details

Data. For both MiniUCF and HMDB51, we sample the videos to 16 frames with a dynamic sampling interval of 4, meaning that the frame indices change across different epochs. In accordance with the setup used in C3D,[15] each of the selected frames is cropped and resized to a resolution of 112x112.

Evaluation of Distilled Dataset. To assess the quality of our synthetic data, we first evaluate how well it performs on the same architectures used during the distillation process. When using FRePo for data distillation, it is important to note that the results should be treated as a reference, as the method incorporates label learning and employs a different optimizer compared to ours. Additionally, we evaluate the performance of our synthetic data on architectures that differ from the ones used for distillation. We conduct this evaluation on the MiniUCF dataset, considering the "1 instance per class" task.

6. RESULTS

6.1 Result Table

Method	MiniUCF Accuracy		HMDB51 Accuracy	
	IPC=1	**IPC=5**	**IPC=1**	**IPC=5**
Full Dataset				
Full Dataset	57.22 ± 0.14	57.22 ± 0.14	28.58 ± 0.69	28.58 ± 0.69
Coreset Selection				
Random	9.9 ± 0.8	22.9 ± 1.1	4.6 ± 0.5	6.6 ± 0.7
Herding	12.7 ± 1.6	25.8 ± 0.3	3.8 ± 0.2	8.5 ± 0.4
K-Center	11.5 ± 0.7	23.0 ± 1.3	3.1 ± 0.1	5.2 ± 0.3
Dataset Distillation				
DM	15.3 ± 1.1	25.7 ± 0.2	8.0 ± 0.2	8.2 ± 0.1
MTT	19.0 ± 0.1	28.4 ± 0.7	8.4 ± 0.6	8.9 ± 0.6
FRePo	20.3 ± 0.5	30.2 ± 1.7	**9.6 ± 0.7**	9.6 ± 0.7
Static-DC	13.7 ± 1.1	24.7 ± 0.5	5.1 ± 0.9	7.8 ± 0.4
DM+SDD[26]	17.5 ± 0.1	27.2 ± 0.4	6.0 ± 0.4	8.2 ± 0.1
MTT+SDD[26]	**23.3 ± 0.6**	28.3 ± 0.0	6.5 ± 0.1	8.9 ± 0.6
FRePo+SDD[26]	22.0 ± 1.0	31.2 ± 0.7	8.6 ± 0.5	10.3 ± 0.6
Ours	18.2 ± 0.8	**31.0 ± 0.4**	6.6 ± 0.5	**10.5 ± 0.3**

Table 1. Accuracy comparison for MiniUCF and HMDB51 datasets with different IPC values.

6.2 Result Analysis

The proposed distillation method demonstrates notable performance both in spatial compression efficiency and training efficiency. According to the results, although the performance is relatively poor when the IPC is set to 1, it shows a strong advantage when the IPC is set to 5. This indicates that the distillation method excels when the compression efficiency requirements are less stringent, i.e., when a larger number of videos are allowed to be retained for each class. In such cases, this method significantly outperforms other compression techniques that retain an equal number of videos. Additionally, our approach requires minimal resources for object recognition and scoring during the distillation process. The original dataset is only used for model inference, unlike other methods where the original dataset is involved in the training of the distillation model, leading to lower efficiency.

7. CONCLUSION

We presented a method that excels in video analysis by balancing efficiency and accuracy. The approach performs well with flexible frame selection and minimal resource usage, achieving high-quality results. By using the dataset

Method	MiniUCF Storage		HMDB51 Storage	
	IPC=1	IPC=5	IPC=1	IPC=5
Full Dataset				
Full Dataset	9.81 GB	9.81 GB	4.93 GB	4.93 GB
Full Dataset				
Random	115 MB	586 MB	115 MB	586 MB
Herding	115 MB	586 MB	115 MB	586 MB
K-Center	115 MB	586 MB	115 MB	586 MB
Dataset Distillation				
DM[11]	115 MB	586 MB	115 MB	586 MB
MTT[9]	115 MB	586 MB	115 MB	586 MB
FRePo[5]	115 MB	586 MB	115 MB	586 MB
Static-DC	8 MB	36 MB	8 MB	36 MB
DM+SDD	94 MB	455 MB	94 MB	455 MB
MTT+SDD	94 MB	455 MB	94 MB	455 MB
FRePo+SDD	48 MB	228 MB	48 MB	228 MB
Ours	35 MB	207 MB	35 MB	207 MB

Table 2. Storage comparison for MiniUCF and HMDB51 datasets with different IPC values.

only for inference, it outperforms other methods, providing a scalable solution for real-world applications.We believe our approach will open new avenues for video distillation.

REFERENCES

[1] Wang, T., Zhu, J.-Y., Torralba, A., and Efros, A. A., "Dataset distillation," *arXiv preprint arXiv:1811.10959* (2018). Submitted on 27 Nov 2018 (v1), last revised 24 Feb 2020 (v3).

[2] Nguyen, T., Chen, Z., and Lee, J., "Dataset meta-learning from kernel ridge-regression," *arXiv preprint arXiv:2011.00050* (2020).

[3] Nguyen, T., Novak, R., Xiao, L., and Lee, J., "Dataset distillation with infinitely wide convolutional networks," in [*NeurIPS*], (2021).

[4] Loo, N., Hasani, R., Amini, A., and Rus, D., "Efficient dataset distillation using random feature approximation," in [*NeurIPS*], (2022).

[5] Zhou, Y., Nezhadarya, E., and Ba, J., "Dataset distillation using neural feature regression," *arXiv preprint arXiv:2206.00719* (2022).

[6] Zhao, B., Mopuri, K. R., and Bilen, H., "Dataset condensation with gradient matching," *arXiv preprint arXiv:2006.05929* (2020).

[7] Zhao, B. and Bilen, H., "Dataset condensation with differentiable siamese augmentation," in [*ICML*], (2021).

[8] Kim, J.-H., Kim, J., Oh, S. J., Yun, S., Song, H., Jeong, J., Ha, J.-W., and Song, H. O., "Dataset condensation via efficient synthetic-data parameterization," in [*ICML*], (2022).

[9] Cazenavette, G., Wang, T., Torralba, A., Efros, A. A., and Zhu, J.-Y., "Dataset distillation by matching training trajectories," in [*CVPR*], (2022).

[10] Cui, J., Wang, R., Si, S., and Hsieh, C.-J., "Scaling up dataset distillation to imagenet-1k with constant memory," in [*ICML*], 6565–6590 (2023).

[11] Zhao, B. and Bilen, H., "Dataset condensation with distribution matching," in [*WACV*], (2023).
[12] Wang, K., Zhao, B., Peng, X., Zhu, Z., Yang, S., Wang, S., Huang, G., Bilen, H., Wang, X., and You, Y., "Cafe: Learning to condense dataset by aligning features," in [*CVPR*], (2022).
[13] Liu, S., Wang, K., Yang, X., Ye, J., and Wang, X., "Dataset distillation via factorization," in [*NeurIPS*], (2022).
[14] Deng, Z. and Russakovsky, O., "Remember the past: Distilling datasets into addressable memories for neural networks," in [*NeurIPS*], (2022).
[15] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M., "Learning spatiotemporal features with 3d convolutional networks," in [*ICCV*], 4489–4497 (2015).
[16] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M., "A closer look at spatiotemporal convolutions for action recognition," in [*CVPR*], 6450–6459 (2018).
[17] Wang, M., Xing, J., and Liu, Y., "Actionclip: A new paradigm for video action recognition," *CoRR* **abs/2109.08472** (2021).
[18] Bertasius, G., Wang, H., and Torresani, L., "Is space-time attention all you need for video understanding?," in [*ICML*], **2**(3), 4 (2021).
[19] Tong, Z., Song, Y., Wang, J., and Wang, L., "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training," in [*NeurIPS*], Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., eds., **35**, 10078–10093, Curran Associates, Inc. (2022).
[20] Sun, P., Shi, B., Yu, D., and Lin, T., "On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm," (2024).
[21] Soomro, K., Zamir, A. R., and Shah, M., "UCF101: A dataset of 101 human actions classes from videos in the wild," *CoRR* **abs/1212.0402** (2012).
[22] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T., "Hmdb: A large video database for human motion recognition," in [*ICCV*], 2556–2563 (2011).
[23] Carreira, J. and Zisserman, A., "Quo vadis, action recognition? a new model and the kinetics dataset," in [*CVPR*], 4724–4733 (2017).
[24] Goyal, R., Kahou, S. E., Michalski, V., Materzyńska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I., and Memisevic, R., "The "something something" video database for learning and evaluating visual common sense," (2017).
[25] Welling, M., "Herding dynamical weights to learn," in [*ICML*], (2009).
[26] Wang, Z., Xu, Y., Lu, C., and Li, Y.-L., "Dancing with still images: Video distillation via static-dynamic disentanglement," (2024).