Abstract

Large-scale datasets have become the standard for achieving state-of-the-art machine learning models. However, working with such extensive datasets poses challenges in terms of storage, preprocessing, and the need for specialized equipment for training. While there is work to minimize datasets for images, text, and audio, there is little to no work for videos, making the application of dataset distillation on videos novel. Moreover, video datasets present distinct obstacles due to their greater size, requiring more extensive storage and processing capabilities when compared to image datasets. This raises the question of how to classify videos quickly without using the entire large-scale dataset. To reduce the training time and overcome these obstacles, a recent approach is dataset condensation, a technique that aims to learn a compact synthetic training set, enabling a model trained on it to achieve comparable testing accuracy to one trained on the full original training set. We extend this approach to videos such that we can synthesize videos rather than images. In this study, we employed the UCF-101 dataset along with a ResNet-18 3D video backbone architecture for developing and evaluating an algorithm that incorporates random coreset selection and distribution matching, or the matching of the embedding distributions of an original dataset and its synthetic subset, on video datasets. By randomly selecting data from the original dataset for condensation, we established an initial baseline for random coreset selection on videos that could be built upon for dataset condensation. To ensure that the model could handle video classification with other architectures, we performed experiments with the random coreset selection algorithm on the MC3-18 and ResNet-34 3D architectures. The results from the random coreset baseline and experiments demonstrated that the random coreset selection algorithm improves video classification accuracy as more data is included in the model training, and it also reduces training time when using smaller subsets.

1. Introduction

The adoption of large-scale datasets, consisting of millions of samples, has become a prevailing trend for achieving state-of-the-art machine learning models. However, working with such massive datasets brings about practical challenges, including storage, preprocessing, and the requirement for specialized infrastructure for model training. Although efforts have been made to reduce dataset sizes for images, text, and audio, there has been limited or no focus on dataset distillation for videos. Consequently, applying dataset distillation techniques to videos is a novel area of research. Additionally, video datasets present unique challenges due to their larger size, requiring more substantial storage and processing capabilities compared to image datasets.

This prompts the question of how we can efficiently classify videos without relying on the entirety of these large-scale datasets. Previous research on curriculum learning [4], active learning [6], and coreset selection [8] has offered insights into the possibility of training neural networks on subsets of the original data, leading to competitive model performance. Nonetheless, these methods often depend on heuristics, lacking guarantees of optimal solutions for downstream tasks, and the presence of representative samples is not assured either. Their efficiency is bounded by the information available in the selected samples from the original dataset.

To overcome the bottleneck of information, an alternative strategy involves generating informative samples instead of mere selection from the given dataset. Recently, a technique known as dataset condensation [11, 14] has emerged, aiming to learn a compact synthetic training set. This enables a model trained on the condensed data to achieve comparable testing accuracy with one trained on the full original training set (Fig. 1).

In the realm of dataset condensation, several methods have been proposed to optimize the distilled data and enhance condensation performance. For images, these methods include gradient matching [14], trajectory matching [1], distribution matching [13], and kernel ridge regression [7].
These advancements have resulted in improved accuracy of trained models on test sets and enhanced generalization across various network architectures. While dataset condensation was initially proposed for images, it has extended to multiple modalities, such as text, audio, and graphs. However, the video domain remains relatively unexplored in terms of dataset condensation. Video datasets pose unique challenges due to their larger size, demanding higher storage and processing capabilities compared to images. As a result, condensing video datasets becomes crucial to alleviate these resource-intensive aspects. Moreover, exploring the potential of dataset condensation for more complex tasks, including object detection and segmentation, named entity recognition, summarization, and machine translation, presents promising opportunities for future research.

In this research, we utilized the UCF-101 dataset, a compact action recognition dataset containing 13,320 videos and 101 action classes. To evaluate our novel algorithm, we employed a ResNet-18 3D architecture as the video backbone. The algorithm utilizes random coreset selection and dataset condensation techniques specifically for video datasets. Through random selection of data from the original dataset for condensation, we established some initial synthetic videos via distribution matching. To ensure the algorithm’s versatility in handling video classification with different architectures, we conducted experiments using two alternative setups. Firstly, we tested the algorithm on an architecture called MC3-18, which is a mixed convolutional layer network having the same convolutional layer depth as ResNet-18 3D. Secondly, we evaluated the algorithm on ResNet-34 3D, a similar type of architecture but with different depth. The advantages of this method are the following:

- A simple baseline algorithm to build from for future work, especially work to develop better video classification results through synthetic datasets.

The rest of the paper is structured as follows. Section II reviews other literature related to the study’s topic. Section III describes the coreset selection (e.g., random) and dataset condensation algorithms and methodological applications. Section IV presents the results from the random coreset baseline algorithm and the baseline algorithm applied to the ResNet-34 3D and MC3-18 architectures. Section V discusses improvements that could have been implemented into the study. Section VI concludes the work with future research directions.

2. Literature Review

2.1. Coreset selection

The main objective of coreset selection methods is to discover a representative subset from the original dataset (Fig. 2).

This area of research has undergone extensive theoretical analysis and has seen considerable empirical advancements over time, underscoring the need for a distinct investigation. However, as our primary focus lies in developing dataset synthesis methods for videos, we specifically highlight a frequently used selection-based baseline in the dataset condensation literature, namely random coreset selection. Random coreset selection randomly selects data from the original dataset.

Figure 2. Obtaining the coreset subset of the original dataset

In this research, we utilized the UCF-101 dataset, a compact action recognition dataset containing 13,320 videos and 101 action classes. To evaluate our novel algorithm, we employed a ResNet-18 3D architecture as the video backbone. The algorithm utilizes random coreset selection and dataset condensation techniques specifically for video datasets. Through random selection of data from the original dataset for condensation, we established some initial synthetic videos via distribution matching. To ensure the algorithm’s versatility in handling video classification with different architectures, we conducted experiments using two alternative setups. Firstly, we tested the algorithm on an architecture called MC3-18, which is a mixed convolutional layer network having the same convolutional layer depth as ResNet-18 3D. Secondly, we evaluated the algorithm on ResNet-34 3D, a similar type of architecture but with different depth. The advantages of this method are the following:

- The establishment of baselines for video condensation using coreset selection.
- The establishment of dataset condensation via distribution matching for videos.

2.2. Dataset condensation

The objective of dataset condensation methods is to create a compact set of data that can achieve competitive performance when used for training, as opposed to using the entire dataset. In the following sections, we present three prominent state-of-the-art methods, each employing a distinct technique, along with an application method for dataset condensation.
2.2.1 Gradient matching

Dataset Condensation with Gradient Matching, or DC, [14] aims to generate a synthetic dataset by aligning the optimization trajectory of a model trained on the synthetic dataset with that of the original dataset. The optimization trajectory is characterized by the gradient direction along stochastic gradient descent (SGD) steps, and the corresponding loss function is expressed by Equation 1, where $S$ represents the synthetic dataset, $T$ denotes the number of iterations, $T$ stands for the real dataset, and $\theta$ represents the model parameters.

$$\min_{\theta} \mathbb{E}_{\theta_0 \sim P(\theta_0)} \left[ \sum_{t=0}^{T-1} D \left( \nabla_{\theta} \mathcal{L}^S (\theta_t), \nabla_{\theta} \mathcal{L}^T (\theta_t) \right) \right] \quad (1)$$

The issue with this dataset condensation method is that the complex bi-level optimization between optimizing the model parameters and the synthetic dataset. To avoid the bi-level optimization and make our video synthesis faster, through distribution matching, we can match the feature distribution between original and synthetic videos using maximum mean discrepancy.

2.2.2 Differentiable siamese augmentation

Dataset Condensation with Differentiable Siamese Augmentation, or DSA, [12] proposes to apply DSA while learning synthetic image, resulting in more informative synthetic images. Similar to the loss function of DC in Equation 1, DSA applies $\mathcal{A}$ which is a family of image transformations that preserves the semantics of the input as shown in Equation 2.

$$\min_{S} D \left( \nabla_{\theta} \mathcal{L} \left( A (S, \omega^S), \theta_i \right), \nabla_{\theta} \mathcal{L} \left( A (T, \omega^T), \theta_i \right) \right) \quad (2)$$

2.2.3 Distribution matching

Dataset Condensation with Distribution Matching, or DM, [13] states that, unlike DC and DSA, DM learns condensed dataset by directly matching the output features between real and synthetic samples. With further development, the features are to be acquired from a model with randomized weights, which would correspond to data distribution in a randomly projected embedding space (Fig. 3). The objective function is shown in Equation 3 where $\psi_i$ is a family of parametric functions to map the input into a lower dimensional space and $\omega \sim \Omega$ is the augmentation parameter.

$$\min_{S} E_{\psi \sim P(\psi)} \left\| \frac{1}{|T|} \sum_{t=1}^{T} \psi_t \left( A (x_t, \omega) \right) - \frac{1}{|S|} \sum_{i=1}^{|S|} \psi_t \left( A (x_i, \omega) \right) \right\|^2 \quad (3)$$

2.2.4 Video backbone

In contrast to images, effectively capturing the spatiotemporal relationships among video frames requires a 3D backbone, such as ResNet 3D [2]. To accelerate video synthesis and avoid gradient matching’s expensive bi-level optimization, our approach for dataset condensation via synthesis involves matching feature distributions between original and synthetic videos using maximum mean discrepancy. Moreover, we can to improve performance by incorporating differentiable augmentation [12].

3. Approach

In this study, we made use of the UCF-101 dataset. To assess the effectiveness of our random coreset selection algorithm, we employed a ResNet-18 3D architecture as the underlying video backbone. The algorithm combines random coreset selection and dataset condensation methods tailored for video datasets. By randomly selecting data from the original dataset for condensation, we created an initial set of synthetic videos through distribution matching (Fig. 4). To perform the distribution matching, the synthetic data can be initialized with either random noise or with random video samples from the input video dataset. We then extract the embeddings of a batch of real and synthetic samples, and we calculate the loss or discrepancy between the distribution embeddings of them. With the loss calculated, we update the synthetic data by backpropagating through the model and differentiable augmentation operations (e.g., random horizontal flip, random vertical flip) to minimize the distribution embedding discrepancy between the real and synthetic samples. Through iterations, we repeat this process for fixed number of epochs. As a result, the algorithm yields optimized synthetic videos that can then be solely used to quickly train the video 3D models.

As an alternative to the video backbone, by converting video data into image data, we can gain the advantage of directly utilizing image-based data distillation techniques. Additionally, processing data with a 2D backbone is more straightforward and cost-effective compared to complex network backbones. A recent study [3] has approached video recognition as an image recognition task, which has inspired us to rearrange video frames into super images (as shown in Fig. 5).

This rearrangement enables us to treat the video distillation problem as a distillation of super images. This novel approach can serve as a robust baseline, allowing for the development of more sophisticated methods in the future. However, the limitation of the 2D backbone lies in its inability to capture temporal relationships crucial for videos, which can be addressed by employing a 3D backbone.
3.1. Architectures

In regard to the video backbone architectures utilized in this project, the small architecture, ResNet-18 3D, is being used to create the initial baseline algorithm for random coreset selection on videos while ResNet-34 3D and MC3-18 are architectures that are applied to the baseline to provide insight on how it fares with other architectures and architecture depths. These architectures are all prominent convolutional neural network (CNN) models tailored for video analysis tasks. ResNet-18 3D is an extension of the ResNet architecture to handle spatiotemporal information in videos. It utilizes 3D convolutions to capture both spatial and temporal features across video frames, making it well-suited for action recognition and other video-related tasks. ResNet-34 3D is a deeper version of the ResNet-18 3D model, incorporating more layers for increased representational power. This additional depth allows ResNet-34 3D to capture more complex temporal patterns in videos, which can be advantageous in scenarios with challenging or intricate actions. On the other hand, MC3-18 is a mixed convolutional layer network with the same depth as ResNet-18 3D. It employs a combination of 2D and 3D convolutions to strike a balance between computational efficiency and modeling spatiotemporal relationships. This makes MC3-18 a compelling choice for video analysis tasks where computational resources may be limited, while still achieving competitive performance in action recognition and related applications.
3.2. Augmentations

The current baseline algorithm for random coreset selection has been used in its original form without any augmentations, leading to its initial performance level. In contrast, the development of the distribution matching algorithm incorporates the use of Differentiable Siamese Augmentation (DSA) [13], which involves various augmentations like random horizontal flip, random vertical flip, random crop, and random rotation. Presently, the baseline algorithm for distribution matching has integrated random horizontal and vertical flips as part of the augmentation process.

4. Experiments and Results

4.1. Experiment designs

For the baseline algorithm, we randomly select 10, 15, 20, and 25% of the videos from the original UCF-101 dataset and separately train the model on the subsets to get the video classification accuracies. To ensure the algorithm’s adaptability for video classification across various architectures, we conducted experiments under two alternative setups. Firstly, we tested the algorithm using an architecture named MC3-18 [10], which is a ResNet with mixed convolutions with the same convolutional layer depth as ResNet-18 3D [2]. Secondly, we evaluated the algorithm on ResNet-34 3D [5], which is a similar architecture but with more convolution layers or, in other words, a deeper network.

4.2. Dataset

In more detail, the UCF-101 dataset [9] is a lightweight widely used benchmark dataset in the field of action recognition. It comprises a diverse collection of 13,320 videos, spanning over 101 action categories (Fig. 6). Each video showcases a specific human action, such as playing instruments, sports activities, or everyday actions. The dataset’s videos vary in length, resolution, and quality, providing a challenging and realistic representation of real-world scenarios. The UCF-101 dataset has played a crucial role in advancing action recognition research and has been used to evaluate the performance of various machine learning and deep learning models. Researchers utilize this dataset to develop and test algorithms that aim to accurately identify and classify actions within videos, contributing to advancements in areas like video understanding.

4.3. Training Details

For comparable results from the random coreset selection’s baseline algorithm and the experiments on the different architectures (i.e., MC3-18 and ResNet-34 3D), the same hyperparameters were utilized. The models that trained on the architectures underwent 10 epochs, a batch size of 64, an initial learning rate of 0.01 with a learning rate scheduler, Stochastic Gradient Descent as the optimizer, CrossEntropy as the loss function, ReLu as the activation function, a momentum of 0.9, and a weight decay of 1.00e − 04.
Figure 6. The UCF-101 dataset is a collection of 13,320 videos over 101 action categories that consists of variations in camera motion, object appearance and pose, object scale, etc. to provide a realistic action recognition dataset.

4.4. Results

The video classification results for random coreset selection on the baseline architecture, ResNet-18 3D (Table 1) and the experimental architectures, ResNet-34 3D (Table 2) and MC3-18 (Table 3), are presented as top 1 and top 5 accuracy scores. Top 1 accuracy score represents the percentage of correct predictions where the model’s top prediction matches the ground truth label, while top 5 accuracy score measures the percentage of correct predictions when the ground truth label is among the model’s top five predicted labels. Results for the time (i.e., in the $HH:MM:SS$ format) in which each video classification accuracy was computed is also included.

<table>
<thead>
<tr>
<th>Sample Percentage</th>
<th>Top 1 Accuracy</th>
<th>Top 5 Accuracy</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>24.757</td>
<td>48.389</td>
<td>1:41:02</td>
</tr>
<tr>
<td>15%</td>
<td>35.141</td>
<td>59.488</td>
<td>1:56:29</td>
</tr>
<tr>
<td>20%</td>
<td>46.496</td>
<td>70.23</td>
<td>2:05:51</td>
</tr>
<tr>
<td>25%</td>
<td>64.297</td>
<td>82.762</td>
<td>2:18:49</td>
</tr>
<tr>
<td>100%</td>
<td>81.228</td>
<td>97.903</td>
<td>9:24:57</td>
</tr>
</tbody>
</table>

Table 1. Results for random coreset selection on the ResNet-18 3D architecture.

5. Discussion

Random coreset selection, while a common method in dataset condensation, does have its limitations. Since it ran-
Firstly, as stated in the discussion, we will finalize the explorations in this novel problem domain. Altogether, these results are promising for future advantage, making the model training less temporally intensive in computational run time offers a promising architectural benefits. These outcomes align with the necessity of further developing the distribution matching component of our algorithm. These objectives collectively propel the advancement of video synthesis and action recognition capabilities.

### 6. Conclusion and Future Work

The findings derived from the random coreset baseline and experimental analysis illustrate that the novel algorithm leads to enhanced video classification accuracy with the inclusion of more data during model training. Additionally, the algorithm demonstrates reduced training time when using smaller subsets, indicating its potential to alleviate computational expenses. These outcomes align with the intended functioning of random coreset selection. The reduction in computational run time offers a promising advantage, making the model training less temporally intensive. Altogether, these results are promising for future explorations in this novel problem domain.

Our future objectives comprise three essential aspects. Firstly, as stated in the discussion, we will finalize the distribution matching component of our algorithm, enabling the synthesis of informative samples by aligning feature distributions between original and synthetic videos. Secondly, we aim to expand our experimentation to include the HMDB51 dataset, allowing us to evaluate the algorithm’s performance on a diverse set of videos. For this, ablation experiments would have to be completed to find the most suitable hyperparameters to provide high video classification accuracy for the HMDB51 dataset. Lastly, we plan to explore the application of distribution matching on videos utilizing image-based 2D backbones, particularly through the utilization of superimages. This approach will convert video condensation into superimage condensation, offering the potential to exploit the benefits of image-based techniques and further enhance the efficiency and efficacy of our novel algorithm. These objectives collectively propel the advancement of video synthesis and action recognition capabilities.

### References