BLG454E Term Project - Dataset Distillation: Efficient Training with Synthetic Datasets via Trajectory Matching

Ömer Faruk San
Student ID: 150220307
san22@itu.edu.tr

Mustafa Kerem Bulut
Student ID: 150220303
bulutm22@itu.edu.tr

***Abstract*—Training modern deep neural networks typically requires large-scale datasets, incurring significant computational and memory costs. Dataset distillation aims to alleviate this by synthesizing a small, highly informative set of synthetic training samples that can substitute for the full dataset while preserving model performance. This project implements a dataset distillation technique based on trajectory matching, where synthetic data is optimized such that a model trained on it follows a similar parameter trajectory to a model trained on the original, larger dataset. We utilize the CIFAR-10 dataset as our benchmark. A Convolutional Neural Network (ConvNet) trained on the full CIFAR-10 achieved a test accuracy of 83.31%, serving as our teacher model. We then distilled a synthetic dataset of 100 images (10 per class). A new ConvNet, trained from scratch solely on these 100 distilled images, achieved a test accuracy of 10.28% on the CIFAR-10 test set. While the distilled dataset enabled perfect memorization during its training, its generalization performance was limited with the current implementation, highlighting the challenges in effectively compressing large dataset knowledge into a very small synthetic set.**

***Index Terms*—Dataset Distillation, Efficient Training, Synthetic Data, Trajectory Matching, Deep Learning, CIFAR-10.**

I. Introduction

Deep learning models have demonstrated remarkable success across various domains, but their performance heavily relies on the availability of large-scale, high-quality datasets. The process of collecting, storing, and training on such extensive datasets can be computationally expensive, time-consuming, and memory-intensive, posing significant challenges in resource-constrained environments or applications requiring rapid model deployment.

Dataset Distillation (DD) [1] emerges as a promising approach to address these challenges. The core idea is to synthesize a small set of synthetic data points that encapsulate the essential knowledge of a much larger dataset. Training a model on this compact, distilled set should ideally yield performance comparable to training on the original full dataset, but with substantially reduced computational overhead.

Several strategies for dataset distillation have been proposed, including matching gradients [1], matching expert trajectories [2], and matching training trajectories [3]. Our project focuses on the latter approach, specifically ”Dataset Distillation by Matching Training Trajectories” as proposed by Cazenavette et al. [3]. This method optimizes synthetic images such that a student model, trained on these synthetic images, mimics the learning trajectory (sequence of model parameters) of a teacher model trained on the full real dataset.

Our project focuses on the latter approach, specifically ”Dataset Distillation by Matching Training Trajectories” as proposed by Cazenavette et al. [3]. This method optimizes synthetic images such that a student model, trained on these synthetic images, mimics the learning trajectory (sequence of model parameters) of a teacher model trained on the full real dataset. This work was a team effort: Ömer Faruk San handled the literature survey and data preparation, while Mustafa Kerem Bulut led the implementation of the distillation algorithm and model training. Both team members contributed to the analysis and reporting of results.

In this project, we implement a simplified version of this trajectory matching technique. We use the CIFAR-10 dataset as our benchmark. Our primary goal is to generate a small synthetic dataset (100 images) and evaluate how well a standard Convolutional Neural Network (ConvNet) trained on this distilled set performs compared to one trained on the full CIFAR-10 dataset. We hypothesize that even with a significant reduction in data size, a competitive level of accuracy can be achieved, thereby demonstrating the potential for efficient training.

II. Problem Statement and Hypothesis

A. Problem Statement

The primary problem addressed is the high cost associated with training deep neural networks on large datasets. This project aims to implement and evaluate a dataset distillation technique to create a small synthetic dataset that can efficiently train models for image classification tasks, specifically on CIFAR-10.

B. Hypothesis

We hypothesized that a model trained on a small distilled dataset (e.g., 100 synthetic images for CIFAR-10, representing 0.2% of the original 50,000 training images) could achieve a significant fraction (e.g., aiming for 50-70% or higher relative performance) of the test accuracy of a model trained on the full dataset. This would demonstrate a substantial reduction in data requirements while retaining a usable level of model performance. Our initial more optimistic hypothesis from the proposal (80-90% of full performance) serves as an ideal target.

C. Literature Survey

Dataset distillation has gained considerable attention. Wang et al. [1] introduced the concept by proposing to match gradients generated by synthetic data to those from real data. Zhao et al. [2] proposed Dataset Condensation with Gradient Matching, optimizing synthetic data so that training a model for a few steps on synthetic data yields similar performance as training on real data. Cazenavette et al. [3] improved upon these by matching longer training trajectories, suggesting that this captures more information about the learning dynamics. Their method involves an outer loop optimizing the synthetic data and an inner loop training a student model on this synthetic data, with the meta-objective being the alignment of student and teacher trajectories. Other related areas include coreset selection, knowledge distillation [4], and generative models, though dataset distillation focuses on synthesizing the data itself rather than compressing a model or learning a generative distribution. Applications of dataset distillation are diverse, including model compression, continual learning, federated learning, and privacy-preserving machine learning.

III. Methodology

Our approach involves several key steps: (1) training a teacher model on the full dataset and saving its parameter trajectory, (2) initializing a small synthetic dataset, (3) iteratively optimizing this synthetic dataset by matching student model trajectories (trained on synthetic data) to the teacher's trajectory, and (4) evaluating the quality of the distilled dataset.

A. Data

The CIFAR-10 dataset [5] was used, consisting of 50,000 32x32 RGB training images and 10,000 test images, categorized into 10 classes. Standard preprocessing involved normalization. For teacher training, random crops and horizontal flips were applied as data augmentation. The goal was to distill this into a synthetic dataset of 100 images, with 10 images per class (IPC=10).

B. Teacher Model Training and Trajectory Saving

A custom Convolutional Neural Network (ConvNet) architecture (detailed below) was chosen as both the teacher and student model.

- **Architecture**: A ConvNet with 4 convolutional layers (32, 64, 128, 128 filters, 3x3 kernels, BatchNorm, ReLU), each pair followed by a MaxPool layer, and two fully connected layers (512 units with Dropout, then 10 output units).
- **Training**: The teacher model was trained on the full CIFAR-10 training set for 15 epochs using SGD with a learning rate of 0.01, momentum of 0.9, weight decay of 5e-4, and a cosine annealing learning rate scheduler.
- **Trajectory**: The state dictionary (parameters) of the teacher model was saved at the end of each of the 15 epochs, forming a trajectory of 15 parameter snapshots.

C. Synthetic Dataset Initialization

The synthetic dataset X_{syn} (100 images, 3x32x32) was initialized by selecting the first 10 images from each class of the original CIFAR-10 training set (after normalization but without random augmentation). The corresponding labels Y_{syn} were fixed. X_{syn} was set to require gradients for optimization.

D. Dataset Distillation via Trajectory Matching

The core distillation process followed an iterative, bi-level optimization:

1) **Outer Loop (Optimizing X_{syn})**: This loop ran for 1000 ”distillation epochs”. The Adam optimizer (lr=0.1, betas=(0.5, 0.999)) was used to update the pixel values of X_{syn}.
2) **Inner Loop (Student Training on X_{syn})**:
   a) A new student ConvNet (same architecture as teacher) was initialized.
   b) Its parameters were set to a state $P_{teacher,t}$ from the saved teacher trajectory (cycling through the 15 saved states).
   c) The student model was then trained for $K = 10$ inner steps on the current X_{syn} and Y_{syn} using SGD (lr=0.01, momentum=0.5). This results in updated student parameters $P_{student,t+K}$.
3) **Meta-Loss Calculation**: The meta-loss (or trajectory matching loss) was defined as the L2 distance between the student's parameters after K inner steps, $P_{student,t+K}$, and the teacher's next trajectory point, $P_{teacher,t+1}$:

$$\mathcal{L}_{meta} = \sum_j ||P^{(j)}_{student,t+K} - P^{(j)}_{teacher,t+1}||_2^2 \quad (1)$$

   where j indexes over the parameter tensors in the model.
4) **Updating X_{syn}**: The gradient of $\mathcal{L}_{meta}$ with respect to X_{syn} was computed via backpropagation through the inner loop operations, and X_{syn} was updated by its optimizer.

E. Evaluation of Distilled Dataset

After 1000 distillation epochs, the final X_{syn} and Y_{syn} were saved. To evaluate their quality:

1) A new ConvNet (same architecture) was initialized with random weights.
2) This model was trained for 200 epochs solely on the 100 distilled images using SGD (lr=0.01, momentum=0.9, weight decay=5e-4, MultiStepLR scheduler at [100, 150] epochs with gamma=0.1). Batch size for this training was 100 (full batch).
3) The performance of this model was evaluated on the standard CIFAR-10 test set.

IV. Results

A. Teacher Model Performance

The teacher ConvNet, trained on the full 50,000 CIFAR-10 training images for 15 epochs, achieved a final test accuracy of **83.31%**. The training progression is summarized in Table I.

TABLE I
Teacher Model Training Summary (Selected Epochs)

Epoch	Train Acc.	Test Acc.
1	40.53%	47.68%
5	67.57%	73.26%
10	77.18%	80.83%
15	81.14%	83.31%

B. Distilled Dataset

The distillation process ran for 1000 outer epochs. The meta-loss showed a cyclical pattern, corresponding to the 15 points in the teacher trajectory being matched. The final meta-loss at epoch 1000 was 1.0709. The resulting 100 synthetic images are shown in Fig. 1. They appear as abstract, class-specific patterns.

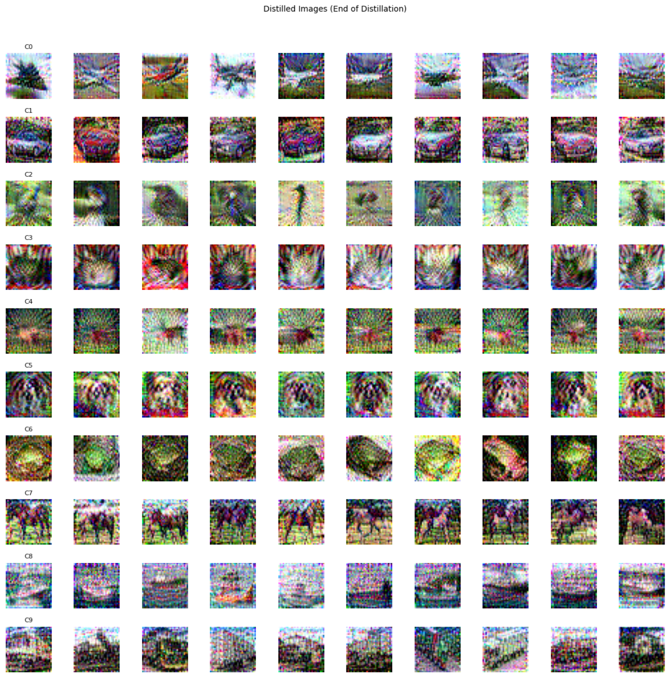

Fig. 1. The 100 distilled synthetic images (10 per class) after 1000 distillation epochs. Images are unnormalized for visualization.

C. Performance on Distilled Dataset

A new ConvNet, trained from scratch for 200 epochs using only the 100 distilled images, achieved:

- **Training Accuracy (on distilled set)**: 100.00% (indicating perfect memorization of the small synthetic set).
- **Test Accuracy (on CIFAR-10 test set)**: **10.28%**.

The training progression of the model on the distilled set is shown in Table II.

TABLE II
Evaluation Model Training on Distilled Data (Test Accuracy on CIFAR-10)

Eval Epoch	Train Acc (Distilled)	Test Acc (CIFAR-10)
20	100.00%	10.24%
100	100.00%	10.30%
200	100.00%	10.28%

V. Discussion

The primary goal of this project was to implement dataset distillation via trajectory matching and evaluate its effectiveness. Our teacher model achieved a respectable 83.31% accuracy on CIFAR-10. The implemented distillation process successfully generated 100 synthetic images.

However, a new model trained solely on these 100 distilled images achieved only 10.28% test accuracy on CIFAR-10. This performance is slightly above random guessing (10

Several factors could contribute to this limited generalization:

1) **Simplified Trajectory Matching**: The implemented method matches the final state of a short student training sequence to a single future teacher state. More sophisticated methods in [3] involve matching gradients or parameter differences over multiple steps within the student's inner training loop, which might capture finer-grained learning dynamics.
2) **Teacher Trajectory Granularity**: The teacher trajectory was saved only at the end of each epoch. This means the parameter "jumps" between consecutive trajectory points ($P_{teacher,t}$ and $P_{teacher,t+1}$) are large, representing many gradient updates on the real data. The student, trained for only 10 inner steps on X_{syn}, might struggle to meaningfully traverse such a large distance in parameter space to match the target. A finer-grained teacher trajectory (e.g., saved every N batches) might be beneficial.
3) **Number of Distillation Epochs**: While 1000 distillation epochs were run, the cyclical nature of the meta-loss (due to 15 teacher trajectory points) means each specific start-target pair was only optimized for approximately $1000/15 \approx 66$ times. More advanced distillation can run for tens of thousands of epochs.
4) **Hyperparameter Sensitivity**: The learning rates for both the synthetic data optimizer (lr_{syn}) and the inner-loop student optimizer ($\text{lr}_{student\_inner}$), as well as the number of inner student steps (K), are critical and likely require more extensive tuning.
5) **Model Capacity and Architecture**: While the same ConvNet was used for teacher and student, dataset distillation results are known to be sensitive to model architecture. It's possible that a different architecture, or

one with properties like those in ResNets, might be more amenable to this distillation technique.

The generated synthetic images (Fig. 1) resemble abstract patterns rather than clear, recognizable objects, which is a common observation in dataset distillation literature [1], [2]. This suggests they are highly optimized features for the specific network and training process.

VI. Conclusion and Future Work

This project successfully implemented a dataset distillation pipeline based on a simplified trajectory matching approach for the CIFAR-10 dataset. We demonstrated the process of training a teacher model, saving its trajectory, and optimizing a small set of synthetic images to mimic parts of this trajectory. While the resulting 100 distilled images led to perfect memorization on themselves, they yielded a test accuracy of 10.28% when used to train a new model from scratch. This highlights the significant challenge in compressing the rich information of a large dataset into an extremely small synthetic counterpart that still promotes good generalization.

Future work could explore several avenues for improvement:

- Implementing a more sophisticated trajectory matching loss, such as matching gradients at each inner student step or minimizing the difference in parameter updates over the inner loop.
- Using a finer-grained teacher trajectory, saved more frequently during teacher training.
- Conducting more extensive hyperparameter tuning for the distillation process (e.g., lr_{syn}, K, $\mathrm{lr}_{student\_inner}$).
- Increasing the number of distillation epochs significantly.
- Experimenting with different model architectures (e.g., ResNet18) for both teacher and student.
- Exploring different initialization strategies for the synthetic data.
- Increasing the number of images per class (IPC) in the distilled set, as performance typically scales with IPC.

Despite the current generalization gap, this project provides a practical implementation and valuable insights into the mechanics and challenges of dataset distillation.

References

[1] T. Wang, J.-Y. Zhu, A. Torralba, and A. A. Efros, "Dataset distillation," in *arXiv preprint arXiv:1811.10959*, 2018.

[2] B. Zhao, K. R. Mopuri, and H. Bilen, "Dataset condensation with gradient matching," in *International Conference on Learning Representations (ICLR)*, 2021. [Online]. Available: https://openreview.net/forum?id=kGUNI5N42V

[3] G. Cazenavette, T. Wang, A. Torralba, A. A. Efros, and J.-Y. Zhu, "Dataset distillation by matching training trajectories," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 4750–4759.

[4] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531 (NIPS Deep Learning Workshop)*, 2015.

[5] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009, technical Report.