

Dataset Distillation via Vision-Language Category Prototype

Yawen Zou¹, Guang Li², Duo Su³, Zi Wang⁴, Jun Yu⁴, Chao Zhang¹

¹University of Toyama, ²Hokkaido University, ³Tsinghua University, ⁴Niigata University

d24c8004@ems.u-toyama.ac.jp, guang@lmd.ist.hokudai.ac.jp, suduo@mail.tsinghua.edu.cn

f241501h@mail.cc.niigata-u.ac.jp, yujun@ie.niigata-u.ac.jp, zhang@eng.u-toyama.ac.jp

Abstract

Dataset distillation (DD) condenses large datasets into compact yet informative substitutes, preserving performance comparable to the original dataset while reducing storage, transmission costs, and computational consumption. However, previous DD methods mainly focus on distilling information from images, often overlooking the semantic information inherent in the data. The disregard for context hinders the model’s generalization ability, particularly in tasks involving complex datasets, which may result in illogical outputs or the omission of critical objects. In this study, we integrate vision-language methods into DD by introducing text prototypes to distill language information and collaboratively synthesize data with image prototypes, thereby enhancing dataset distillation performance. Notably, the text prototypes utilized in this study are derived from descriptive text information generated by an open-source vision-language model. This framework demonstrates broad applicability across datasets without pre-existing text descriptions, expanding the potential of dataset distillation beyond traditional image-based approaches. Compared to other methods, the proposed approach generates logically coherent images containing target objects, achieving state-of-the-art validation performance and demonstrating robust generalization. Source code and generated data are available in <https://github.com/zou-yawen/Dataset-Distillation-via-Vision-Language-Category-Prototype/>.

1. Introduction

The rapid development of deep learning, driven by powerful computational resources and extensive datasets, has posed substantial challenges for researchers due to the increasing demands for computational power and storage capacity [5, 14, 22]. Confronted with these issues, dataset distillation (DD) has emerged as a promising approach to extracting and distilling information from large datasets. DD synthesizes smaller datasets with high information density



Figure 1. Visualization results of SRe²L, GR (w/o text), and GR (w/ text). GR (w/o or w/ text) denotes the generative model outputs without and with text descriptions. Notably, GR (w/ text) captures rich details of target objects while preserving background diversity, leading to more comprehensive and visually coherent images.

that approximate the performance of downstream tasks conducted on the original dataset [8, 12, 30, 39]. Moreover, the surrogate dataset helps alleviate concerns about privacy and copyright issues [2, 7, 16].

Dataset distillation was first introduced by Wang et al. [39] and the follow-up studies have made significant advancements in recent years [9, 17, 32, 44, 46, 48]. The earliest methods primarily include meta-learning based and matching based methods. In meta-learning methods, the distilled data are optimized as hyperparameters within a bi-level optimization framework [6, 24, 25, 27, 33]. And the matching based methods distill images via parameter matching [3, 12, 15, 23, 43] and distribution matching [21, 37, 45–47].

However, these methods are compute-intensive and require substantial runtime due to iterative optimization, which ensures their representativeness. Recently, several studies utilize generative models [9, 26, 32, 38] for DD to

optimize latent features rather than image pixels, achieving faster training and improved performance. Gu et al. [9] propose extra minimax criteria for diffusion models to generate representative and diverse synthetic data. Su et al. [32] integrate the diffusion model into DD to extract embedding features, employing clustering centers as class prototypes, which are subsequently combined with label texts to generate images. Compared to traditional methods, generative models exhibit consistent GPU consumption across various images per class (IPC) settings while significantly reducing computational costs and demonstrating superior performance.

Despite the considerable progress in applying diffusion models to DD, these methods still face significant challenges. As illustrated in Fig. 1, existing methods (GR (w/o text)) occasionally generate images that consist only of background features without the target objects. Additionally, they struggle to generate logically coherent images, often producing unrealistic outputs such as dogs with five legs. Another critical issue is the co-occurrence bias inherent in the dataset, where certain objects or features frequently appear together. For example, fish and green plants often co-occur in surrogate data. This bias causes models to overemphasize these co-occurring features, prioritizing their coexistence over the accurate representation of individual elements. These challenges arise because existing methods focus solely on distilling information from image features while neglecting semantic information. As a result, they lack the necessary contextual understanding to generate coherent images, ultimately compromising the quality of the synthesized data.

In this work, we propose a novel framework that integrates vision-language methods into DD to improve the performance of the distilled datasets. Unlike traditional approaches that rely solely on visual features, our method leverages paired image-text representations to guide the generative process, enabling the generation of logically coherent and semantically enriched datasets. We first obtain image prototypes by applying K-means clustering to the features compressed from a pre-trained autoencoder, thereby capturing representative visual characteristics. Building on this foundation, we construct text prototypes for each cluster from textual descriptions generated by open-source vision-language models (VLMs). To ensure representativeness and diversity, words common across all clusters are excluded, as they fail to characterize individual clusters effectively. The sentence with the highest matching score to these feature words is selected as the final text prototype, ensuring an accurate representation of both the central theme and the unique characteristics of each cluster.

Compared with previous methods, our approach integrates both text and image prototypes to improve the performance of DD. Our method alleviates the previous issues

mentioned in [9, 32], and the generated images contain the intended objects and exhibit logical coherence. Furthermore, the experimental results demonstrate that the distilled dataset outperforms the state-of-the-art methods in top-1 accuracy. Specifically, we observe improvements of 3.9%, 4.9%, and 3.5% on ImageNette [10], and 2.9%, 4.2%, and 2.5% on ImageIDC [12], achieving superior performance over previous methods under IPC settings of 10, 20, and 50, respectively. The source code is provided in the supplementary material.

The contributions of this study are summarized as follows:

- To the best of our knowledge, this is the first work that integrates language information into visual dataset distillation for classification tasks. By leveraging textual descriptions, our approach enriches image-based information with crucial details such as shape, color, background, *etc.*, thereby mitigating existing limitations.
- We employ open-source vision-language models in DD to generate descriptive text for unimodal data, addressing the lack of textual descriptions in existing DD benchmarks and improving the generalization of our method.
- We propose a novel text prototype scheme for DD, which leverages word frequency within each cluster to ensure the representativeness and diversity of the text prototypes.

2. Preliminaries

Dataset distillation aims to construct a small compact dataset $S = \{(X_i, y_i)\}_{i=1}^{N_S}$ that encapsulates key information from a large-scale one $T = \{(X_i, y_i)\}_{i=1}^{N_T}$, where X_i represents an image, y_i denotes its corresponding class label, and $N_S \ll N_T$ [39, 42]. By synthesizing such a dataset, DD enables models trained on S to achieve comparable performance to those trained on T , while significantly reducing storage and computational costs.

Recently, diffusion models have emerged as powerful tools for generating high-quality synthetic data, making them a promising approach for dataset distillation. These models, known for their outstanding performance in generative tasks, synthesize high-quality images by adding Gaussian noise and then reversing the process to reconstruct them, ensuring consistency between input and output spaces. In this work, we employ Stable Diffusion [29, 36] for training, which comprises three key components: an image encoder (VAE), a text encoder (CLIP), and a U-Net. The VAE consists of an encoder E and a decoder D , where E projects the image into a latent space $z = E(x)$, and D reconstructs the latent code back to the image space $\hat{x} = D(z)$. The text encoder projects input prompts into the same feature space as the images, facilitating text-guided image generation. The U-Net utilizes a conditional model to process noisy latent z_t and predicts the noise, incorporating both the timestep (t) and the text embedding for guid-

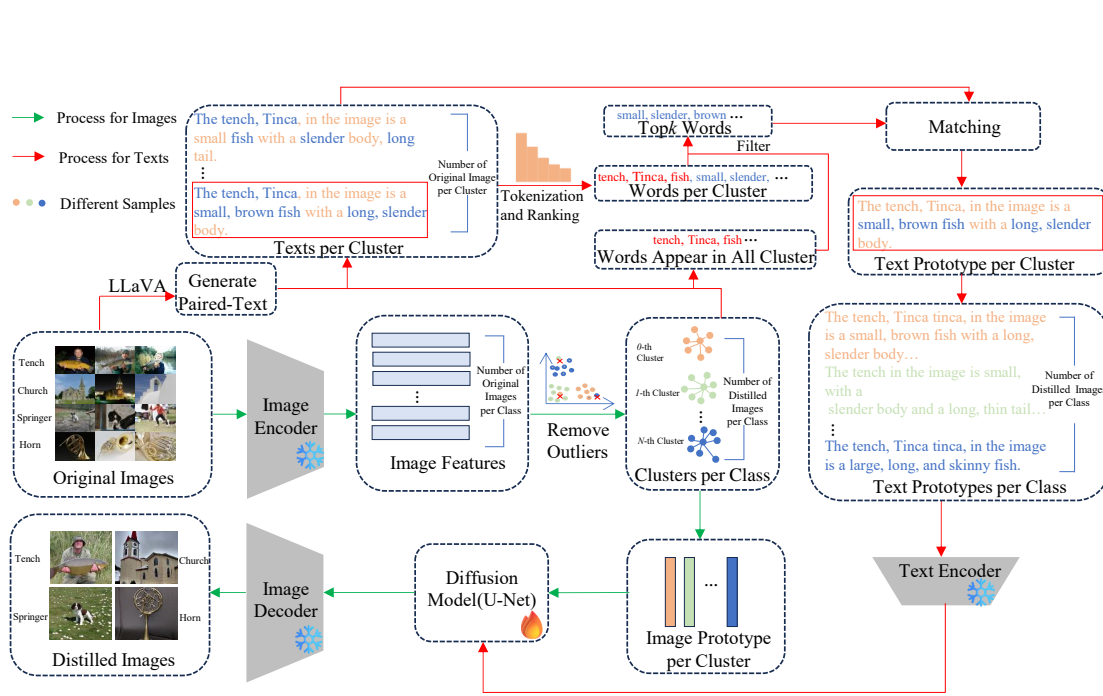


Figure 2. Overview of the proposed framework. The framework starts with generating image-text pairs using the LLaVA model, followed by training a diffusion model. Image features are then compressed with an autoencoder, outliers are removed, and K-means clustering is applied to create image prototypes. For text prototypes, frequent words are extracted from descriptions, and the most representative sentence is selected. Finally, these prototypes guide the diffusion model to synthesize diverse and representative images.

ance. The training objective of diffusion models is defined as:

$$\mathcal{L}_{\mathcal{DM}} = \|\epsilon_{\theta}(z_t, c) - \epsilon\|_2^2, \quad (1)$$

where c is conditioning vector encoded with corresponding text, $\epsilon_{\theta}(z_t, c)$ is the predicted noise, and ϵ is the ground truth.

3. Method

In this section, we present a vision-language dataset distillation method designed to enhance the validation performance of synthetic datasets, as illustrated in Fig. 2. By incorporating visual and textual information, our method addresses the limitations of conventional approaches that rely solely on image data. The complete distillation process is outlined in Algorithm 1.

3.1. Paired Description Generation

In contrast to [9, 32], which distills information solely from images, we also distill information from description texts to enhance dataset quality. However, the existing benchmark datasets lack corresponding descriptive textual information, and annotating these data manually is time-consuming and labor-intensive. Fortunately, the rapid development of vision-language models has made this task feasible. Hence, we leverage the open-source vision-language model LLaVA [18–20] to generate the corresponding text by designing

structured prompts. These descriptions capture additional semantic attributes, such as logical relationships and contextual details, which are not directly inferred from image features. The prompt is designed as follows:

Prompt = “Describe the physical appearance of the { $\$CLASSNAME$ } in the image. Include details about its shape, posture, color, and any distinct features.”

3.2. Outlier Removal

We apply Local Outlier Factor (LOF) [1], a widely used unsupervised outlier detection method, to identify and remove data points with significantly lower density compared to surrounding samples. In comparison to other methods, LOF does not require ground truth, making it suitable for datasets with unknown distributions. We set two parameters for the LOF algorithm: $n_{neighbors} = 10$ is set for all datasets, while $contamination$ is adjusted based on the dataset characteristics.

3.3. Cross-Modal Information Distillation

3.3.1. Image Prototypes

Following the work of [32], we first employ encoder E to compress features in the latent space. Subsequently, we apply k-means clustering, a widely used unsupervised algorithm, to partition each category into a predefined number of clusters. The number of clusters is dynamically set according to the IPC. For instance, when $IPC = 10$, the number of clusters is set to 10. The cluster centers are then used

Algorithm 1 Dataset Distillation via Vision-Language Category Prototype

```
1: Input:  $(R, L)$ : Real images and their labels,  $VLM$ :  
vision-language model,  $DM$ : Diffusion model  
2: Generate descriptive text:  $T = VLM(R)$   
3: DM training: Fine-tune the  $DM$  using image-text  
pairs  $(R, T)$ .  $E$ : Encoder,  $D$ : Decoder,  $\tau_\theta$ : Text en-  
coder,  $U_t$ : Time-conditional U-Net.  
4: for each  $l \in L$  do  
5:   Apply K-Means to partition  $l$  into  $C$  clusters.  
6:   Tokenize class text to obtain word-frequency set  
    $(w, freq)$ .  
7:   for each cluster do  
8:     Calculate cluster center  $z^c$  as image prototype  
9:     Extract text prototype  $T^c$  via Algorithm 2.  
10:  end for  
11:   $y = \tau_\theta(T^c)$            {Descriptive text embedding}  
12:  for each  $z^c$  do  
13:     $z_t^c \sim q(z_t^c | z^c)$        {Diffusion process}  
14:     $\tilde{z}^c = U_t(\text{Concat}(z_t^c, y))$  {Denoising process}  
15:  end for  
16: end for  
17:  $S = D(\tilde{Z}^c)$            {Generate image}  
18: Output:  $S$ : Distilled images
```

as image prototypes, effectively capturing the representative visual features of each category. These prototypes provide compact yet informative features, facilitating more effective dataset distillation.

3.3.2. Text Prototypes

Effective dataset distillation should consider not only visual features but also semantic information, including details such as shape, posture, color, and other distinct features that may not be captured by the image features. Hence, we introduce the frequency-based prototype extraction method to obtain the text prototype for each cluster. This approach involves tokenizing the description text, filtering out non-representative words, and selecting the most representative sentence as the text prototype based on word frequency. The words w with high frequency, appearing in more than β proportion of the samples within the same category are excluded, as they are unlikely to characterize individual clusters. The procedure for calculating text prototypes is summarized in Algorithm 2.

First, all textual descriptions within a given class l are tokenized to generate word-frequency set $(w, freq)$. Non-representative words N are identified and removed:

$$N = \left[w \mid w, \text{freq} \in (w, \text{freq}) \text{ and } \frac{\text{freq}}{\text{len}(l)} > \beta \right]. \quad (2)$$

Next, the text data within each cluster are tokenized into individual words, and common stop words (e.g., “is,”

Algorithm 2 Generate text prototype for each cluster

```
1: Input:  $(w, freq)$ : word-frequency set of class  $l$ ,  $T$ :  
descriptive texts of cluster  
2: Identify nonrepresentative words  $N$  based on Eq. 2  
3: Tokenize texts  $T$  to obtain word-frequency set  $(w_c, f_c)$   
4: Select top- $k$  representative words  $R_w$  based on Eq. 3  
5: Token: Tokenize sentences into words.  
6: for each text  $t \in T$  do  
7:   words = Token( $t$ )  
8:   calculate score via Eq. 4  
9: end for  
10: Output: Select the top-score  $t$  as the text prototype  $T^c$ 
```

“the,” “of”) are removed to generate a word-frequency set (w_c, f_c) . Nonrepresentative words are then excluded. The remaining words are ranked by frequency f_c and the top- k words are selected to generate representative words R_w :

$$R_w = \{w_c : f_c \mid w_c, f_c \in (w_c, f_c) \text{ and } w_c \notin N\}. \quad (3)$$

Subsequently, the frequency of words in R_w is then used as an importance score. Each text t in the cluster is evaluated based on its matching score within the top- k words, and the text with the highest score will be chosen as the text prototype.

$$\text{Score}(t) = \sum R_w[w] \cdot \mathbb{I}(w \in t), \quad (4)$$

$$\mathbb{I}(w \in t) = \begin{cases} 1, & \text{if } w \in R_w \\ 0, & \text{otherwise} \end{cases}. \quad (5)$$

3.4. Image Synthesis via LDM

Finally, we integrate the image and text prototypes into the latent diffusion model (LDM) to synthesize diverse and representative images. LDM employs a text encoder τ_θ to project the descriptive text into the latent space, which then conditions the U-Net architecture to guide image synthesis, facilitating the fusion of cross-modal representation. For each cluster, the synthesis process is formulated as follows:

$$\text{output} = D(U_t(\text{Concat}(z_t^c, \tau_\theta(T^c))))), \quad (6)$$

where D denotes the decoder and z_t^c represents the cluster image prototype with noise. T^c refers to the corresponding descriptive text prototype.

4. Experiments

4.1. Datasets

We evaluate the performance of our proposed method on both low-resolution and high-resolution datasets. For low-resolution datasets (32×32), we use CIFAR-10 [13] and CIFAR-100 [13]. For high-resolution data, we conduct

IPC (Ratio)	Test Model	Random	K-Center	Herdling	DiT	DM	IDC-1	GLaD	Minimax	D ⁴ M	Ours	Full
10 (0.8%)	ConvNet-6	24.3±1.1	19.4±0.9	26.7±0.5	<u>34.2±1.1</u>	26.9±1.2	33.3±1.1	33.8±0.9	33.3±1.7	29.4±0.9	34.8±2.4	86.4±0.2
	ResNetAP-10	29.4±0.8	22.1±0.1	32.0±0.3	34.7±0.5	30.3±1.2	<u>39.1±0.5</u>	32.9±0.9	36.2±3.2	33.2±2.1	39.5±1.5	87.5±0.5
	ResNet-18	27.7±0.9	21.1±0.4	30.2±1.2	34.7±0.4	33.4±0.7	<u>37.3±0.2</u>	31.7±0.8	35.7±1.6	32.3±1.2	39.9±2.6	89.3±1.2
20 (1.6%)	ConvNet-6	29.1±0.7	21.5±0.8	29.5±0.3	36.1±0.8	29.9±1.0	35.5±0.8	-	<u>37.3±0.1</u>	34.0±2.3	37.9±1.9	86.4±0.2
	ResNetAP-10	32.7±0.4	25.1±0.7	34.9±0.1	41.1±0.8	35.2±0.6	<u>43.4±0.3</u>	-	43.3±2.7	40.1±1.6	44.5±2.2	87.5±0.5
	ResNet-18	29.7±0.5	23.6±0.3	32.2±0.6	40.5±0.5	29.8±1.7	38.6±0.2	-	<u>41.8±1.9</u>	38.4±1.1	44.5±2.0	89.3±1.2
50 (3.8%)	ConvNet-6	41.3±0.6	36.5±1.0	40.3±0.7	46.5±0.8	44.4±1.0	43.9±1.2	-	<u>50.9±0.8</u>	47.4±0.9	54.5±0.6	86.4±0.2
	ResNetAP-10	47.2±1.3	40.6±0.4	49.1±0.7	49.3±0.2	47.1±1.1	48.3±1.0	-	<u>53.9±0.7</u>	51.7±3.2	57.3±0.5	87.5±0.5
	ResNet-18	47.9±1.8	39.6±1.0	48.3±1.2	50.1±0.5	46.2±0.6	48.3±0.8	-	<u>53.7±0.6</u>	<u>53.7±2.2</u>	58.9±1.5	89.3±1.2
70 (5.4%)	ConvNet-6	46.3±0.6	38.6±0.7	46.2±0.6	50.1±1.2	47.5±0.8	48.9±0.7	-	<u>51.3±0.6</u>	50.5±0.4	55.8±1.7	86.4±0.2
	ResNetAP-10	50.8±0.6	45.9±1.5	53.4±1.4	54.3±0.9	51.7±0.8	52.8±1.8	-	<u>57.0±0.2</u>	54.7±1.6	60.6±0.3	87.5±0.5
	ResNet-18	52.1±1.0	44.6±1.1	49.7±0.8	51.5±1.0	51.9±0.8	51.1±1.7	-	<u>56.5±0.8</u>	56.3±1.8	60.3±0.3	89.3±1.2
100 (7.7%)	ConvNet-6	52.2±0.4	45.1±0.5	54.4±1.1	53.4±0.3	55.0±1.3	53.2±0.9	-	57.8±0.9	<u>57.9±1.5</u>	62.7±1.4	86.4±0.2
	ResNetAP-10	59.4±1.0	54.8±0.2	61.7±0.9	58.3±0.8	56.4±0.8	56.1±0.9	-	<u>62.7±1.4</u>	59.5±1.8	65.7±0.5	87.5±0.5
	ResNet-18	61.5±1.3	50.4±0.4	59.3±0.7	58.9±1.3	60.2±1.0	58.3±1.2	-	62.7±0.4	<u>63.8±1.3</u>	68.3±0.4	89.3±1.2

Table 1. Comparison of state-of-the-art methods on ImageWoof under various IPC settings and model architectures. All the results are obtained at a resolution of 256×256 . The best results are marked as bold, and the second-best are underlined.

experiments on ImageNet-1K [5] dataset and its subsets: ImageWoof [11], ImageNette [10], ImageIDC [12], and ImageNet-100 [35]. ImageWoof includes 10 fine-grained dog breeds with high inter-class similarity, while ImageNette and ImageIDC comprise 10 coarse-grained classes. ImageIDC is derived from the first 10 classes of ImageNet-100.

4.2. Implementation Details

We conduct three independent trials with different seeds and report the average accuracy. We fine-tune the stable diffusion V1-5 model for each dataset using the generated image-text pairs. The batch size for fine-tuning is set to 8, and the training lasts 8 epochs. The resolution of the generated samples is set to 256×256 for ImageNet-1K subsets and 224×224 for the full ImageNet-1K dataset. For CIFAR-10 and CIFAR-100, the resolution is 32×32 . For fair evaluation, we utilize the publicly available source code from [9] to assess the performance of our method and report the top-1 accuracy on the original testing set. More implementation details are provided in the supplementary material.

4.3. Comparison with the SOTA Methods

We evaluate our method against the state-of-the-art approaches, including generative methods such as Minimax [9], D⁴M [32], GLaD [4], and DiT [9, 28]. Additionally, we compare our approach with decoupled distillation methods, including SRe²L [41] and RDED [34], as well as other techniques such as DM [44], IDC-1 [12], Herding [40], and K-Center [31]. The mean and standard deviation of the results are reported. We reproduce Minimax [9] method using the publicly available GitHub repository and conduct experiments under identical conditions to ensure a fair compari-

son.

ImageWoof We evaluate our method under varying IPC settings using three architectures: ConvNet-6, ResNetAP-10, and ResNet-18, as shown in Table 1. Across all settings and models, our method consistently outperforms the second-best method, demonstrating the robustness and adaptability of our approach. Notably, even with a low IPC (IPC = 10), our proposed method achieves 39.9% accuracy with the ResNet-18 model, surpassing the second-best method by 2.6%. As the IPC increases, the method still maintains its superiority, reaching 68.3% accuracy at IPC = 100 with ResNet-18, which yields an improvement of 4.5% compared to the D⁴M method. Furthermore, our method consistently achieves the best performance across various models—ConvNet-6, ResNetAP-10, and ResNet-18—demonstrating its robustness and generalization across different network architectures.

ImageNette and ImageIDC We assess our method using the ResNetAP-10 architecture under IPC 10, 20, and 50, as shown in Table 2. The results show that our method consistently outperforms all other approaches across varying IPC settings on both datasets. On the ImageNette dataset, our method significantly surpasses other methods, achieving a 4.1% improvement in average accuracy.

The lower performance on ImageIDC compared to ImageNette may be attributed to the presence of two similar fine-grained classes in IDC: Saluki, and Doberman. Despite this, our method achieves notable performance improvements, with a 3.2% increase in average accuracy, outperforming the state-of-the-art methods. The effectiveness of our method lies in its ability to simultaneously integrate both image and semantic information, unlike previous methods that only considered image features.

ImageNet-1K We conduct experiments under IPC val-

	IPC	Random	DiT	DM	Minimax	D ⁴ M	Ours
Nette	10	54.2±1.6	59.1±0.7	60.8±0.6	57.7±1.2	<u>60.9±1.7</u>	64.8±3.6
	20	63.5±0.5	64.8±1.2	<u>66.5±1.1</u>	64.7±0.8	66.3±1.3	71.4±0.5
	50	76.1±1.1	73.3±0.9	76.2±0.4	73.9±0.3	<u>77.7±1.1</u>	81.2±0.8
IDC	10	48.1±0.8	<u>54.1±0.4</u>	52.8±0.5	51.9±1.4	50.3±1.0	57.0±1.4
	20	52.5±0.9	58.9±0.2	58.5±0.4	<u>59.1±3.7</u>	55.8±0.2	63.3±1.2
	50	68.1±0.7	64.3±0.6	69.1±0.8	<u>69.4±1.4</u>	69.1±2.4	71.9±0.4

Table 2. Comparison of the state-of-the-art methods on ImageNette and ImageIDC under various IPC settings. All the results are obtained on ResNetAP-10. The best results are marked as bold, and the second-best are underlined.

IPC	SRe ² L	RDED	DiT	Minimax	Ours
10	21.3±0.6	42.0±0.1	39.6±0.4	44.3±0.5	46.7±0.4
50	46.8±0.2	56.5±0.1	52.9±0.6	58.6±0.3	60.5±0.2

Table 3. Performance comparison on ImageNet-1K.

Dataset	IPC	SRe ² L	RDED	Ours
CIFAR-10	10	29.3±0.5	37.1±0.3	39.0±0.7
	50	45.0±0.7	62.1±0.1	63.2±0.3
CIFAR-100	10	27.0±0.4	42.6±0.2	50.6±0.7
	50	50.2±0.4	62.6±0.1	66.1±0.3

Table 4. Performance comparison on CIFAR-10 and CIFAR-100.

ues of 10 and 50. All synthetic images are resized to 224×224 to ensure consistency with RDED [34]. Table 3 presents a performance comparison of various methods, including SRe²L [41], RDED [34], DiT [9, 28], Minimax [9]. The results indicate that our method consistently outperforms the others, achieving superior performance on large-scale datasets.

CIFAR-10 and CIFAR-100 We also evaluate our method on two low-resolution datasets, CIFAR-10 and CIFAR-100, both with a resolution of 32×32 . As shown in Table 4, our method surpasses SRe²L and RDED across different IPC settings on both datasets. Notably, on CIFAR-100 at IPC = 10, our method achieves a significant 8.0% improvement over RDED [34]. Our approach is especially effective for complex datasets like CIFAR-100, showcasing its robustness across datasets with various resolutions and complexities.

4.4. Ablation Study

As shown in Table 5, we conduct experiments with four configurations to evaluate various semantic strategies. These are assessed on two datasets, ImageIDC and ImageNette, under different IPC settings (10, 20, and 50).

Among the configurations, DCS consistently achieves the best performance across both datasets, except for ImageIDC at IPC-50. For instance, on ImageNette with IPC-

Semantic Methods	ImageIDC			ImageNette		
	IPC-10	IPC-20	IPC-50	IPC-10	IPC-20	IPC-50
L	54.1±0.5	61.5±1.1	71.2±1.2	60.1±1.6	69.7±1.6	76.6±0.5
L+FK	50.3±0.5	58.7±2.1	68.8±2.3	55.7±1.4	65.0±4.6	76.2±0.9
GGs	54.8±3.1	62.0±1.9	72.1±0.4	62.6±2.7	69.9±1.3	78.0±0.1
DCS	57.0±1.4	63.3±1.2	71.9±0.4	64.8±3.6	71.4±0.5	81.2±0.8

Table 5. Performance comparison on ImageNette and ImageIDC under various semantic methods: Label (L), Label + Feature Keywords (L+FK), GPT-Generated Sentences (GGs), and Descriptions of Closest Samples (DCS).

50, DCS achieves an accuracy of $81.2 \pm 0.8\%$, significantly outperforming other methods. Semantically closest sample descriptions provide highly relevant and context-rich information, enhancing synthesis quality and improving representativeness. In contrast, L+FK performs the worst overall. On ImageIDC with IPC-10, it achieves only $50.3 \pm 0.5\%$, as the feature keywords lack logical relationships and are often disorganized, resulting in the synthesis of poor-quality images. The baseline (L) shows better performance than L+FK in most cases, as its simplicity avoids the noise introduced by poorly contextualized keywords. However, it lacks the semantic depth necessary for further improvements. GGs demonstrates moderate performance by introducing richer semantic context, leading to improved results compared to L and L+FK. Notably, it reaches $72.1 \pm 0.4\%$ on ImageIDC with IPC-50, surpassing DCS in terms of average accuracy.

These results highlight the critical role of semantic information in improving the quality of the synthesized image. DCS consistently outperforms other methods, demonstrating the importance of context-rich descriptions to achieve superior synthesis performance.

4.4.1. Text prototype

The text prototype provides insights into the linguistic patterns associated with different clusters, highlighting both representative and nonrepresentative (N-R) characteristics, as shown in Table 6. In the “Saluki, gazelle hound” class, nonrepresentative words appear in more than 70% of the samples, including the class name itself and its common characteristics. For example, “white”, “long” and “slender” are classified as nonrepresentative words since they describe fundamental characteristics of the class: a slender dog with a long body and a coat that includes white. As these characteristics are prevalent across multiple clusters, they are unlikely to characterize individual clusters.

Feature keywords are selected on the basis of their frequency, which serves as an importance score. In the feature keywords of Cluster 3 in Table 6, we observe that 112 out of 166 samples describe a “grassy” background, 112 mention a “large” target size, 96 feature the color “brown” and 72 de-




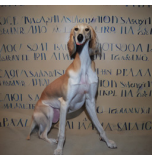
Cluster (N)	Image	N-R	Feature Keyword	Text Prototype
1 (145)			(large, 105), (shape, 94), (elegant, 80), (appearance, 77), (standing, 76), (curved, 75), (predominantly, 74), (markings, 74), (appears, 73), (head, 71), (brown, 67), (black, 62), (alert, 62), (legs, 60), (face, 59), (field, 59), (sleek, 56), (graceful, 53), (ears, 49), (grassy, 48), (possibly, 47), (breed, 42), (relaxed, 38), (pointed, 35), (features, 33), (adds, 33), (attentive, 31), (neck, 30), (coat, 29), (athletic, 27)	The Saluki, gazelle hound in the image is a large, slender dog with a long, lean body and a long tail. It has a distinctive shape, with a long head, pointed ears, and a long, curved muzzle. The dog is standing on a dirt road, and its posture appears alert and attentive. The color of the dog is predominantly white, with some brown markings on its face and body. The Saluki's features, such as its long legs and elegant posture, give it a graceful and athletic appearance.
2 (105)		white gazelle long hound dog image saluki	(appearance, 78), (large, 76), (brown, 55), (shape, 53), (appears, 53), (curved, 49), (markings, 49), (predominantly, 48), (elegant, 46), (standing, 46), (ears, 45), (possibly, 43), (face, 43), (alert, 43), (legs, 42), (head, 41), (attentive, 41), (black, 33), (sleek, 32), (relaxed, 29), (breed, 29), (graceful, 27), (features, 26), (pointed, 25), (unique, 24), (suggests, 22), (eyes, 22), (looking, 22), (field, 21), (adds, 21)	The Saluki, gazelle hound in the image is small and slender, with a long and sleek body. It has a distinctive shape, with a long head, large ears, and a long tail. The dog is standing on a red carpet, and its posture appears to be relaxed and comfortable. The color of the dog is predominantly brown, with some black markings on its face and body. The Saluki's unique features, such as its long legs, long neck, and elegant appearance, make it an attractive and graceful breed.
3 (166)		body slender color tail posture distinctive	(field, 130), (large, 112), (grassy, 112), (appearance, 107), (shape, 105), (brown, 96), (legs, 92), (predominantly, 82), (curved, 79), (markings, 79), (sleek, 79), (elegant, 78), (graceful, 77), (alert, 72), (running, 72), (head, 67), (ears, 64), (athletic, 62), (appears, 62), (standing, 59), (face, 56), (breed, 47), (black, 43), (focused, 41), (lean, 40), (possibly, 39), (adds, 38), (build, 36), (features, 34), (grass, 33)	The Saluki, gazelle hound in the image, has a slender and athletic build, with a long, lean body and a sleek coat. It has a distinctive shape, with a long, curved tail that extends downward. The dog's posture is energetic and graceful, as it is running swiftly across the grassy field. The Saluki's color is predominantly white, with some brown markings on its face and legs. The dog's eyes are open, and it appears focused on its surroundings, which adds to its overall dynamic appearance.
4 (120)			(appearance, 92), (large, 90), (brown, 70), (elegant, 67), (head, 64), (shape, 62), (curved, 60), (appears, 54), (alert, 51), (legs, 50), (ears, 48), (standing, 48), (predominantly, 48), (markings, 45), (graceful, 41), (coat, 40), (relaxed, 38), (face, 37), (pointed, 37), (black, 31), (unique, 31), (looking, 30), (possibly, 30), (breed, 29), (features, 29), (eyes, 29), (adds, 29), (sleek, 29), (attentive, 28), (field, 27)	The Saluki, gazelle hound in the image is a large, white dog with a slender body and long legs. It has a distinctive shape, with a long head, a long neck, and a long tail. The dog appears to be well-groomed and well-behaved, standing on a blue carpet with a woman. The Saluki's posture is relaxed, and its color is predominantly white, with possibly some black markings on its face or body. The dog's overall appearance is elegant and graceful, which is typical of the Saluki breed.

Table 6. An example of text prototypes corresponding to a “Saluki, gazelle hound” class from dataset ImageIDC. Cluster (N) represents the cluster ID and sample size, while N-R denotes nonrepresentative words. Feature keywords are represented as (word, frequency) pairs.

pick the action “running”. This indicates that nearly half of the brown dogs are running on the grass, which is consistent with the generated images. Moreover, compared to images generated using only labels shown in Fig. 3, our method produces more natural running postures and preserves detailed target features such as a curved tail. It also enhances logical consistency, such as dogs with four legs rather than the five legs seen in the label-only images.

4.5. Visualization

To evaluate the quality of the synthesized images, we compare samples generated using the same image prototype (corresponding to each column) across different semantic strategies, as illustrated in Fig. 3. The images on the left of the dashed line are sourced from ImageIDC and depict a Saluki, while those on the right are from ImageNette and represent a tench. It can be observed that images generated by L, L+FK, and GGS all exhibit illogical outputs and the absence of objects. In contrast, DCS generates images that are more natural and structurally coherent, effectively preventing the absence of target objects.

In the Saluki case, L, L+FK, and GGS generate images with severe flaws, such as extra or missing limbs, while DCS consistently generates a logically coherent Saluki with

the correct number of legs. Additionally, in the fourth column, only DCS successfully synthesizes images containing the target object, while the other methods fail to do so. Although L+FK contains more features than L, these words are unordered and unstructured, and they lack logical relationships, leading to misinterpretations. GGS generates sentences based on FK not encountered during the model’s training, which may fail to provide the necessary context for accurate object generation. In contrast, DCS offers more detailed and context-rich descriptions, ensuring that the target object is consistently included in the generated image with all relevant features. More sample visualizations are provided in the supplementary material.

4.6. Parameter Analysis

We analyze the sensitivity of parameters α (Contamination), β (Nonrepresentative Threshold), and k (Top- k words) on ImageIDC, as shown in Fig. 4 (a)-(c). The contamination parameter α has a significant impact on the performance. Models with lower IPC values (IPC = 10, 20, and 50) exhibit greater sensitivity to noisy data, resulting in more pronounced fluctuations in accuracy. In contrast, models with higher IPC values (IPC = 100) demonstrate stronger robustness, maintaining relatively stable accuracy.

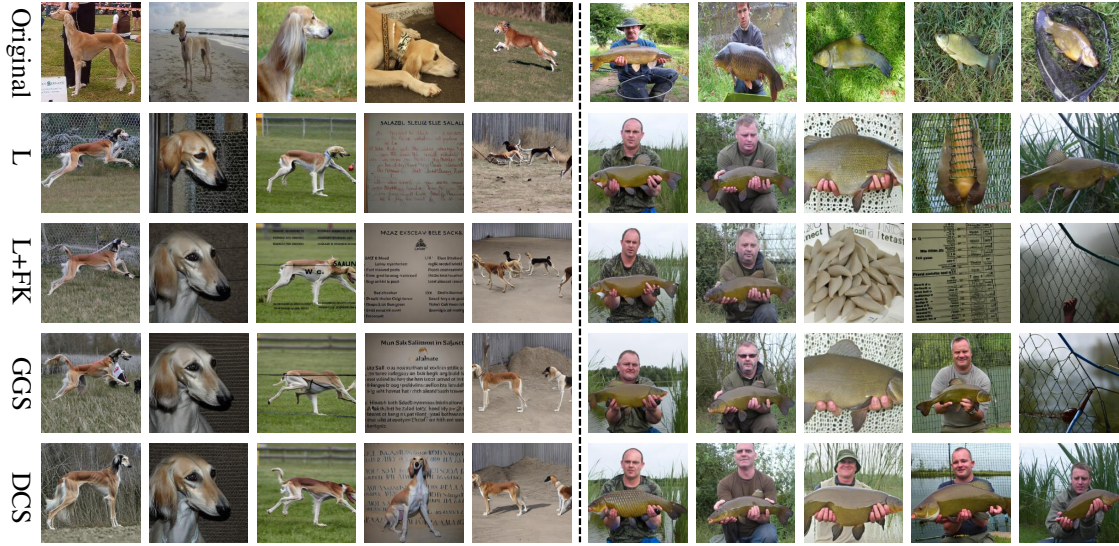


Figure 3. Visualization of images generated using different semantic strategies. For each column, the images are generated using the same image prototype and random seed. In comparison, DCS produces images that are significantly more natural and logical.

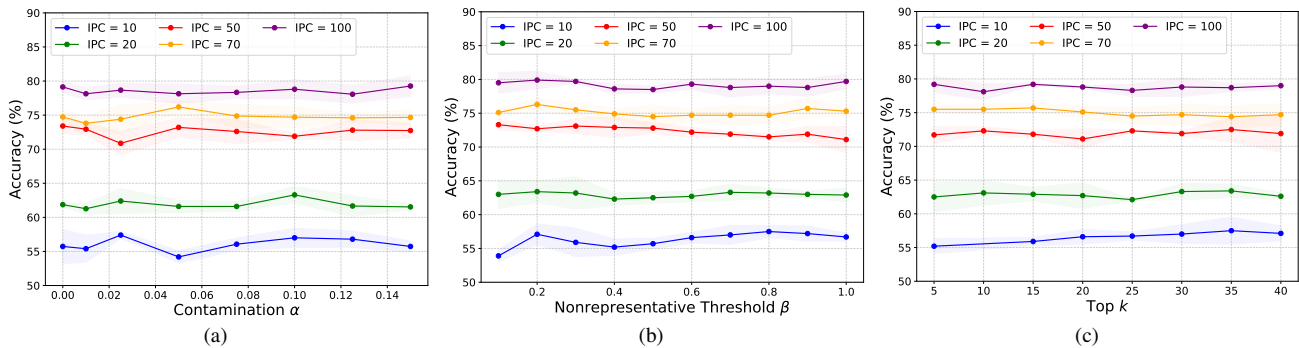


Figure 4. Parameter Analysis of α (Contamination), β (Nonrepresentative Threshold), and k (Top- k words) on ImageIDC.

For β , except for IPC = 50, which exhibits a decreasing trend, other settings reach a peak at 0.2 before declining and stabilizing. Words appearing in more than 20% of the samples in each class are classified as nonrepresentative words. As this threshold increases, high-frequency words within a class may be incorrectly selected as feature keywords for the cluster, reducing diversity. Regarding parameter k , as the value of top- k increases, models with lower IPC values show a gradual increase in accuracy, reaching a maximum of 35, after which performance declines. This decline likely results from the inclusion of an increasing number of non-representative words at higher top- k values, leading to the selection of suboptimal text prototypes.

5. Conclusion and Future Work

In this work, we have proposed a novel dataset distillation method based on vision-language category prototypes.

For the first time, we introduce text prototypes to complement image prototypes in dataset distillation, significantly enhancing the performance of the generated surrogate dataset. Compared to previous approaches, our method not only generates more logically coherent images containing target objects but also achieves outstanding performance across multiple benchmarks. By integrating the complementary strengths of visual and textual information, our approach provides a fresh perspective on dataset distillation, advancing the development of more efficient distillation techniques.

Limitations and Future Works. Our current work primarily focuses on classification tasks. In future research, we plan to extend our method to more complex vision tasks, such as object detection and segmentation, to evaluate its broader applicability. Additionally, we aim to explore alternative strategies for integrating text and image prototypes to further enhance the effectiveness of dataset distillation.

Acknowledgement: This work was partly supported by JSPS KAKENHI Grant Number 23K10712.

References

- [1] Omar Alghushairy, Raed Alsini, Terence Soule, and Xiaogang Ma. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data and Cognitive Computing*, 5(1):1, 2020. 3
- [2] Nicholas Carlini, Vitaly Feldman, and Milad Nasr. No free lunch in” privacy for free: How does dataset condensation help privacy”. *arXiv preprint arXiv:2209.14987*, 2022. 1
- [3] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4750–4759, 2022. 1
- [4] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Generalizing dataset distillation via deep generative prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3739–3748, 2023. 5
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 1, 5
- [6] Zhiwei Deng and Olga Russakovsky. Remember the past: Distilling datasets into addressable memories for neural networks. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 34391–34404, 2022. 1
- [7] Tian Dong, Bo Zhao, and Lingjuan Lyu. Privacy for free: How does dataset condensation help privacy? In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 5378–5396, 2022. 1
- [8] Zongxion Geng, Jiahui andg Chen, Yuandou Wang, Herbert Woisetschlaeger, Sonja Schimmler, Ruben Mayer, Zhiming Zhao, and Chunming Rong. A survey on dataset distillation: Approaches, applications and future directions. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2023. 1
- [9] Jianyang Gu, Saeed Vahidian, Vyacheslav Kungurtsev, Haonan Wang, Wei Jiang, Yang You, and Yiran Chen. Efficient dataset distillation via minimax diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15793–15803, 2024. 1, 2, 3, 5, 6
- [10] Jeremy Howard. Imagenette: A smaller subset of 10 easily classified classes from imagenet, 2019. 2, 5
- [11] Jeremy Howard. Imagewoof: a subset of 10 classes from imagenet that aren’t so easy to classify, 2019. 5
- [12] Jang-Hyun Kim, Jinuk Kim, Seong Joon Oh, Sangdoon Yun, Hwanjun Song, Joonhyun Jeong, Jung-Woo Ha, and Hyun Oh Song. Dataset condensation via efficient synthetic-data parameterization. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 11102–11118, 2022. 1, 2, 5
- [13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images.(2009), 2009. 4
- [14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015. 1
- [15] Saehyung Lee, Sanghyuk Chun, Sangwon Jung, Sangdoon Yun, and Sungroh Yoon. Dataset condensation with contrastive signals. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 12352–12364, 2022. 1
- [16] Shiye Lei and Dacheng Tao. A comprehensive survey to dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):17–32, 2023. 1
- [17] Guang Li, Bo Zhao, and Tongzhou Wang. Awesome dataset distillation. <https://github.com/Guang000/Awesome-Dataset-Distillation>, 2022. 1
- [18] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 34892–34916, 2023. 3
- [19] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.
- [20] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. <https://llava-vl.github.io/blog/2024-01-30-llava-next/>, 2024. 3
- [21] Mengyang Liu, Shanchuan Li, Xinshi Chen, and Le Song. Graph condensation via receptive field distribution matching. *arXiv preprint arXiv:2206.13697*, 2022. 1
- [22] Ping Liu and Jiawei Du. The evolution of dataset distillation: Toward scalable and generalizable solutions. *arXiv preprint arXiv:2502.05673*, 2025. 1
- [23] Yanqing Liu, Jianyang Gu, Kai Wang, Zheng Zhu, Wei Jiang, and Yang You. Dream: Efficient dataset distillation by representative matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17314–17324, 2023. 1
- [24] Noel Loo, Ramin Hasani, Alexander Amini, and Daniela Rus. Efficient dataset distillation using random feature approximation. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 13877–13891, 2022. 1
- [25] Noel Loo, Ramin Hasani, Mathias Lechner, and Daniela Rus. Dataset distillation with convexified implicit gradients. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 22649–22674, 2023. 1
- [26] Brian B Moser, Federico Raue, Sebastian Palacio, Stanislav Frolov, and Andreas Dengel. Latent dataset distillation with diffusion models. *arXiv preprint arXiv:2403.03881*, 2024. 1
- [27] Timothy Nguyen, Zhourong Chen, and Jaehoon Lee. Dataset meta-learning from kernel ridge-regression. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 1
- [28] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF interna-*

- tional conference on computer vision (CVPR), pages 4195–4205, 2023. 5, 6
- [29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2
- [30] Noveen Sachdeva and Julian McAuley. Data distillation: A survey. *Transactions on Machine Learning Research*, 2023. 1
- [31] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 5
- [32] Duo Su, Junjie Hou, Weizhi Gao, Yingjie Tian, and Bowen Tang. D⁴: Dataset distillation via disentangled diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5809–5818, 2024. 1, 2, 3, 5
- [33] Iliia Sucholutsky and Matthias Schonlau. Soft-label dataset distillation and text dataset distillation. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2021. 1
- [34] Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9390–9399, 2024. 5, 6
- [35] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 776–794, 2020. 5
- [36] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. <https://github.com/huggingface/diffusers>, 2022. 2
- [37] Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12196–12205, 2022. 1
- [38] Kai Wang, Jianyang Gu, Daquan Zhou, Zheng Zhu, Wei Jiang, and Yang You. DiM: Distilling dataset into generative model. *arXiv preprint arXiv:2303.04707*, 2023. 1
- [39] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A. Efros. Dataset distillation. *arXiv preprint arXiv:1811.10959*, 2018. 1, 2
- [40] Max Welling. Herding dynamical weights to learn. In *Proceedings of the international conference on machine learning (ICML)*, pages 1121–1128, 2009. 5
- [41] Zeyuan Yin, Eric Xing, and Zhiqiang Shen. Squeeze, recover and relabel: Dataset condensation at imagenet scale from a new perspective. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 5, 6
- [42] Ruonan Yu, Songhua Liu, and Xinchao Wang. A comprehensive survey to dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1):150–170, 2023. 2
- [43] Bo Zhao and Hakan Bilen. Dataset condensation with differentiable siamese augmentation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 12674–12685, 2021. 1
- [44] Bo Zhao and Hakan Bilen. Dataset condensation with gradient matching. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 1, 5
- [45] Bo Zhao and Hakan Bilen. Synthesizing informative training samples with gan. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Workshop*, 2022. 1
- [46] Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6514–6523, 2023. 1
- [47] Ganlong Zhao, Guanbin Li, Yipeng Qin, and Yizhou Yu. Improved distribution matching for dataset condensation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7856–7865, 2023. 1
- [48] Yongchao Zhou, Ehsan Nezhadarya, and Jimmy Ba. Dataset distillation using neural feature regression. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 9813–9827, 2022. 1