# Training-Free Synthetic Data Generation with Dual IP-Adapter Guidance

Luc Boudier* [1]
luc.boudier@polytechnique.edu

Loris Manganelli* [1]
loris.manganelli@polytechnique.edu

Eleftherios Tsonis* [1]
eleftherios.tsonis@polytechnique.edu

Nicolas Dufour[1][2]
nicolas.dufour@enpc.fr

Vicky Kalogeiton[1]
vicky.kalogeiton@polytechnique.edu

[1] LIX,
École Polytechnique,
IP Paris, CNRS

[2] LIGM
École des Ponts,
IP Paris, CNRS, UGE

## Abstract

Few-shot image classification remains challenging due to the limited availability of labeled examples. Recent approaches have explored generating synthetic training data using text-to-image diffusion models, but often require extensive model fine-tuning or external information sources. We present a novel training-free approach, called DIPSY, that leverages IP-Adapter for image-to-image translation to generate highly discriminative synthetic images using only the available few-shot examples. DIPSY introduces three key innovations: (1) an extended classifier-free guidance scheme that enables independent control over positive and negative image conditioning; (2) a class similarity-based sampling strategy that identifies effective contrastive examples; and (3) a simple yet effective pipeline that requires no model fine-tuning or external captioning and filtering. Experiments across ten benchmark datasets demonstrate that our approach achieves state-of-the-art or comparable performance, while eliminating the need for generative model adaptation or reliance on external tools for caption generation and image filtering. Our results highlight the effectiveness of leveraging *dual image prompting* with positive-negative guidance for generating class-discriminative features, particularly for fine-grained classification tasks. Project page: https://www.lix.polytechnique.fr/vista/projects/2025_bmvc_dipsy/.

## 1 Introduction

Few-shot image classification remains challenging when labeled data is scarce or expensive to collect. The ability to accurately categorize images into classes with only a handful of labeled examples has significant implications across domains like medical imaging, industrial inspection, and rare species identification [41, 43, 48]. Traditional approaches to
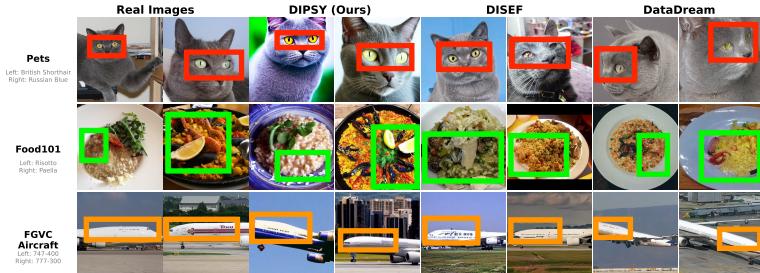
* Equal contribution.

Figure 1: Qualitative comparison of synthetic image generation for visually similar class **pairs** across datasets: British Shorthair vs Russian Blue (Pets), Risotto vs Paella (Food101), and Boeing 747-400 vs 777-300 (FGVC Aircraft). DIPSY generates semantically faithful and visually distinct images, preserving class-specific cues such as eye color in pets, food-specific textures and toppings, and structural aircraft details. Competing methods (DISEF and DataDream) often produce ambiguous results. Real images included for reference. **Note:** Colored bounding boxes are manually drawn to emphasize the discriminative features captured by our method.

few-shot learning often struggle to extract generalizable representations from limited samples, resulting in models that perform poorly on novel examples [62]. Generative models [14, 20, 51, 53] have opened promising avenues to address this limitation by creating synthetic training data to augment the small number of real examples available [3, 6, 7, 13, 56].

Recent work has explored leveraging text-to-image diffusion models to generate synthetic data for training image classifiers. Methods such as DataDream [56] adapt generative models through parameter-efficient fine-tuning (e.g. LoRA [31, 54]) to align with the target data distribution, achieving impressive results across multiple datasets. Similarly, DISEF [13] employs a strategy that combines in-domain synthesis with efficient fine-tuning of vision-language models. Despite their promising results, these approaches face important limitations: DataDream requires complex model fine-tuning procedures, while DISEF, although training-free for the generative model, relies on captioning and filtering stages and sometimes fails to capture the fine-grained discriminative features necessary for challenging classification tasks. Moreover, on complex datasets with high inter-class similarity, such as FGVC Aircraft, Food, Flowers or Pets, it necessitates the use of very low denoising strengths, which produces images overly close to the originals.

In this paper, we propose DIPSY (**D**ual **IP**-Adapter **Sy**nthesizer), a novel approach that leverages image-to-image translation via IP-Adapter to generate highly discriminative synthetic images for few-shot classification. Unlike DataDream [56], DIPSY is training-free, eliminating computationally expensive generative model adaptation; unlike DISEF [13], it requires no external captioning or filtering, relying solely on the few-shot examples themselves. Our framework introduces key innovations including a class similarity-based sampling strategy for selecting effective image prompts and an extended formulation of classifier-free guidance (CFG) that independently modulates positive and negative IP-Adapter guidances. This dual guidance system provides fine-grained control over the generation process, where stronger positive guidance enhances class-specific feature preservation while stronger negative guidance increases inter-class feature discrimination. This allows us to optimize the synthetic data distribution for superior classifier performance.

DIPSY achieves performance comparable to or exceeding state-of-the-art methods on ten few-shot classification benchmarks, particularly excelling in fine-grained classification by accurately capturing discriminative features. Ablation studies confirm the importance of the negative guidance and class-similarity sampling strategies. By eliminating the need for model fine-tuning, captioning and filtering, DIPSY offers a more accessible solution for real-world applications with limited computational resources or domain-specific knowledge. As illustrated in Fig. 1, our method generates semantically faithful, visually distinct images that preserve class-specific discriminative cues, such as eye color in pets, textures in cuisine, and structural details in aircrafts, while competing methods often produce ambiguous results.

Our contributions are: (1) A training-free image synthesis framework that avoids external captioning, filtering, and generative model fine-tuning, making it efficient and practical. (2) A novel extension to classifier-free guidance (CFG) [29] providing independent control over text, positive, and negative image conditioning for precise generation. DIPSY is the first to apply such distinct positive and negative image conditioning within CFG. (3) A class similarity-based sampling strategy that selects images as negative prompts, enhancing the discriminative power of the generated images. All these yield a powerful yet simple approach that achieves performance comparable to the state of the art while requiring fewer resources.

# 2  Related Work

**Generative and Diffusion Models.** Advances in Variational AutoEncoders (VAEs) [37] and Generative Adversarial Networks (GANs) [24] led to substantial advancements in tasks like high-fidelity image synthesis [9], text-to-image synthesis with GANs [34], image translation [19], and even audio synthesis [18]. More recently, diffusion models [30, 46, 59] have demonstrated superior performance in generating high-quality and diverse samples, particularly in image synthesis and text-to-image applications [16, 50, 51, 53, 56]. Our method leverages IP-Adapter [64] with diffusion models like Stable Diffusion [51, 53], integrating image prompts for controllable generation without requiring base model fine-tuning.

**Classifier-Free Guidance.** Controlling generative models often involves conditioning the generation process on input data. Diffusion models can be trained to be unconditional or to incorporate conditioning via architectural mechanisms such as cross-attention [4, 53, 56], addition to features within network blocks [46], or input concatenation. To exert finer control over the generation process or to impose conditions not explicitly learned by the diffusion model during its training, guidance strategies are employed. Two prominent strategies are Classifier Guidance (CG) [16], which relies on an external classifier to steer sampling, and Classifier-Free Guidance (CFG) [29]. CFG, which avoids an external classifier by jointly training conditional and unconditional objectives (see Eq. (2)) is widely used in text-to-image diffusion models, enabling strong performance in systems like Stable Diffusion [53] (often using CLIP [52]), GLIDE [45], and Blended Diffusion [2]. Notably, InstructPix2Pix [10] extended CFG for dual conditioning on an image and a text instruction, a principle related to our multi-modal CFG. Our method takes inspiration from the classical CFG formulation but extends it to handle one textual prompt and two distinct image prompts from IP-Adapter. While recent work has explored CFG refinements such as guidance scheduling [61], interval-based guidance [40], timestep skipping [17], or reinterpreting CFG with weaker guiding models [55], our focus is on adapting the core CFG mechanism for multi-modal image conditioning.

**Training with Synthetic Data.** Recent research has extensively explored training models
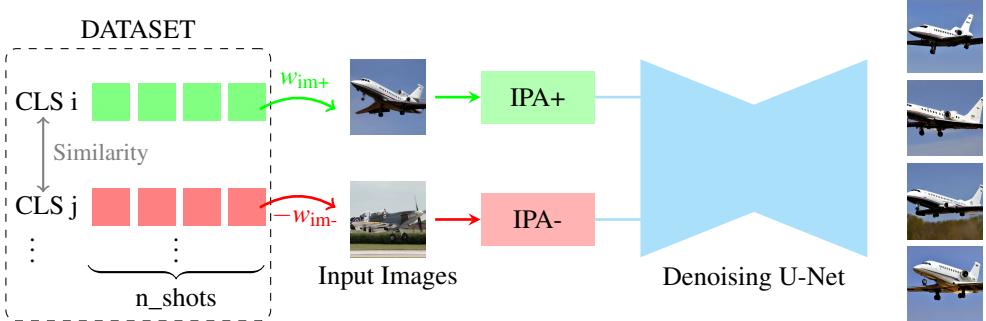
Figure 2: **Dual IP-Adapter generation pipeline.** An image from the target class (CLS *i*) provides positive conditioning (IPA+, weight $w_{im+}$), while an image from a similar class (CLS *j*) provides negative conditioning (IPA-, weight $w_{im-}$). These guide the Denoising U-Net to produce images of the target class.

with synthetic data. This includes augmenting real datasets [5, 11, 15, 21, 56], pre-training on synthetic data before fine-tuning on real samples [25, 60], and training entirely on synthetic datasets [25, 27, 57]. These strategies have been applied to various tasks such as classification [5, 11, 26, 57, 58], object detection [42], generative model self-consumption [1], and representation learning [60]. To improve synthetic data quality, studies have focused on enhancing faithfulness (e.g., using CLIP filtering [21, 26, 42], class-based selection [57], or spectral clustering [42]) and promoting diversity (e.g., by reducing guidance scales [57], using varied LLM-generated prompts [25, 26], incorporating diverse domains/backgrounds [21, 57, 58], or employing prompt templates [11]). Synthetic data's cost-effectiveness and scalability make it a compelling alternative or supplement to real data [57].

In the few-shot setting, which is our focus, a key challenge is personalizing large text-to-image models from just a few example images. Methods follow two main paths: prompt-tuning and model fine-tuning. Prompt-tuning methods learn new word embeddings to represent concepts [23, 33]. Fine-tuning methods adjust model weights, either by modifying the entire U-Net with DreamBooth [54] as in DataDream [36], or more efficiently by updating only the cross-attention layers with Custom Diffusion [39]. Our approach aligns with using diffusion models for new training data but distinctively leverages a pre-trained IP-Adapter without fine-tuning the generator, saving considerable training time. Modulating image guidances allows a balance between generation quality and diversity. Furthermore, integrating class similarity to select negative prompts from related classes actively enhances unique class features in generated images, optimizing them for classifier training.

# 3     Method

This section presents our proposed **D**ual **IP**-Adapter **Sy**nthesizer (DIPSY). As Fig. 2 illustrates, DIPSY generates a synthetic dataset accurately representing target class characteristics from $n_{shots}$ real images by leveraging IP-Adapter [54] and Stable Diffusion 1.5 [53] (chosen for its compatibility with both DataDream and IP-Adapter). DIPSY introduces a novel extension to classifier-free guidance (CFG) for independent control over textual, positive (green block in Fig. 2), and negative (red block in Fig. 2) image conditioning (Sec. 3.1).

A class similarity-based sampling strategy to select effective negative prompts further enhances this (Sec. 3.2). The resulting synthetic images and original few-shot examples are then used to fine-tune a CLIP model via a weighted loss function (Sec. 3.3).

## 3.1 Extending Classifier-Free Guidance for Multi-Modal Conditioning

IP-Adapter with Stable Diffusion 1.5 enables multi-modal conditioning: guidance originates from a textual prompt, a positive image prompt, and a negative image prompt (both using few-shot examples). Standard CFG typically assumes a single conditioning source (text) for scaling the deviation from the unconditional prediction. Our setup requires independent control over text, positive, and negative image conditioning contributions. We achieve this by implementing a modified CFG scheme that permits separate guidance factors for the text prompt, the positive IP-Adapter prompt, and the negative IP-Adapter prompt.

To understand our approach, let's first revisit standard classifier-free guidance (CFG). The goal of CFG is to estimate the score function conditioned on a text prompt $c_{\text{text}}$:

$$\varepsilon_\theta(x_t, c_{\text{text}}) \approx \nabla_{x_t} \log p(x_t | c_{\text{text}}) \quad . \tag{1}$$

Here, $\varepsilon_\theta$ represents the noise prediction model parameterized by $\theta$ at timestep $t$ for noisy input $x_t$. Standard CFG computes the final noise prediction $\hat{\varepsilon}_\theta$ by extrapolating from the unconditional prediction $\varepsilon_\theta(x_t)$ towards the text-conditional prediction $\varepsilon_\theta(x_t, c_{\text{text}})$, scaled by the text guidance factor $w_{\text{text}}$:

$$\hat{\varepsilon}_\theta(x_t, c_{\text{text}}) = \varepsilon_\theta(x_t) + w_{\text{text}}(\varepsilon_\theta(x_t, c_{\text{text}}) - \varepsilon_\theta(x_t)) \quad . \tag{2}$$

Our objective is to extend this framework to incorporate guidance from image prompts in addition to the text prompt, allowing for separate guidance strengths. We first consider the case with one image prompt, $c_{\text{im}}$. We aim to compute the jointly conditioned noise prediction $\hat{\varepsilon}_\theta(x_t, c_{\text{text}}, c_{\text{im}})$, which corresponds to approximating the score function $\nabla_{x_t} \log p(x_t | c_{\text{text}}, c_{\text{im}})$. To do so, we first use Bayes' theorem:

$$p(x_t | c_{\text{text}}, c_{im}) = \frac{p(c_{\text{text}}, c_{\text{im}} | x_t) p(x_t)}{p(c_{\text{text}}, c_{\text{im}})} = \frac{p(c_{\text{text}} | x_t) p(c_{\text{im}} | x_t, c_{\text{text}}) p(x_t)}{p(c_{\text{text}}, c_{\text{im}})} \quad . \tag{3}$$

Following common practice in guided diffusion models, we make an approximation involving guidance parameters $w_{\text{text}}$ and $w_{\text{im}}$:

$$p(x_t | c_{\text{text}}, c_{\text{im}}) \propto p(c_{\text{text}} | x_t)^{w_{\text{text}}} p(c_{\text{im}} | x_t, c_{\text{text}})^{w_{\text{im}}} p(x_t) \quad , \tag{4}$$

Taking the gradient of the log with respect to $x_t$ gives the score function approximation:

$$\nabla_{x_t} \log p(x_t | c_{\text{text}}, c_{\text{im}}) \approx w_{\text{text}} \nabla_{x_t} \log p(c_{\text{text}} | x_t) + w_{\text{im}} \nabla_{x_t} \log p(c_{\text{im}} | x_t, c_{\text{text}}) + \nabla_{x_t} \log p(x_t) \quad . \tag{5}$$

Now, we apply Bayes' theorem again to express the conditional probabilities in terms of the diffusion model's predictive capabilities:

$$p(c_{\text{text}} | x_t) = \frac{p(x_t | c_{\text{text}}) p(c_{\text{text}})}{p(x_t)} \quad \text{and} \quad p(c_{\text{im}} | x_t, c_{\text{text}}) = \frac{p(x_t | c_{\text{im}}, c_{\text{text}}) p(c_{\text{im}} | c_{\text{text}})}{p(x_t | c_{\text{text}})} \quad . \tag{6}$$

By substituting the logarithm gradients derived from (6) into (5), and assuming the gradients of the log priors $\log p(c_{\text{text}})$ and $\log p(c_{\text{im}}|c_{\text{text}})$ with respect to $x_t$ are negligible, we get:

$$\nabla_{x_t} \log p(x_t|c_{\text{text}}, c_{\text{im}}) \approx \nabla_{x_t} \log p(x_t) + w_{\text{text}}\left(\nabla_{x_t} \log p(x_t|c_{\text{text}}) - \nabla_{x_t} \log p(x_t)\right)$$
$$+ w_{\text{im}}\left(\nabla_{x_t} \log p(x_t|c_{\text{text}}, c_{\text{im}}) - \nabla_{x_t} \log p(x_t|c_{\text{text}})\right) \quad . \quad (7)$$

Replacing the score functions $\nabla_{x_t} \log p(x_t|\dots)$ with their corresponding noise predictions $\varepsilon_\theta(x_t|\dots)$, we arrive at the CFG formula for one text and one image prompt:

$$\hat{\varepsilon}_\theta(x_t, c_{\text{text}}, c_{\text{im}}) = \varepsilon_\theta(x_t) + w_{\text{text}}\left(\varepsilon_\theta(x_t|c_{\text{text}}) - \varepsilon_\theta(x_t)\right) + w_{\text{im}}\left(\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im}}) - \varepsilon_\theta(x_t|c_{\text{text}})\right) \quad . \quad (8)$$

Finally, we extend this reasoning iteratively to handle two image prompts, $c_{\text{im+}}$ (positive) and $c_{\text{im-}}$ (negative), with distinct guidance scale magnitudes $w_{\text{im+}}$ and $-w_{\text{im-}}$. We apply the logic sequentially: first add the conditioning $c_{\text{im+}}$ with weight $w_{\text{im+}}$ based on $c_{\text{text}}$ (as in Eq. (8)), and then add the conditioning $c_{\text{im-}}$ with weight $-w_{\text{im-}}$ based on the joint conditioning $(c_{\text{text}}, c_{\text{im+}})$. This yields our proposed formula for DIPSY's dual IP-Adapter guidance.

**DIPSY's Guidance Scheme**: *Given one text prompt $c_{\text{text}}$, one positive image prompt $c_{\text{im+}}$, and one negative image prompt $c_{\text{im-}}$, with corresponding guidance scales $w_{\text{text}}$, $w_{\text{im+}}$, and $-w_{\text{im-}}$, our proposed extended CFG scheme yields the following noise prediction $\hat{\varepsilon}_\theta$:*

$$\hat{\varepsilon}_\theta(x_t, c_{\text{text}}, c_{\text{im+}}, c_{\text{im-}}) = \varepsilon_\theta(x_t) + w_{\text{text}}\left(\varepsilon_\theta(x_t|c_{\text{text}}) - \varepsilon_\theta(x_t)\right)$$
$$+ w_{\text{im+}}\left(\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}}) - \varepsilon_\theta(x_t|c_{\text{text}})\right) - w_{\text{im-}}\left(\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}}, c_{\text{im-}}) - \varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}})\right) \quad . \quad (9)$$

Compared to standard text-only CFG, this extended guidance scheme requires one additional forward pass through the diffusion model's U-Net for the single IP-Adapter case (Eq. (8)), evaluating $\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}})$, and two additional forward passes to compute $\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}})$ and $\varepsilon_\theta(x_t|c_{\text{text}}, c_{\text{im+}}, c_{\text{im-}})$, for the dual (positive/negative) IP-Adapter case (Eq. (9)).

## 3.2   Image generation

Given a dataset with only a few images per class (referred to as the number of shots $n_{\text{shots}}$), the number of real images is $n_{\text{real}} = n_{\text{classes}} \cdot n_{\text{shots}}$. This data is used to generate new synthetic images, with a diffusion model and two IP-Adapters. This number is referred to as $n_{\text{synth}}$.

**Class Similarity.** We model inter-class relationships by computing pairwise CLIP ViT-B/16 cosine similarities from few-shot images, averaging them for class-level scores. A softmax yields a probability distribution for sampling negative classes. To generate synthetic images, we repeat the process of Fig. 2 $n_{\text{synth}}$ times: A random real image from class $i$ acts as the positive IP-Adapter prompt. A negative prompt image is chosen from a different class $j$ based on the similarity distribution. Both augmented (random crop/rotation) images guide diffusion with weights $w_{\text{im+}}$ (positive) and $-w_{\text{im-}}$ (negative). This encourages target class features and discourages contrastive ones, aiming for sharper class separation.

## 3.3   Classification

We follow the classification approach of DataDream. A CLIP ViT-B/16 model is fine-tuned on $n_{real}$ real and $n_{synth}$ synthetic images. For training, each batch $B_i$ contains as many real

| Method | Venue | R | S | Train Free | Captioning Filtering-free | DTD | Food | Pets | SUN | EuSAT | AirC | IN | FLO | Cars | CAL | Avg. Acc. |
|--------|-------|---|---|------------|---------------------------|-----|------|------|-----|-------|------|-----|-----|------|-----|-----------|
| Visual Prompt Tuning [ ] | ECCV22 | ✓ | | | | 66.06 | 86.99 | 91.84 | 70.47 | 92.33 | 36.21 | 69.57 | 90.95 | 69.01 | 95.40 | 66.31 |
| Text Prompt Tuning [ ] | IJCV'22 | ✓ | | | | 70.73 | 83.65 | 89.89 | 72.95 | 87.05 | 45.50 | 67.97 | 97.62 | 81.40 | 95.23 | 79.20 |
| VPT + TPT | arXiv'23 | ✓ | | | | 72.02 | 83.70 | 88.87 | 72.58 | 90.42 | 48.03 | 67.34 | 98.12 | 81.99 | 95.75 | 79.88 |
| Classifier Tuning [ ] | ICLR23 | ✓ | | | | 73.64 | 87.28 | 92.81 | 76.16 | 87.13 | 46.73 | 73.41 | 86.52 | 82.55 | 96.01 | 80.22 |
| DISEF [ ] | arXiv'23 | ✓ | ✓ | ✓ | | 75.36 | 87.11 | 94.33 | 77.43 | 94.31 | 63.89 | 73.94 | 98.85 | 88.63 | 96.94 | 85.08 |
| DataDream [ ] | ECCV24 | ✓ | ✓ | | ✓ | 73.76 | 87.51 | 93.79 | 76.78 | 92.61 | 63.52 | 72.74 | 98.52 | 90.28 | 98.14 | 84.76 |
| DIPSY (ours) | | ✓ | ✓ | ✓ | ✓ | 75.55 | 87.81 | 94.73 | 77.78 | 94.13 | 62.35 | 73.24 | 98.56 | 88.95 | 99.18 | 85.23 |

Table 1: **Performance (% accuracy) on all datasets with 16-shot real data.** Columns "R" and "S" indicate whether the method uses few-shot real data and synthetic data, respectively. Top section shows results from PEFT-based methods using only real data. Bottom section includes methods that generate synthetic data and use it alongside few-shot real data. "Train-free" denotes methods that require no training prior to generation, while "Captioning Filtering-free" refers to those that do not rely on external models for captioning or filtering during the generation pipeline. DataDream results were reproduced using the authors' released code, adapted to use a Stable Diffusion 1.5 backbone for consistency with other methods. Best and second best results are highlighted.

as synthetic images, replicating real images if necessary. The loss for a given batch is a weighted sum of two Cross Entropy (CE) components as:

$$\text{Loss}_i = \lambda \sum_{\substack{j \in B_i \\ j \text{ real}}} \text{CE}\left(\text{CLIP}(\text{Img}_j), T_j^{\text{real}}\right) + (1 - \lambda) \sum_{\substack{k \in B_i, \\ k \text{ synthetic}}} \text{CE}\left(\text{CLIP}(\text{Img}_k), T_k^{\text{synth}}\right) \quad . \quad (10)$$

The first component measures how well CLIP classifies real images ($\text{Img}_j$) against their true labels ($T_j^{\text{real}}$). The second one measures how well the model classifies synthetic images ($\text{Img}_k$) against their intended target labels ($T_k^{\text{synth}}$); this target label is set to be the class of the real image used as the positive prompt during the generation of that synthetic image. $\lambda$ (between 0 and 1) controls the relative importance of learning from real vs synthetic images.

# 4 Experiments

## 4.1 Experimental setup

**Datasets.** Following [36], we use ten benchmarks for few-shot classification, covering a wide range of domains, e.g. textures, objects, scenes, satellite, and fine-grained: Describable Textures Dataset (DTD) [12], Food-101 (Food) [8], Oxford-IIIT Pet Dataset (Pets) [49], SUN397 (SUN) [63], EuroSAT (EuSAT) [23], FGVC-Aircraft (AirC) [44], ImageNet (IN) [55], OxfordFlowers 102 (FLO) [47], StanfordCars (Cars) [38], and Caltech-101 (CAL) [22].
**Implementation details.** We use Stable Diffusion 1.5 (SD 1.5) as the generative backbone. For each dataset, we randomly sample few-shot real images, ensuring that smaller shot configurations are strict subsets of the larger ones (e.g., the 8-shot set is a subset of the 16-shot set) to enable fair comparisons. For a fair comparison, in all configurations we follow the setup from the state of the art [36]. We generate 200 images per class, and we use 50 denoising steps. For the classifier, we use CLIP ViT-B/16 as the base vision-language model and applying LoRA to both the image and text encoders. We set the LoRA rank to 16, $\lambda = 0.8$, and optimize using AdamW optimizer. We tune the guidance scales for both the text and the two image prompts ($w_{\text{text}}, w_{\text{im1}}, -w_{\text{im2}}$) and the learning rate on a per-dataset basis.

| Method | DTD | Food | Pets | SUN | EuSAT | AirC | IN | FLO | Cars | CAL | Average Accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|
| SD1.5 (no adapters) | 74.41 | 87.74 | 94.58 | 77.41 | 92.89 | 63.44 | 73.21 | 98.21 | 88.92 | 98.98 | 84.98 |
| DIPSY (empty text prompt) | 74.42 | 87.60 | 94.57 | 77.42 | 93.58 | 61.68 | 72.68 | 98.53 | 87.29 | 99.16 | 84.69 |
| DIPSY (ours) | 75.55 | 87.81 | 94.73 | 77.78 | 94.13 | 62.35 | 73.24 | 98.56 | 88.95 | 99.18 | 85.23 |

Table 2: **Comparison (% accuracy) between our method and two baselines**: (1) Stable Diffusion 1.5 (SD1.5) without image prompts, and (2) our method without text prompts.

| Number of Classes | DIPSY (ours) | DataDream |
|---|---|---|
| 10 | 1s | 36m |
| 100 | 57s | 6h |
| 1000 | 8m | 61h |

Table 3: **Precompute time comparison.** Runtime for precomputing using DIPSY and DataDream on a single H100 GPU with 16 shots.

| Negative Image Prompt | Class Similarity | Data Augmentation | DTD | Pets | EuSAT | AirC |
|---|---|---|---|---|---|---|
| ✗ | ✗ | ✗ | 75.02 | 94.72 | 93.44 | 59.70 |
| ✗ | ✗ | ✓ | 74.70 | 94.57 | 92.74 | 60.13 |
| ✗ | ✓ | ✗ | Class Sim. only defined for negative. | | | |
| ✗ | ✓ | ✓ | | | | |
| ✓ | ✗ | ✗ | 75.22 | 94.79 | 93.36 | 61.54 |
| ✓ | ✗ | ✓ | 75.36 | 94.70 | 93.23 | 60.49 |
| ✓ | ✓ | ✗ | 74.93 | 94.77 | 93.49 | 61.55 |
| ✓ | ✓ | ✓ | 75.55 | 94.73 | 94.13 | 62.35 |

Table 4: **DIPSY component ablation**: *Negative Image Prompt* introduces contrastive guidance; *Class Similarity* selects challenging negatives; and *Data Augmentation* adds variability to prompt inputs. All components contribute to improved performance, while the full setup yields the best results.

## 4.2 Comparison to the state of the art

**Quantitative results.** We compare DIPSY with two groups of methods: (i) few-shot parameter efficient fine-tuning (PEFT) approaches that use no synthetic data for tuning, and (ii) methods that generate synthetic data and use them, along with few-shot real data, to finetune. For consistency, we report baseline results (VPT, TPT, Classifier Tuning, and DISEF), taken directly from DISEF [13], where all methods are evaluated under the same few-shot setting. We reproduce DataDream using its official implementation, adapted to SD1.5.

Tab. 1 reports the top-1 classification accuracy of all methods in all 10 datasets. DIPSY achieves the highest average accuracy overall, despite being entirely training-free and not relying on external captioning and filtering. It establishes new SOTA performance on 5 datasets and ranks within the top-2 on 8, including challenging benchmarks such as the finegrained DTD and the satellite imagery EuroSAT. In the remaining 2 datasets, DIPSY results in competitive performance. Our method shows relative limitations on datasets like FGVC-Aircraft, where the high inter-class similarity and the limited discriminative capacity of the diffusion backbone drop its performance. In contrast, the training-based DataDream benefits from learning class-specific features, thus better separating visually similar categories. Overall, DIPSY demonstrates strong generalization and competitive accuracy across diverse visual domains, without the need for fine-tuning or extra supervision.

**Qualitative results.** Comparing DIPSY with DISEF and DataDream on challenging, visually similar class pairs (Fig. 1) demonstrates its superior ability to capture critical discriminative features and generate high-fidelity images. For instance, in the Pets dataset, DIPSY accurately renders distinct eye colors (e.g., orange for British Shorthairs, green-gray for Russian Blues), details DISEF inverts and DataDream entirely misses. In Food101, DIPSY not only successfully differentiates paella from risotto by highlighting typical seafood toppings

| Method | Prep Time (1000 classes) | Gen. Time per Image | Gen. Time per Class | Denoising Steps |
|---|---|---|---|---|
| DISEF [13] | 0 | 0.80s | 160s | 15 |
| DataDream [56] | 61h | 1.36s | 271s | 50 |
| DIPSY (single IP-Adapter) | 0 | 2.59s | 518s | 50 |
| DIPSY (ours) | 8m | 3.91s | 781s | 50 |

Table 5: **Pipeline efficiency comparison for 1000 classes.** While DIPSY requires additional forward passes per diffusion step, it eliminates training overhead and external dependencies. Single IP-Adapter version offers faster generation while maintaining training-free operation. Experiments run on H100 GPU.

where other methods produce ambiguous dishes but also, as further illustrated for Food101 classes like Cupcakes and Steak (Fig. 3), captures defining visual characteristics and maintains aesthetic consistency by generating realistic textures and compositions. Furthermore, for aircraft, DIPSY captures the Boeing 747-400's characteristic upper-deck bump, a key structural cue absent in DISEF's generation. These examples underline DIPSY's robustness in producing class-distinctive, visually plausible synthetic samples crucial for fine-grained, few-shot regimes.

**Efficiency.** To provide a comprehensive efficiency analysis, we present two comparisons that reveal the trade-offs between different approaches. Tab. 3 shows the precompute time required by DIPSY and DataDream, demonstrating our method's scalability advantage. Since DIPSY is entirely training-free, the only computation involved is building a similarity matrix between classes, a lightweight operation. In contrast, DataDream requires training a separate LoRA module for each class, resulting in significantly higher computational cost.

A pipeline comparison 5 shows that while DIPSY's dual IP-Adapter requires more forward passes, it eliminates training overhead and external dependencies. In contrast, DISEF is faster due to fewer noising steps, but this reduces the variability of generated images. The single IP-Adapter version of DIPSY offers a faster, training-free alternative with competitive performance as seen in Tab. 4 (line 2). This makes DIPSY more practical for immediate deployment, avoiding training failures and external dependencies despite a higher per-image computational cost.

## 4.3 Analysis and ablations

**Impact of conditioning**. Tab. 2 presents a comparison between DIPSY and two ablation baselines: SD1.5 without image prompts, and DIPSY with an empty text prompt. The complementary role of multimodal conditioning in DIPSY consistently improves results, enabling better alignment with class semantics and visual characteristics.

**Ablation of DIPSY components.** An ablation study on DIPSY's components, detailed in Tab. 4, confirms the value of our positive-negative guidance. This strategy pairs a random positive image prompt from the target class with a negative prompt from a visually similar class, determined by CLIP feature similarities. The model is consequently pushed to learn subtle, discriminative features. While the single IP-Adapter with only positive guidance is effective, adding a negative prompt signficantly enhances inter-class feature discrimination and boosts performance, especially for fine-grained tasks.
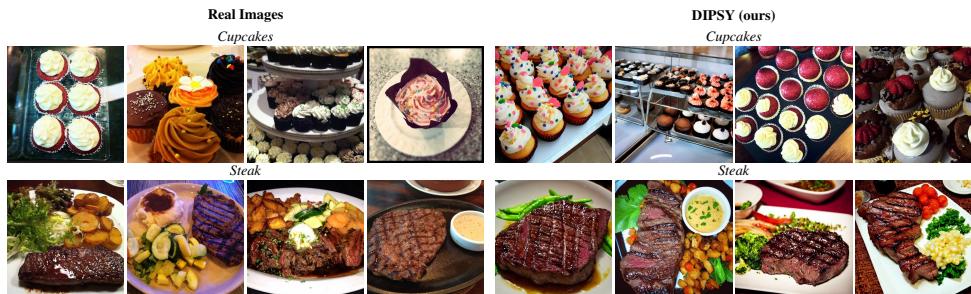
Figure 3:  Qualitative comparison of real and DIPSY generated synthetic images from the classes Cupcakes and Steak from the Food101 dataset.
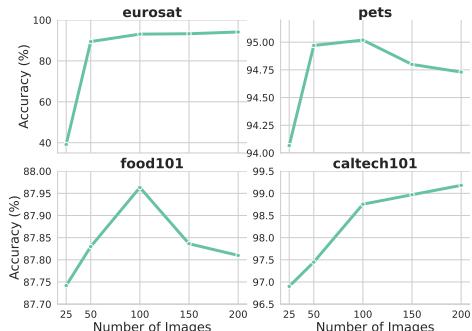


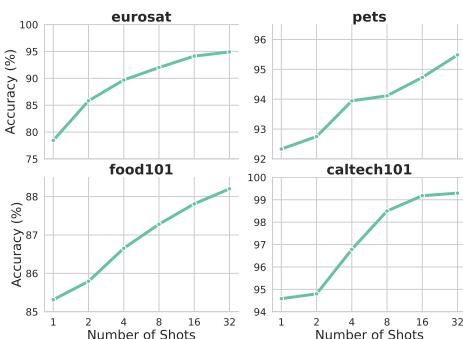Figure 4:  Varying the number of generated images per class.

Figure 5:  Varying the number of real few-shot images per class.

**Ablations of the number of generated and the few-shot real images.**    We study how the number of generated and few-shot real images affects performance (Figs. 4 and 5). Accuracy generally improves with more generated data, with 200 images per class providing a good balance of performance and computational cost. For Food101, the accuracy drop beyond 100 images is likely due to its high intra-class variability, where extra samples deviate from the real test distribution, leading the classifier to overfit synthetic-specific cues. We then test the impact of the number of real images used for generation. While even one shot yields acceptable accuracy, performance improves with more examples, and 16 shots offer a strong trade-off between data efficiency and generalization.

# 5    Conclusion

We proposed DIPSY, a novel training-free approach to few-shot image classification through dual image prompting with positive-negative guidance. By extending classifier-free guidance for independent control of image conditioning and implementing a class similarity-based sampling strategy, our method achieves competitive performance across ten benchmark datasets without requiring model fine-tuning, captioning, or filtering as modern methods do. Future work includes extending DIPSY to video and 3D domains, and applying it to more complex tasks, such as few-shot object detection and segmentation.

# Acknowledgements

# References

[1] Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G. Baraniuk. Self-consuming generative models go mad. In *ICLR*, 2024.

[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022.

[3] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023.

[4] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

[5] Hritik Bansal and Aditya Grover. Leaving reality to imagination: Robust classification via generated datasets. *arXiv preprint arXiv:2302.02503*, 2023.

[6] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. One-shot unsupervised domain adaptation with personalized diffusion models. In *CVPR-W*, 2023.

[7] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. Collaborating foundation models for domain generalized semantic segmentation. In *CVPR*, 2024.

[8] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *ECCV*, 2014.

[9] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *ICLR*, 2019.

[10] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023.

[11] Max F. Burg, Florian Wenzel, Dominik Zietlow, Max Horn, Osama Makansi, Francesco Locatello, and Chris Russell. Image retrieval outperforms diffusion models on data augmentation. *IEEE Transactions on Machine Learning Research*, 2023.

[12] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014.

[13] Victor G Turrisi da Costa, Nicola Dall'Asen, Yiming Wang, Nicu Sebe, and Elisa Ricci. Diversified in-domain synthesis with efficient fine-tuning for few-shot classification. *arXiv preprint arXiv:2312.03046*, 2023.

[14] Lucas Degeorge, Arijit Ghosh, Nicolas Dufour, David Picard, and Vicky Kalogeiton. How far can we go with imagenet for text-to-image generation? *arXiv preprint arXiv:2502.21318*, 2025.

[15] Thanos Delatolas, Vicky Kalogeiton, and Dim Papadopoulos. Studying image diffusion features for zero-shot video object segmentation. In *CVPR-W*, 2025.

[16] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[17] Anh-Dung Dinh, Daochang Liu, and Chang Xu. Compress guidance in conditional diffusion sampling. *arXiv preprint arXiv:2408.11194*, 2024.

[18] Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. In *ICLR*, 2019.

[19] Nicolas Dufour, David Picard, and Vicky Kalogeiton. Scam! transferring humans between images with semantic cross attention modulation. In *ECCV*, 2022.

[20] Nicolas Dufour, Victor Besnier, Vicky Kalogeiton, and David Picard. Don't drop your samples! coherence-aware training benefits conditional diffusion. In *CVPR*, 2024.

[21] Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E. Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. In *NeurIPS*, 2023.

[22] Li Fei-Fei, Robert Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

[23] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2023.

[24] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *NeurIPS*, 2014.

[25] Hasan Abed Al Kader Hammoud, Hani Itani, Fabio Pizzati, Philip Torr, Adel Bibi, and Bernard Ghanem. Synthclip: Are we ready for a fully synthetic clip training? *arXiv preprint arXiv:2402.01832*, 2024.

[26] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *ICLR*, 2023.

[27] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *ICLR*, 2023.

[28] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019.

[29] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[30] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arxiv:2006.11239*, 2020.

[31] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

[32] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022.

[33] Chen Jin, Ryutaro Tanno, Amrutha Saseendran, Tom Diethe, and Philip Alexander Teare. An image is worth multiple words: Discovering object level concepts using multi-concept prompt learning. In *ICML*, 2024.

[34] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *CVPR*, 2023.

[35] Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. In *NeurIPS*, 2024.

[36] Jae Myung Kim, Jessica Bader, Stephan Alaniz, Cordelia Schmid, and Zeynep Akata. Datadream: Few-shot guided dataset generation. In *ECCV*, 2024.

[37] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2022.

[38] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV-W*, 2013.

[39] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, 2023.

[40] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *NeurIPS*, 2024.

[41] Xiaoxia Liang, Ming Zhang, Guojin Feng, Duo Wang, Yuchun Xu, and Fengshou Gu. Few-shot learning approaches for fault diagnosis using vibration data: a comprehensive review. *Sustainability*, 2023.

[42] Shaobo Lin, Kun Wang, Xingyu Zeng, and Rui Zhao. Explore the power of synthetic data on few-shot object detection. In *CVPR-W*, 2023.

[43] Jiamin Lu, Song Zhang, Shili Zhao, Daoliang Li, and Ran Zhao. A metric-based few-shot learning method for fish species identification with limited samples. *Animals*, 2024.

[44] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[45] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2022.

[46] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021.

[47] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

[48] Eva Pachetti and Sara Colantonio. A systematic review of few-shot learning in medical imaging. *Artificial intelligence in medicine*, 2024.

[49] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012.

[50] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *ICLR*, 2024.

[51] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICLR*, 2021.

[53] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[54] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023.

[55] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[56] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

[57] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *CVPR*, 2023.

[58] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *CVPR-W*, 2023.

[59] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[60] Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. In *NeurIPS*, 2023.

[61] Xi Wang, Nicolas Dufour, Nefeli Andreou, Marie-Paule Cani, Victoria Fernandez Abrevaya, David Picard, and Vicky Kalogeiton. Analysis of classifier-free guidance weight schedulers. *IEEE Transactions on Machine Learning Research*, 2024.

[62] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys*, 2020.

[63] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[64] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arxiv:2308.06721*, 2023.

[65] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022.

[66] Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data. *arXiv preprint arXiv:2305.15316*, 2023.

# Supplementary Material

This appendix provides supplementary materials for our paper. We include: (1) a detailed pseudocode for our proposed DIPSY synthetic image generation process and (2) additional qualitative results demonstrating the visual fidelity of generated images across various datasets and classes. These materials aim to offer greater insight into our methodology.

# A   Algorithm Details

The core of our proposed DIPSY method involves a pipeline for generating synthetic images that are both class-representative and discriminative. This process leverages dual IP-Adapter guidance, conditioning on a positive image prompt from the target class and a negative image prompt from a similar class, alongside a text prompt. Alg. 1 provides a step-by-step breakdown of this generation process.

# B   Additional Qualitative Results

To further illustrate the visual fidelity and class-specificity of images generated by DIPSY, this section presents additional qualitative examples across diverse fine-grained categories, such as Pets and Flowers. As can be observed in the comparisons (Figs. 6 and 7), the synthetic images demonstrate a high degree of realism, closely resembling genuine photographs from the respective datasets. DIPSY effectively captures the defining visual characteristics and intricate details specific to each target class. For instance, pet breeds exhibit their distinct features and fur patterns, and flower varieties display appropriate petal structures and natural coloration. Moreover, the generated images generally exhibit an overall aesthetic distribution consistent with the original, real-world examples for each class. These examples show DIPSY's capability to synthesize visually plausible and class-consistent data, suitable for augmenting training sets in few-shot learning scenarios.

---

**Algorithm 1** DIPSY: Core Synthetic Image Generation

---

1: **procedure** DIPSYGENERATION(RealFewShotDataset, $n_{synth}$, $w_{\text{text}}, w_{\text{im+}}, w_{\text{im-}}$)
2:    **Input:**
3:      RealFewShotDataset:  Small set of labeled real images for each class.
4:      $n_{synth}$:  Number of synthetic images to create per class.
5:      $w_{\text{text}}$:  Guidance scale for the text prompt.
6:      $w_{\text{im+}}$:  Guidance scale for the positive image prompt.
7:      $w_{\text{im-}}$:  Guidance scale for the negative image prompt.

8:    **Output:**  SyntheticDataset

9:    SyntheticDataset $\leftarrow \emptyset$
10:    **for all** class $C_+$ in RealFewShotDataset **do**
11:      **for** $i = 1$ **to** $n_{synth}$ **do**
12:          // Step 1:  Select Guiding Images from Few-Shot Examples
13:          $I_+ \leftarrow$ Randomly pick one real image from $C_+$.
14:          $C_- \leftarrow$ Probabilistically sample a class from AllClasses (excluding $C_+$)
                 where classes more similar to $C_+$ are more likely to be chosen,
                 based on CLIP similarity of few-shot images
15:          $I_- \leftarrow$ Randomly pick one real image from $C_-$.

16:          // Step 2:  Prepare Prompts and Augment Images
17:          TextPrompt $\leftarrow$ "A photo of a " + name of $C_+$ *(dataset-specific formulation)*.
18:          $I_+ \leftarrow$ Augment($I_+$)                     ▷ e.g., random crop, rotation
19:          $I_- \leftarrow$ Augment($I_-$)

20:          // Step 3:  Generate Image using Dual IP-Adapter Guidance
21:          NewImage $\leftarrow$ DiffusionModel.Generate(
             text_condition $\leftarrow$ TextPrompt,
             positive_image_condition $\leftarrow I_+$,
             negative_image_condition $\leftarrow I_-$,
             text_guidance_scale $\leftarrow w_{\text{text}}$,
             positive_image_guidance_scale $\leftarrow w_{\text{im+}}$,
             negative_image_guidance_scale $\leftarrow w_{\text{im-}}$
            )
22:          Add (NewImage, $C_+$) to SyntheticDataset.

23:      **end for**
24:    **end for**
25:    **return** SyntheticDataset
26: **end procedure**

---

Figure 6: Qualitative comparison of real and DIPSY generated synthetic images from the classes English Cocker Spaniel, Persian and Beagle from the Oxford-IIIT Pet dataset. DIPSY generates realistic, class-specific images that capture breed characteristics like coat patterns and facial structures.



Figure 7: Qualitative comparison of real and DIPSY generated synthetic images from the classes Siam Tulip, Rose and Yellow Iris from the Oxford Flowers 102 dataset. The DIPSY generated images accurately represent the unique features, colors, and forms of each flower class, closely resembling the real images.