A Minimization of KL-Divergence Objective in Diffusion Probabilistic Model

Consider the training objective of the diffusion model, minimization of KL-divergence between the joint distributions of forward and backward process q and $p_{\boldsymbol{\theta}}$ over all time steps:

$$\mathrm{KL}\left(q\left(\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_T\right) \| p_{\boldsymbol{\theta}}\left(\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_T\right)\right) \tag{10}$$

By definition of KL-divergence and the property of Markovian transition, we can further decompose the expression in time (Yang et al., 2022):

$$\mathrm{KL}\left(q \| p_{\boldsymbol{\theta}}\right)=\mathbb{E}_q\left[-\log p\left(\boldsymbol{x}_T\right)-\sum_{t=1}^T \log \frac{p_{\boldsymbol{\theta}}\left(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t\right)}{q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}\right)}\right]+\text{const} \tag{11}$$

Using Jensen's inequality, we can derive:

$$\mathrm{KL}\left(q\left(\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_T\right) \| p_{\boldsymbol{\theta}}\left(\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_T\right)\right) \geq -\log p_{\boldsymbol{\theta}}\left(\boldsymbol{x}_0\right)+\text{const} \tag{12}$$

The objective in training is to maximize the variational lower bound (VLB) of the log-likelihood of the data $\boldsymbol{x}_0$, which is equivalent to minimizing the negative VLB as in Eq. (1).

B Training Data Synthesis for Supervised Learning

Consider the following formulation of supervised learning (Sutskever, 2023; Cunningham et al., 2008):

$$\mathrm{Pr}_{S\sim D^{|S|}}\left[\mathrm{Test}_D(f)-\mathrm{Train}_S(f) \leq \sqrt{\frac{\log|\mathcal{F}|+\log 1/\delta}{|S|}} \forall f \in \mathcal{F}\right]>1-\delta \tag{13}$$

where D denotes the distribution of the whole target dataset and S is a training subset of D. $\mathcal{F}$ represents all potential functional space, which can be represented by a neural network with a fixed structure, where each unique combination of parameter values corresponds to a unique f. The training and test errors are defined as $\mathrm{Train}_S(f)=\frac{1}{|S|}\sum_{(X_i,Y_i)\in S} L\left(f\left(X_i\right), Y_i\right)$ and $\mathrm{Test}_D(f)=\frac{1}{|D|}\sum_{(X_i,Y_i)\in D} L\left(f\left(X_i\right), Y_i\right)$, respectively. Mathematically, it is evident that when the training loss is minimal and the training set cardinality significantly exceeds the model parameters, the test loss will also be notably low.

We now prove this proposition. Consider the formulation of test error:

$$\begin{aligned}
\mathrm{Pr}_{S\sim D^{|S|}}\left[\mathrm{Test}_D(f)-\mathrm{Train}_S(f) \geq t \text{ for some } f \in \mathcal{F}\right] &\leq \sum_{f\in\mathcal{F}} \mathrm{Pr}_{S\sim D^{|S|}}\left[\mathrm{Test}_D(f)-\mathrm{Train}_S(f) \geq t\right] \\
&= \sum_{f\in\mathcal{F}} \mathrm{Pr}_{\sim D^{|S|}}\left[\frac{1}{|S|}\sum\left(L_i-\mathbb{E}[L]\right) \geq t\right] \\
&\leq |\mathcal{F}| \exp\left(-2|S|t^2\right)
\end{aligned} \tag{14}$$

The first step is a straightforward extension of the original formulation. The second step employs Hoeffding's inequality (Hoeffding, 1994). Given a series of independent random variables $X_1, X_2, \ldots, X_n$ bounded by $a_i \leq X_i \leq b_i$, Hoeffding's inequality states:

$$\mathrm{Pr}\left[\sum_{i=1}^n X_i-\mathbb{E}\left[\sum_{i=1}^n X_i\right] \geq t\right] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n\left(b_i-a_i\right)^2}\right) \tag{15}$$

It bounds the probability that the sum of independent bounded random variables deviates from its expected value beyond a set limit. Consider each $L_i-\mathbb{E}[L]$ as a bounded random variable, capturing the deviation of a specific data point's loss from its expected value. By assuming a bounded loss function L_i, which is valid as we can project any unbounded loss function monotonically to a bounded range of $[0,1]$ through sigmoid function, we obtain the upper bound on the probability of this disparity surpassing threshold t, culminating in the expression $|\mathcal{F}| \exp\left(-2|S|t^2\right)$. This implies the convergence of test error to zero when the size of the training dataset goes to infinite:

$$\lim_{|S|->\infty} \Pr\left[\sup_{f\in\mathcal{F}} \left(\text{Test}_D(f) - \text{Train}_S(f)\right) > \varepsilon\right] = 0 \tag{16}$$

which eventually validates our statement on the implication of training data synthesis.

C Maximum Mean Discrepancy (MMD) in RKHS

The Maximum Mean Discrepancy (MMD) in a Reproducing Kernel Hilbert Space (RKHS) (Aronszajn, 1950) $\mathcal{H}$ provides a metric to measure the difference between two probability distributions.

$$\text{MMD}[\mathcal{F}, p, q] = \sup_{\|\psi_\vartheta\|_\mathcal{H} \le 1} \left(\mathbb{E}_q\left[\psi_\vartheta(\mathcal{T})\right] - \mathbb{E}_p\left[\psi_\vartheta(\mathcal{S})\right]\right) \tag{17}$$

$$\text{MMD}[\mathcal{F}, p, q] = \sup_{\|\psi_\vartheta\|_\mathcal{H} \le 1} \langle \mu_\mathcal{T} - \mu_\mathcal{S}, \psi_\vartheta \rangle_\mathcal{H} \tag{18}$$

$$\text{MMD}^2[\mathcal{F}, p, q] = \left[\sup_{\|\psi_\vartheta\|_\mathcal{H} \le 1} \langle \mu_\mathcal{T} - \mu_\mathcal{S}, \psi_\vartheta \rangle_\mathcal{H}\right]^2 \tag{19}$$

$$\text{MMD}^2[\mathcal{F}, p, q] = \|\mu_\mathcal{T} - \mu_\mathcal{S}\|^2_\mathcal{H} \tag{20}$$

In Eq. (17), the MMD between distributions p and q is formulated as the supremum of the difference of expectations under these distributions, constrained by the norm of the function ψ_ϑ in the RKHS. This difference can be further expanded using the inner product in the RKHS, leading to an expression in terms of the mean embeddings of the distributions, as shown in Eq. (18). Squaring the MMD gives us a more interpretable metric, which is the squared distance between the mean embeddings of the two distributions in the RKHS, as depicted in Eq. (19) and Eq. (20). This leads to the derivation from Eq. (5) to Eq. (6) in Sec. 3.1.

D Dataset Details

We adopt seven datasets in total across different settings: ImageNette (IN-10) (Howard, 2019), ImageNet100 (IN-100) (Tian et al., 2020), and ImageNet1K (IN-1K) (Deng et al., 2009), CUB (Wah et al., 2011), Cars (Krause et al., 2013), PET (Parkhi et al., 2012), EuroSAT (Helber et al., 2018). The detailed dataset statistics are specified in Tab. 5.

Table 5: Detailed Dataset Statistics.

	IN-10	IN-100	IN-1K	CUB	Cars	PETS	EuroSAT
Dataset Size	14197	141971	1419712	11788	16,185	7349	27000
Training Data Size	12811	128116	1281167	5994	8144	3680	21600
Test Data Size	1000	10000	100000	5794	8041	3669	5400
No. Classes	10	100	1000	200	196	37	7
Domain	Natural Image	Natural Image	Natural Image	Fine-grain Bird	Fine-grain Cars	Pet Breed	Satellite Image

E Training Data Synthesis Details

We use Stable Diffusion v1.5 (Rombach et al., 2022) across all benchmarks. Unless specified (i.e. we specifically scale up synthetic data in Sec. 4.3), we generate an equal amount of synthetic data samples as in the corresponding real dataset for both replacing and augmenting real training data. The stable diffusion generation parameters are specified in Tab. 6. We use text prompt mentioned in Sec. 3.2, of the form "`photo of [classname], [Image Caption], [Intra-class Visual Guidance]`" for ImageNet dataset (IN-10, IN-100, IN-1K); for the rest four datasets, we do not use the image caption generated with BLIP2 (Li et al., 2023). Also, all generations use a uniform set of negative prompts "`distorted, unrealistic, blurry, out of frame`".

Table 6: Hyperparameters used in training data synthesis.

Model	Sampling steps	Scheduler	Guidance scale	Img_strength	Image size
Stable Diffusion v1.5	30	UniPC	2.0	0.75	512 × 512

For fine-tuning the Stable Diffusion, we use Low-Rank Adaptation (LoRA) (Hu et al., 2021) with the MMD augmented objective Sec. 3.1 and inject the visual guidance Sec. 3.2. The fine-tuning hyperparameters used are specified in Tab. 7. We use the loss function mentioned in Sec. 3.1, where

Table 7: Hyperparameters used in LoRA fine-tuning

Rank	Epoch	Leanring Rate	Batch Size	SNR-γ	Image size
Stable Diffusion v1.5	300	1e-4	8 × 8	0.75	512 × 512

we augment the MMD distribution matching loss with the simple diffusion model training objective as in Eq. (21), we we choose $\gamma = 0.05$.

$$L_{overall} = L_{DM} + L_{simple} = \frac{1}{|\mathcal{N}|} \sum_{i=1}^{|\mathcal{N}|} || \left(\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon}_{\boldsymbol{\theta}} \left(\boldsymbol{x_t}, \boldsymbol{t}\right)\right) ||_{\mathcal{H}}^2 + \gamma || \frac{1}{|\mathcal{N}|} \sum_{i=1}^{|\mathcal{N}|} \left(\boldsymbol{\epsilon}_0 - \boldsymbol{\epsilon}_{\boldsymbol{\theta}} \left(\boldsymbol{x_t}, \boldsymbol{t}\right)\right) ||_{\mathcal{H}}^2 \tag{21}$$

F Downstream Image Classification Training Details

We adopt ResNet-50 (He et al., 2016) through all benchmarks in the image classification task. For all three ImageNet datasets, we train the classifier model from scratch. Follow the training recipe of (Lei et al., 2023) and (Wightman et al., 2021), the training hyperparameters are specified in Tab. 8

G Experiment Setting Details and More Results

G.1 Image Classification With Synthetic Data Only

We compare with the baselines of previous works on synthetic ImageNet data for supervised image classification. All baselines synthesize equal amounts of data as real datasets. The accuracies reported are from the original baseline papers, except for FakeIt (Sarıyıldız et al., 2023) in CUB (Wah et al., 2011), Cars (Krause et al., 2013), PET (Parkhi et al., 2012), and EuroSAT (Helber et al., 2018). The number reported is based on our implementation of the method with the base prompt `"a photo/image of [classname]"`.

Note that there is a difference in the generation resolution among different methods. In particular BigGAN-deep (Brock et al., 2018), VQ-VAE-2 (Razavi et al., 2019), and CDM (Ho et al., 2022), as reported in (Azizi et al., 2023), use image resolution of 256 × 256. CiP (Lei et al., 2023), FakeIt (Sarıyıldız et al., 2023), and our method uses image resolution of 512 × 512. Imagen (Azizi et al., 2023) uses image resolution of 1024 × 1024. Even though we resize all samples in downstream training to 224 × 224, the generation resolution still has influences as reported in (Azizi et al., 2023) potentially due to the generation details and quality. Thus, comparing our method and non-Stable Diffusion baselines can potentially be an unfair comparison. However, our main focus in training data synthesis is in comparison with real data and the comparison between ours and other Stable Diffusion-based method show our framework leads to improved synthetic data informativeness to close the gap between real and synthetic data.

G.2 Augmenting Real Data With Synthetic Data

For CUB (Wah et al., 2011), Cars (Krause et al., 2013), and PET (Parkhi et al., 2012) datasets, we use a pre-trained ResNet-50 due to the scarcity of data to train a decent ResNet-50 classifier. For EuroSAT (Helber et al., 2018), ImageNette (IN-10) (Howard, 2019), ImageNet100 (IN-100) (Tian et al., 2020), and ImageNet1K (IN-1K) (Deng et al., 2009) we train the model from scratch. We augment the real data with an equal amount of synthetic data, which effectively raises the amount of training by two.

Table 8: Training Hyperparameters of Different Dataset.

	IN-10	IN-100	IN-1K	CUB	CARS	PETS	EuroSAT
Train Res → Test Res	224 → 224	224 → 224	224 → 224	448 → 448	448 → 448	224 → 224	448 → 448
Epochs	200	200	300	200	200	200	200
Batch size	128 × 8	128 × 8	512 × 8	128 × 8	128 × 8	128 × 8	128 × 8
Optimizer	SGD	SGD	LAMB	SGD	SGD	SGD	SGD
LR	0.1	0.1	5e-3	0.1	0.1	0.1	0.1
LR decay	multistep	multistep	cosine	multistep	multistep	multistep	multistep
decay rate	0.2	0.2	0.2	0.2	0.2	0.2	0.2
decay epochs	50/100/150	50/100/150	-	50/100/150	50/100/150	50/100/150	50/100/150
Weight decay	5e-4	5e-4	0.02	5e-4	5e-4	5e-4	5e-4
Warmup epochs	-	-	5	-	-	-	-
Label smoothing	-	-	0.1	-	-	-	-
Dropout	x	x	x	x	x	x	x
Stoch. Depth	x	x	0.05	x	x	x	x
H. flip	-	-	✓	-	-	-	-
Rand Augment	x	x	7 / 0.5	x	x	x	x
Mixup alpha	x	x	0.2	x	x	x	x
Cutmix alpha	x	x	1.0	x	x	x	x
CE loss → BCE loss	x	x	✓	x	x	x	x
Mixed precision	✓	✓	✓	✓	✓	✓	✓

G.3 Generalization To Out-of-Distribution Data

We train the image classifier on synthetic in-distribution synthetic and real ImageNet-1K and then test on four Out-of-Distribution test sets: (1) ImageNet-v2 (IN-v2) (Recht et al., 2019), which is a different ImageNet test set collected on purpose to eliminate the effect of adaptive overfitting for original ImageNet test set; ImageNet-Sketch (IN-Sketch) (Wang et al., 2019), which contains sketch of ImageNet classes with black and white scheme only; ImageNet-R (IN-R) (Hendrycks et al., 2021a), which contains renditions of ImageNet classes such as art, cartoon, graphics, etc.; and ImageNet-A (IN-A) (Hendrycks et al., 2021b), which contain natural adversarial examples for ResNet models trained with ImageNet-1K training set. In particular, ImageNet-A and ImageNet-R only contain 200 classes as a subset of ImageNet-1K, thus we only consider the output logits of the image classifier over these sub-classes.

G.4 Privacy Analysis

Membership Attack In this study, we implement a membership inference attack, which allows an adversary to determine whether a specific example belongs to the training set of a trained model. To achieve this, we adopt the Likelihood Ratio Attack (LiRA) as the state-of-the-art method, as proposed by Carlini et al. Carlini et al. (2022). Our objective is to evaluate the performance of this attack by conducting experiments on two different training approaches: the privacy training dataset and synthetic data. Specifically, we focus on the low false-positive rate regime, where the attack is expected to be more effective. To begin, we train a total of 16 shadow models using random samples from the original IN-10 (ImageNette) dataset. Half of these models are trained on a target point (x, y), while the other half are not. This division allows us to create both IN and OUT models for the target point (x, y). By fitting two Gaussian distributions to the confidences of the IN and OUT models, measured in logit scale, we are able to capture the characteristics of these models. Next, we query the confidence of the target model f on the target point (x, y). Using a parametric Likelihood-ratio test, we compare this confidence value with the distributions obtained from the IN and OUT models. Based on this comparison, we can infer whether the target point (x, y) is likely to be a member of the training set or not. In order to train the shadow models, we utilize the ResNet50 architecture. We always begin by finetuning the diffusion model using the sampled data. Once the finetuning is complete, we generate synthetic data of equal size to the original dataset. Subsequently, we train the shadow models using this synthetic data. As a result, we have a total of 16 finetuned diffusion models and 16 ResNet50 models for our experiments. By implementing the membership inference attack and evaluating its performance on both the privacy training dataset and synthetic data, we aim to provide insights into the vulnerability of trained models to such attacks. The results of our experiments shed light on the effectiveness of the Likelihood Ratio Attack in the low false-positive rate regime and contribute to the ongoing research on model privacy and security.

Visual Appearance In this study, we build upon the prior research conducted by Zhang et al. (Zhang et al., 2023) and Somepalli et al. (Somepalli et al., 2023) and utilize the Self-Supervised Content

Duplication (SSCD) method proposed by Pizzi et al. (Pizzi et al., 2022) for content plagiarism detection. SSCD is a self-supervised approach that is based on SimCLR (Chen et al., 2020) and incorporates InfoNCE loss (Oord et al., 2018), entropy regularization of latent space representation, and various data augmentation techniques. For the experiment, we use the ResNeXt101 model trained on the DISC dataset (Douze et al., 2021) provided by the official `https://github.com/facebookresearch/sscd-copy-detection/tree/main` as our detection model. To train our model, we employ the ResNeXt101 architecture and train it on the DISC dataset (Douze et al., 2021). We first extract 1024-dimensional feature vectors for both the query image and all reference images using the trained detection model. These feature vectors are then normalized using L2 normalization. By computing the inner product between the query feature vector and each reference feature vector, we obtain a similarity score, which indicates the degree of similarity between the query and reference images. Based on these similarity scores, we select the top five reference features that exhibit the highest similarity with the query feature. These selected features allow us to retrieve the corresponding images from the dataset. To further evaluate the similarity between the query and reference images, we rank the similarity scores and present the top 3 images along with their corresponding real data.

G.5 Further Ablation Study Analysis

In Sec. 4.1, we conducted an ablation study on three key techniques: (1) Latent Prior, (2) Visual Guidance, and (3) Distribution Matching with MMD loss. Note that, *Visual Guidance* and *Distribution Matching* require finetuning on Stable Diffusion with either simple diffusion model loss Ho et al. (2020) if Distribution Matching not adopted or otherwise augmented distribution matching loss. *Latent Prior* can be adopted with or without finetuning on Stable Diffusion. This section provides a more detailed analysis of the effectiveness of each module.

Latent Prior The application of Latent Prior demonstrates significant performance improvements. Specifically, implementing Latent Prior results in top-1 accuracy enhancements of 8.2% and 10.3% on ImageNette and ImageNet-100, respectively, when compared to the baseline. This underscores the efficacy of Latent Prior in our framework. However, even without Latent Prior, Visual Guidance and Distribution Matching Loss also contribute to performance enhancements. Without Latent Prior, these methods lead to improvements of 1.1% and 1.8% on ImageNet-10 and ImageNet-100, respectively. When integrated with Latent Prior, the improvement further increases to 2.5% and 3.7% on these datasets.

Visual Guidance For Visual Guidance, we employ the ”[intra-class Visual Guidance]” token in addition to the base prompt template ”photo of [classname], [Image Caption]”. We achieve substantial performance gains with the inclusion of Visual Guidance, leading to improvements of 1.5% and 0.7% on ImageNet-10, and 0.6% and 0.2% on ImageNet-100, respectively, when combined with the Latent Prior. This demonstrate the benefit of having detailed image features along with textual description for informative training data synthesis.

Distribution Matching Loss Motivated by the known limitations of diffusion loss's looseness, as discussed in Sec. 3.1, we further include MMD loss to have a tigher bound over the distribution discrepancy between real and synthetic data. The results demonstrate that the addition of MMD loss brings about consistent improvements. Specifically, on top of the vanilla diffusion loss, the MMD loss contributes an increase of 0.4%/0.5% on ImageNette and ImageNet-100. When combined with Visual Guidance, these improvements are 0.6%/1.1%, and with both Latent Prior and Visual Guidance, they are 1.0%/0.7%, respectively.

G.6 Multi-Seed Experiment

To examine the stability of our proposed method, we deliver a three-seed multi-run experiment on the ImageNet (IN-10, 1N-100, IN-1K) dataset, following the same training recipe as in Tab. 6. As demonstrated in Tab. 9, we observe that for small-scale dataset (IN-10), the variability is larger compared to larger datasets (IN-100, IN-1K), but for all settings, variation remains within an acceptable range and maintains a substantial margin against the baseline method.

Table 9: **Multi-Seed Experiment Results**. Top-1 Classification Accuracy across various datasets (IN-10, IN-100, IN-1k) over multiple runs. *Average* shows the mean and standard deviation of results of multiple runs.

	IN-10	IN-100	IN-1K
Run 1	90.5	80.0	70.9
Run 2	91.2	79.9	70.6
Run 3	90.7	80.2	70.9
Average	90.8 ± 0.29	80.0 ± 0.12	70.8 ± 0.14

G.7 More Complex Human Face Dataset

To further explore the applicability of our proposed framework, we conduct experiments on more complex tasks involving human faces. The challenges involved with the task on the human face usually come with more fine-grained facial attributes and natural demographic bias with spurious correlation in data distribution. Thus, we leverage the CelebA Liu et al. (2015) dataset under a general facial attribution classification task. Consistent with other experiments, we synthesize training data with Stable Diffusion 1.5 and train a ResNet50 from scratch for 30 epochs.

Facial Attribute Classification One of the primary tasks related to human face recognition is attribute classification. We conduct this on the CelebA dataset, with three different attributes for prediction. For synthesizing training data, the `"A [Attribute Name] person"` serves as the base prompt template. We select a subset of CelebA dataset and formulate three binary classifications of *Smiling*, *Attractive*, and *Heavy Makeup*. As shown in Tab. 10, for concert facial attribute like *Smiling* and *Heavy Makeup*, we observe synthetic data from our framework can represent facial attribute levels close to the training data. For more subtle concepts like *Attractive*, the distribution shift between synthetic data and real data is larger, leads to slightly less optimal results. In all cases, our synthetic data outperform the baseline by a considerable margin. This validates the capability of our framework to adapt to fine-grained facial features.

Table 10: CelebA Facial Attribute Classification Result.

	Smiling	Attractive	Heavy Makeup
Real Data	89.65	86.22	85.60
Baseline	76.90	65.11	73.35
OURS	86.93	74.74	82.55

Future work in More Complex Face Recognition While we primarily explore the utility of synthetic data in capturing general facial features, delving into the domain of individual human face recognition presents a considerably more challenging task. This complexity arises from the extreme scarcity of individual-specific data and the heightened need for privacy preservation in such contexts. Recognizing the significance and complexity of this area, we identify it as an important avenue for future work.

H Further Discussion

H.1 Significant of Class-likelihood Matching in Fine-grained Classification

In Sec. 4.1, we observe a more significant advantage of our synthetic data over fine-grain classification tasks (i.e. CUB (Wah et al., 2011), Cars (Krause et al., 2013)) than natural image classification (i.e ImageNet (Deng et al., 2009)) in comparison to baseline. This is due to the requirement of more detailed decision boundaries among classes. This highlights the importance of the class-likelihood alignment $q(y|\boldsymbol{x}) = p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$. This also indicates solely increasing in the capacity of the generative model in synthesizing high-quality images is insufficient to handle hard image classification without explicit alignment between the joint distribution between image and annotation.

H.2 Better Out-of-Distribution Generalization Performance

In the assessment of OOD generalization performance Sec. 4.4, we find that with scaling-up ImageNet-1K performance, although the in-distribution performance of training with synthetic data is still inferior to real data, the OOD generalization performance already outperforms the real counterpart at an earlier breaking point (i.e. in ImageNet-R (IN-R) (Hendrycks et al., 2021a) and ImageNet-A (IN-A) (Hendrycks et al., 2021b)). While we synthesize training data to reproduce the target data distribution, the superior OOD generalization for models trained on synthetic data indicates a greater potential for synthetic data in strengthing the OOD generalization performance.