PixelAsParam: A Gradient View on Diffusion Sampling with Guidance

Anh-Dung Dinh [1] **Daochang Liu** [1] **Chang Xu** [1]

Abstract

Diffusion models recently achieved state-of-the-art in image generation. They mainly utilize the denoising framework, which leverages the Langevin dynamics process for image sampling. Recently, the guidance method has modified this process to add conditional information to achieve a controllable generator. However, the current guidance on denoising processes suffers from the trade-off between diversity, image quality, and conditional information. In this work, we propose to view this guidance sampling process from a gradient view, where image pixels are treated as parameters being optimized, and each mathematical term in the sampling process represents one update direction. This perspective reveals more insights into the conflict problems between updated directions on the pixels, which cause the trade-off as previously mentioned. We then investigate the conflict problems and propose to solve them by a simple projection method. The experimental results evidently improve over different baselines on datasets with various resolutions.

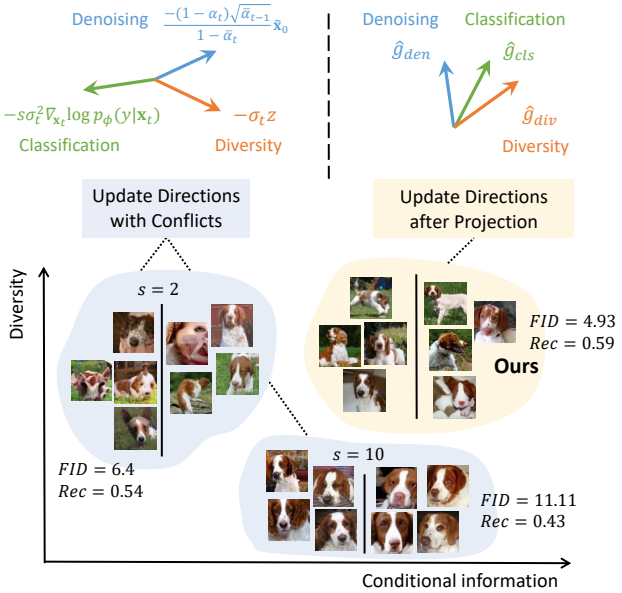


Figure 1. Conditional guidance generation of two classes Welsh Sprinter Spaniel and Brittany Spaniel from ImageNet64x64. The conditional guidance (Dhariwal & Nichol, 2021; Chao et al., 2022), with a small scale, achieving diversity leads to a very good FID score and high recall value. However, images do not achieve good quality, and conditional information is not obtained. In contrast, with a huge classification scale, conditional information is achieved, but the FID score is too high and has a low recall value due to the lack of diversity. Our method simultaneously achieves conditional information and diversity. More samples could be found in Figure 3 and Appendix D.

1. Introduction

Generative models are currently one of the most active research areas in machine learning because of their essential properties in understanding the essence and features of data. Among many generative models (Brock et al., 2018; Karras et al., 2020; Razavi et al., 2019; Karras et al., 2019; Wang et al., 2021; Tran et al., 2018), Diffusion Generative Models (DGMs) (Ho et al., 2020; Nichol & Dhariwal, 2021; Bao et al., 2022) emerge as one of the most potentials for the future of image generation. The main idea of DGMs is to turn an intractable form of data distribution into a tractable form of noise distribution and convert from tractable noise to the original data distributions.

Like conditional GAN-based methods (Long et al., 2018; Kang et al., 2021; Hou et al., 2022), diffusion models also benefit from exploiting class conditional distribution during training and inference process (Batzolis et al., 2021; Sinha et al., 2021; Ho et al., 2020; Ho & Salimans, 2022). However, one of their drawbacks is its exorbitant retraining cost for adding conditional information. Guidance technique (Ho et al., 2020; Song et al., 2020b) allows us to achieve conditional DGMs without retraining the diffusion model. Guidance classifiers in (Dhariwal & Nichol, 2021) are trained as a noise-aware discriminative model to understand images at different noise levels. During the sampling process of the

[1]School of Computer Science, University of Sydney. Correspondence to: Anh-Dung Dinh <dinhanhdung1996@gmail.com>, Chang Xu <c.xu@sydney.edu.au>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

DGMs, guidance signals from the noise-aware model will be injected into the process as conditional information. This guidance method achieves a conditional generative model without fine-tuning or retraining diffusion models.

However, most of the guidance works suffer from the problem of the trade-off between image quality and conditional class (Dhariwal & Nichol, 2021; Ho & Salimans, 2022). In guidance sampling, ADM-G (Dhariwal & Nichol, 2021; Song et al., 2020b) utilizes the classification guidance scale s to control classification information that adds to the sampling process. If s is small, conditional information is not achieved, and the image quality is poor. In contrast, if s is set too large, the conditions are achieved, but the diversity is sacrificed, leading to bad FID score. The problem is visualized in Figure 1 where the vanilla guidance ADM-G (Dhariwal & Nichol, 2021) samples images with two conditional classes. Two guidance scales $s = 2$ and $s = 10$ are selected based on tuning to achieve the best performance for ADM-G.

From the optimization perspective, the trade-off only happens when conflicts exist in different objectives (Sener & Koltun, 2018). Based on that, our intuition about the reason for the trade-off between image quality, conditional class, and diversity is twofold. Firstly, the conditional information contradicts the required diversity, which is reasonable when the conditional information limits the search field to some constrained classes. In contrast, diversity tends to explore as much as possible. Secondly, the classification condition and the objective to generate images into a predefined distribution has some incompatibility problem.

This work aims to formulate the image quality, classification information, and diversity information of the DGMs inference process to make them easy to analyze and manipulate. Instead of optimizing model parameters, our model considers the pixels of the images as parameters that need optimization. Given the pretrained diffusion model and classifier, the guidance signals obtained by the classifier and the output received by the diffusion model are treated as gradients to optimize the original noisy image. The image now turns into an optimizing variable with more than one objective.

From this point of view of the sampling process, we analyze the gradient imposed by each objective and find out the conflicts between them. This helps explain the trade-off we find in previous works (Dhariwal & Nichol, 2021; Chao et al., 2022) where different update directions on an image conflict with each other. Moreover, we then utilize a simple projection technique to reduce conflicts between the pairs of these conflicting objectives, which can generate samples with fewer trade-off effects. In general, we have three main contributions:

- Model the denoising process with classification guidance of the DGMs into an optimization problem where the pixels are considered as parameters. This could be a new perspective for the research community to improve DGMs further.
- Analyse the problem of conflicts to provide further insights into the guidance sampling process.
- Propose a method to alleviate conflicts between update directions. The experimental results show a significant improvement using our proposed scheme.

2. Related works

Diffusion models: This work is primarily based on the framework of the Denoising Diffusion Probabilistic Model (DDPM), and its variants (Ho et al., 2020; Nichol & Dhariwal, 2021; Song et al., 2020a; Wang et al., 2023). Besides the DDPM series, we have the theoretical counterpart score-based models (Song & Ermon, 2019; 2020). These two series of models achieve similar sample quality as GANs. Although these two types of models have different optimization objectives and different motivations, they are proven to be closely related (Song et al., 2020b). Since then, diffusion scheme have become very active in both generative and discriminative applications such as (Saharia et al., 2022; Shan et al., 2023; Chen et al., 2022), which often provides comparable or better results than conventional approaches such as (Qiu et al., 2023; 2022; Wu et al., 2019b).

Conditional generative models: Conditional generative models have received much attention in generative models research (Odena et al., 2017; Gong et al., 2019). GANs often employ a classification head attached to its Discriminator network to learn the classifier conditions (Kang et al., 2021; Gong et al., 2019; Brock et al., 2018; Zhao et al., 2020; Dung & Binh, 2022). The classifier's training takes place at the same time as training GANs. DGMs (Ho et al., 2020; Song et al., 2020b) also offer a conditional version by attaching a classification head to the diffusion models. Similar to GANs, DGMs also benefit from conditional information (Dhariwal & Nichol, 2021). Some other DGMs condition on image to transfer styles between instances (Preechakul et al., 2022).

Dhariwal & Nichol provide a method to achieve a controllable sampler/generator without retraining DGMs. (Liu et al., 2023) generalizes the idea by adapting different types of modality. Classifier-free guidance (Ho & Salimans, 2022) shows that guidance properties can also be achieved without a classifier. CompDiffusion (Liu et al., 2022) proposes to guide the diffusion model by combining different conditions given by a pretrained model. Most guidance works suffer from the trade-off between sample quality, diversity, and conditional information. (Chao et al., 2022) handles the

problem by including a score model during the training of a noise-aware classifier to avoid gradient mismatch, causing an increase in the running time. We deal with this problem without re-training the diffusion or noise-aware classifiers.

Gradient conflicts: Reducing gradient conflicts is a relevent topic in multi-task learning (Vandenhende et al., 2021; Liu et al., 2021a; Sener & Koltun, 2018). (Liu et al., 2021b) alleviates the conflict problem by equalizing the sum of raw gradients projection on individual tasks. (Yu et al., 2020) analyzes the problem of conflicts and concludes on three conditions for conflict: direction conflict, magnitude conflict, and the high curvature between tasks. PCGrad (Yu et al., 2020) mitigates the conflicts by projecting one task over the orthonormal plane of other tasks. GradNorm (Chen et al., 2018) solves the magnitude conflict by proposing a method to normalize the magnitude of all gradients. RotoGrad (Javaloy & Valera, 2021) follows the line of PCGrad (Yu et al., 2020) by using a rotation method to solve the direction conflict and combining it with GradNorm for magnitude conflicts. DGMs sampling is not a multi-task problem, yet this process could still employ the philosophy behind multi-task learning gradient conflicts to mitigate the conflicts between its update directions.

3. Background

DDPM: The Denoising Diffusion Probabilistic Model (DDPM) has the form of $p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T})d\mathbf{x}_{1:T}$ with $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T$ being latent variables sharing the same dimensionality with the data $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ The variable $\mathbf{x}_T$ follows the distribution $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$. The $p_\theta(\mathbf{x}_{0:T})$ is the *reverse process* characterized by the Markov chain:

$$p_\theta := p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \tag{1}$$

where $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$.

This *reverse process* is also known as the diffusion model's sampling process or inference process.

In contrast to the *reverse process*, the *forward process* aims to corrupt the original data $\mathbf{x}_0$ to $\mathbf{x}_T$ with Gaussian noise. This process is a fixed Markov chain with Gaussian noise:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{2}$$

where $q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$. β_t is the fixed variance scheduled before the process starts. We have the $\mathbf{x}_{t-1}$ conditioned on $\mathbf{x}_0$ and $\mathbf{x}_t$ as:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}) \tag{3}$$

Where $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\mathbf{x}_t$ and $\tilde{B}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$ with $\alpha_t = 1 - \beta_t$ and $\bar{\alpha} = \prod_{s=1}^{t} \alpha_s$.

The parameters θ will be optimized via variational bound on negative log-likelihood:

$$\mathbb{E}[-\log p_\theta(\mathbf{x}_0)] \leq \mathop{\mathbb{E}}_{q}[-\log p(\mathbf{x}_T) - \Sigma_{t\geq 1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}] \tag{4}$$

The Eq. 4 can be re-written as:

$$\mathop{\mathbb{E}}_{q}[D_{KL}(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T)) + \sum_{t>1} D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)] \tag{5}$$

The θ is parameters of the noise predictor $\epsilon_\theta(\mathbf{x}_t, t)$. After the θ are trained using Eq. 4, we have the sampling equation:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)) + \sigma_t\mathbf{z} \tag{6}$$

Guidance: The guidance aims to provide the DGMs with conditional information during the sampling process so that the output image satisfies the predefined conditions. From Eq. 6, we denote $\mu_t := \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t))$,

$\log_\phi p(y|\mathbf{x}_t)$ is the conditional distribution of class labels; we have a sampling equation for $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ as:

$$\mathbf{x}_{t-1} \sim \mathcal{N}(\mu_t + s\sigma_t^2\nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t), \sigma_t) \tag{7}$$

4. Pixels as Parameters

From the sampling process of the DDPM model with guidance as Eq. 7, combined with Eq. 6 and reparameterization trick, the equation can be re-written as:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)) + s\sigma_t^2\nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t) + \sigma_t z \tag{8}$$

Derivation of the Eq. 8 is detailed in Appendix C.

From Eq. 8, the $\epsilon_\theta(\mathbf{x}_t, t)$ can be intuitively interpreted as the data density gradient, and the classification gradient as part of the update from $\mathbf{x}_t$ to $\mathbf{x}_{t-1}$. This motivates us to think of the whole process from a gradient point of view. We will form the whole process as an optimization with the initial parameter $\mathbf{x}_T \sim \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$.

4.1. Objectives of the process

Likelihood objective: Eq. 6 can be re-written in the form with $\mathbf{x}_0$ prediction as below:

$$\mathbf{x}_{t-1} = \frac{(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}(\frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}) + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t z \tag{9}$$

Derivation of the Eq. 9 can be found in Appendix C.

$(\frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t,t)}{\sqrt{\bar{\alpha}_t}})$ is the prediction of $\mathbf{x}_0$. Assume that $\epsilon_\theta(\mathbf{x}_t, t)$ is trained optimally, we can turn the Eq. 9 into:

$$\mathbf{x}_{t-1} = \frac{(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0 + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t z \tag{10}$$

The Eq. 10 can be derived from the Eq. 3 using the reparameterization trick where the sampling process is trying to match the distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$. Thus, the sampling process's first objective is to achieve the approximate maximum likelihood of the distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ at each timestep.

Classification condition objective: Consider the classification gradient term in the Eq. 8 with the Eq. 10, we have finally had two objectives as approximate maximum likelihood at timestep and the classification condition:

$$\mathbf{x}_{t-1} = \underbrace{\frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0 + \sigma_t z}_{\text{approximate likelihood}} + \underbrace{s\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t)}_{\text{classification}} \tag{11}$$

4.2. Gradients of the process

As we can see in the Eq. 11, by assuming that $\epsilon_\theta(\mathbf{x}_t, t) \approx \epsilon$ or $\tilde{\mathbf{x}_0} \approx \mathbf{x}_0$, the update from x_t to x_{t-1} includes two objectives which are approximate likelihood and classification condition. In practice, $\tilde{\mathbf{x}}_0$ is nowhere close to $\mathbf{x}_0$ causing the likelihood function's intractability. This results in the importance of the $\mathbf{x}_0$ prediction in the likelihood objective mentioned in Eq. 11. Thus, the model is interpreted as follows. As the term $\frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t$ is not affected by the output of the diffusion model as well as the classifier at each time-step, we can treat this term as the initial parameters with normalized constant $\mathbf{Z} = \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}$. The other terms could be the gradients that update into the initial parameters $\mathbf{Z}\mathbf{x}_t$. We can view the Eq. 11 as:

$$\mathbf{x}_{t-1} = \underbrace{\frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t}_{initial\ parameters} - \underbrace{\frac{-(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0}_{denoising\ gradient} - \underbrace{(-\sigma_t z)}_{diversity\ gradient} - \underbrace{(-s\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t)}_{\text{classification gradient}} \tag{12}$$

$\frac{-(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0$ is based on the prediction of $\mathbf{x}_0$ value, we call this update direction is the *denoising gradient*. For the $-\sigma_t z$ with z is random noise, we see that the generated image's diversity mainly depends on this randomness. Thus, we name this one as *diversity gradient*. The third term $-s\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t)$ is a common term in (Dhariwal & Nichol, 2021) which is the *classification gradient*.

We can generalize the gradient definition from Eq. 12 by deriving several ways to define the g_{den}:

$\mathbf{x}_0$ ***prediction:*** The first way is to define denoising gradient as in the Eq. 12, we denote this scheme as the $\mathbf{x}_0$ *prediction*.

$\mathbf{x}_0$ ***sampling:*** The second way we can define is the combination between $\mathbf{x}_0$ prediction and noise term as below:

$$\mathbf{x}_{t-1} = \underbrace{Z\mathbf{x}_t}_{initial\ parameters} - \underbrace{(\frac{-(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0 - \sigma_t z)}_{denoising\ gradient} - \underbrace{(-s\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t))}_{\text{classification gradient}} \tag{13}$$

noise prediction: Another way to define the denoising gradient is to base it on the original sampling equation of the DDPM (Eq. 6 and 8).

$$\mathbf{x}_{t-1} = \underbrace{\frac{1}{\sqrt{\alpha_t}}\mathbf{x}_t}_{\text{initial parameters}} - \underbrace{\frac{1-\alpha_t}{\sqrt{\alpha_t(1-\bar{\alpha}_t)}}\epsilon_\theta(\mathbf{x}_t,t)}_{\text{denoising gradient}} - \underbrace{(-s\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t))}_{\text{classification gradient}} - \underbrace{(-\sigma_t z)}_{\text{diversity gradient}} \tag{14}$$

We intuitively select $\mathbf{x}_0$ *prediction* term as the denoising gradient in the primary analysis of the paper as it is consistent with the final objective of the whole denoising diffusion model is to construct the original image $\mathbf{x}_0$. We will discuss other possible choices of denoising gradient in the Ablation study section 6.4.

In general, we model the denoising process of the DDPMs model into an optimization problem with two objectives. These two objectives have three updated gradients.

4.3. Conflict between gradients

Given the initial shared parameters $\mathbf{x}_t$, each objective's updated directions in section 4.2 might conflict.

In (Yu et al., 2020), the authors point out three conditions for the gradient conflict between any two tasks, which are **direction conflict**, **magnitude conflict**, and **high multi-task curvature**. Conflict happens only when all three conditions are satisfied. Given the gradients defined in Eq. 12, we can assume the high curvature between objectives due to the complication of data distribution. We will empirically show that there are conflicts in terms of direction conflict and magnitude conflict between gradients in Eq. 12.

Given three types of gradients defined in section 4.2, we have three pairs:

- *cls-den*: Pair of classification and denoising gradients.

Algorithm 1 DDPM denoising process with guidance

Input: class labels y, classification scale s
$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, ..., 1$ **do**
  $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $g \leftarrow s\nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t)$
  $\mathbf{x}_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)) + \sigma_t^2 g + \sigma_t z$
end for

- *cls-div*: Pair of classification and diversity gradients.
- *den-div*: Pair of denoising and diversity gradients.

Direction conflict: Given φ_{ij} is the angle between two gradient g_i and g_j of the i^{th}gradient and j^{th}gradient. The two gradients have direction conflicts if $\cos\varphi_{ij} < 0$. Based on this definition, we measure the percentage of instances having negative $\cos\varphi_{ij}$ for each pair in Figure 2(a),(b), and (c). From observation, all pairs maintain high direction conflicts during sampling. It is also reasonable to see the noisy direction conflict between the diversity gradient with the other two gradients, as the diversity is mainly constructed by the random noise z.

Magnitude conflict: In (Yu et al., 2020), gradient magnitude conflict between two gradient g_i and g_j is defined as:

$$\Phi(g_i, g_j) = \frac{-2||g_i||_2||g_j||_2}{||g_i||_2^2 + ||g_j||_2^2} \tag{15}$$

The Eq. 15 shows the magnitude conflict between two gradients. If the two gradients have the same magnitude, the $\Phi(g_i, g_j)$ will go to -1. In contrast, the $\Phi(g_i, g_j)$ goes to 0. The average magnitude conflict of each pair of gradients in Eq. 12 is plotted in Figure 2(d),(e), and (f). The magnitude conflict of the *cls-den* and *cls-div* pair is extremely high. However, the magnitude conflict of the *den-div* pair is relatively low and reaches nearly -1 at the process's end resulting in less conflict. This is due to the fixed schedule given by the diffusion model. As the magnitude conflict condition is not satisfied for *den-div*, we can say no conflict between the diversity and the denoising gradient. As a result, only two pairs have conflicts *cls-den* and *cls-div*.

5. Conflict projection

In section 4.3, conflicts between the defined gradients in Eq. 12 are detected. We propose alleviating these conflicts by projecting conflict if negative transfer exists between two gradients. Negative transferring between two gradients g_i and g_j happens if the $\cos\varphi_{g_i,g_j} < 0$. After that, the gradient of each objective is projected on the orthonormal plane of the other gradients to remove the destructive conflicts.

Different from a standard optimization on neural network parameters, the update directly to the pixels of the image has several constraints. Firstly, the $\mathbf{x}_{t-1}$ distribution must satisfy a predefined Gaussian distribution as Eq. 10. Secondly, pretrained diffusion model $\epsilon_\theta(\mathbf{x}_t, t)$ is sensitive to the out-of-distribution of input data. If the input $\mathbf{x}_t$ distribution changes dramatically, the output of $\epsilon_\theta(\mathbf{x}_t, t)$ will be less meaningful. As a result, to avoid abrupt change to the output distribution of $\mathbf{x}_{t-1}$, we introduce a weight for projection to avoid the effects of the conflict projection in some cases. Therefore, we have the formula for projecting the gradient i onto the orthonormal plane of gradient j as:

$$\hat{g}_i := g_i - \delta\frac{g_i \cdot g_j}{||g_j||^2} g_j \tag{16}$$

δ is the projection weight to control the effect of conflict projection.

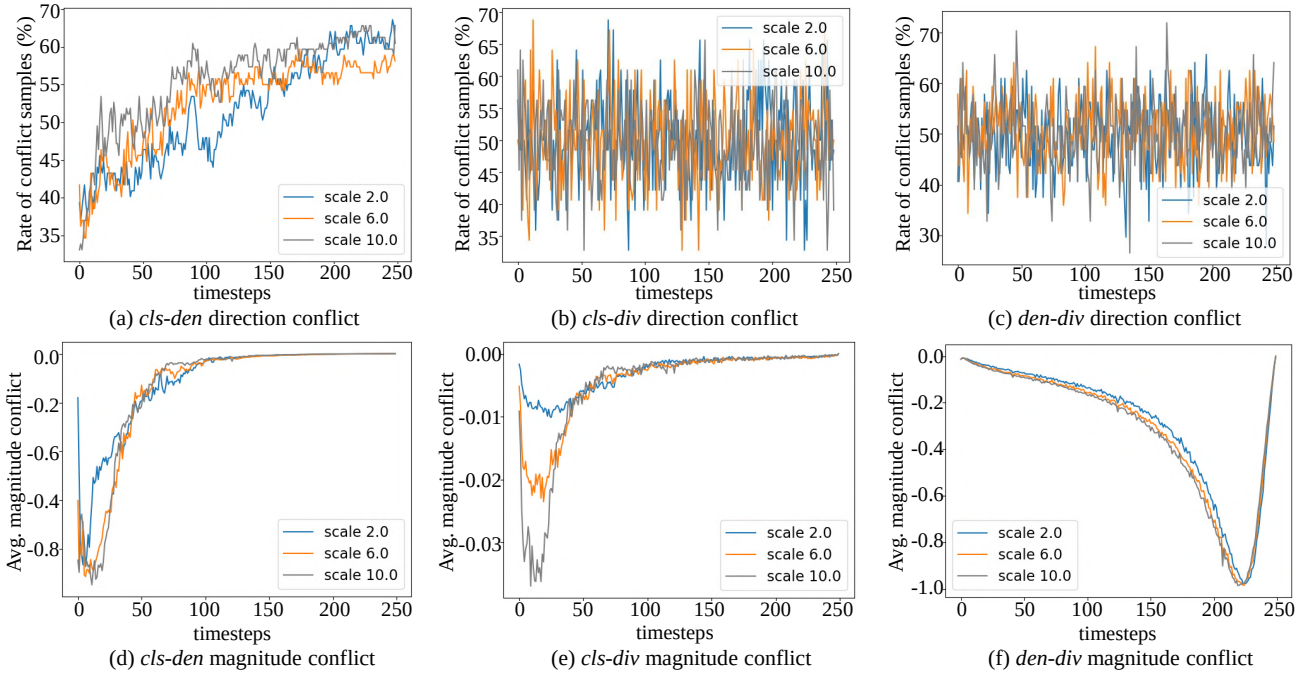
In section 4.2, we have defined three gradients that affect the sampling process of a DGM. Following the $\mathbf{x}_0$ *prediction* to define denoising gradient as in the Eq. 12, We denote:

- $g_{den} = \frac{-(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0$ as the denoising gradient
- $g_{div} = -\sigma_t z$ as the diversity gradient
- $g_{cls} = -s\sigma_t^2\nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t)$ as the classification gradient

From the analysis in section 4.3, the conflicts exist in the *cls-den* and *cls-div* pair but not in the *den-div* pair due to theoretical guarantee. Therefore, in the proposed method, we only project conflict for the *cls-den* and *cls-div*. We modify the standard DDPM sampling algorithm 1 into a conflict projection version. The whole algorithm is presented in

Algorithm 2 PixelAsParams Denoising Process (PxP)

Input: class labels y, gradient scale s and project scale $\delta_1, \delta_2, \delta_3, \delta_4$
$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, ..., 1$ **do**
  $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $g_{cls} \leftarrow -s\sigma_t^2\nabla p_\phi(y|\mathbf{x}_t)$
  $g_{den} \leftarrow \frac{(\alpha_t-1)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}(\frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t,t)}{\sqrt{\bar{\alpha}_t}})$
  $g_{div} \leftarrow -\sigma_t z$
  if $\cos\varphi(g_{cls}, g_{den}) < 0$ **then**
    $\hat{g}_{cls} \leftarrow g_{cls} - \delta_1\frac{g_{cls}\cdot g_{den}}{||g_{den}||^2} g_{den}$
    $\hat{g}_{den} \leftarrow g_{den} - \delta_2\frac{g_{den}\cdot g_{cls}}{||g_{cls}||^2} g_{cls}$
  end if
  if $\cos\varphi(g_{cls}, g_{div}) < 0$ **then**
    $\hat{g}_{cls} \leftarrow \hat{g}_{cls} - \delta_3\frac{\hat{g}_{cls}\cdot g_{div}}{||g_{div}||^2} g_{div}$
    $\hat{g}_{div} \leftarrow g_{div} - \delta_4\frac{g_{div}\cdot g_{cls}}{||g_{cls}||^2} g_{cls}$
  end if
  $\mathbf{x}_{t-1} \leftarrow \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t - \hat{g}_{den} - \hat{g}_{div} - \hat{g}_{cls}$
end for



(a) *cls-den* direction conflict (b) *cls-div* direction conflict (c) *den-div* direction conflict

(d) *cls-den* magnitude conflict (e) *cls-div* magnitude conflict (f) *den-div* magnitude conflict

Figure 2. The percentage conflicting samples increase as the number of timesteps of cls-den (a), of cls-div (b) and of den-div (c) conflicting samples remain high during the denoising process. The magnitude conflict increases during the denoising process for the cls-den pair (d) and stays very high for the cls-div pair (e). At the same time, we see very low den-div magnitude conflict (f). These observations indicate low conflict for the den-div pair but evident conflict for both cls-den and the cls-div pairs.

Algorithm 2. This proposed algorithm utilizes $\mathbf{x}_0$ *prediction* scheme to define g_{den}, g_{div} and g_{cls}. Similar to section 4.2, the methods can also be generalized for conflict-solving on different definitions of gradients in Eq. 13 and 14.

The algorithm's four hyper-parameters $\delta_1, \delta_2, \delta_3, \delta_4$ are likely to be the default values. In several cases, one of the values is expected to reduce when little conflict is detected, or $||g_{cls}||$ is too large compared to other terms or redundant conflict solving. For example, if the diffusion model is conditionally trained, the conditional information is compatible with the diversity leading to unnecessary solving the conflict between the two gradients, hence the δ_3 and δ_4 should be lowered down.

6. Experiments

All experiments are conducted on the ImageNet dataset at 64x64, 128x128, 256x256, and CIFAR10 to evaluate our method since these datasets offer classification classes and pretrained diffusion models. We will compare our methods with several baselines, including BigGAN (Brock et al., 2018), LOGAN (Wu et al., 2019a), DCTrans (Nash et al., 2021), ClS-Free (Ho & Salimans, 2022), VQ-VAE2 (Razavi et al., 2019), IDDPM (Nichol & Dhariwal, 2021), and ADM/ADM-G (Dhariwal & Nichol, 2021). IDDPM-G is the DDIM model with guidance sampling (Algorithm 1 on pretrained IDDPM.

The proposed PixelAsParams sampling technique in Algorithm 2 will be denoted as PxP. If the PxP is applied to the pretrained DDPM, the experiment will be noted as DDPM-PxP. Similar to other models.

6.1. Conflict alleviation for guidance sampling

We first evaluate the proposed algorithms for solving the trade-off between image quality and classification conditions. All the diffusion models are trained unconditionally. Our first objective is to show there is no trade-off between quantitative measures. After that, our method is expected to balance conditional information, image quality, and diversity qualitatively.

Quantitative improvement: Table 1 shows the results utilizing the PxP method to solve the conflict between classification guidance and other diffusion elements. The method achieves excellent results in all three generative scores IS, FID and sFID. The ADM and the ADM-G (Dhariwal & Nichol, 2021) often achieve a very high IS score but must sacrifice the FID or sFID. The proposed ADM-PxP can improve all scores simultaneously. The Recall value, which has been proved to show the generated samples' diversity property (Kynkäänniemi et al., 2019), shows the ADM-PxP's superiority over ADM-G means we successfully achieved

Table 1. Using the proposed method would help avoid the trade-off to achieve better IS/FID. The improvement is significant on both IDDPM (Nichol & Dhariwal, 2021) and ADM (Dhariwal & Nichol, 2021). All of the diffusion models in this table are unconditionally trained. Bold models are our proposes.

MODEL	IS	FID	SFID	PREC	REC
IMAGENET 64X64					
ADM	25.64	9.95	6.58	0.60	0.65
ADM-G	46.90	6.40	9.67	0.65	0.54
ADM-PXP	**61.88**	**4.96**	**6.57**	**0.69**	0.59
IDDPM	16.02	18.35	5.08	0.60	0.57
IDDPM-G	18.89	13.62	**4.43**	0.63	0.55
IDDPM-PXP	**33.27**	**8.49**	4.49	**0.67**	0.58
IMAGENET 256X256					
ADM	39.7	26.21	6.35	0.61	0.63
ADM-G	96.15	11.96	10.28	**0.75**	0.45
ADM-PXP	**124.25**	**9.03**	**6.25**	**0.75**	0.51
CIFAR10 32X32					
ADM	9.55	2.87	4.36	**0.69**	**0.60**
ADM-G	**9.58**	2.85	4.30	0.68	**0.60**
ADM-PXP	9.56	**2.78**	**4.26**	0.68	**0.60**

better diversity than vanilla guidance. The recall value of ADM-PxP is lower than ADM in some cases due to the constraints of the conditional generator ADM-PxP to samples following conditions. The unconditional ADM has more freedom in search space, leading to more diverse samples.

Qualitative improvement Figure 1 and 3 show our solution to the trade-off between diversity and conditional information. In section 4.3, the conflicts are detected in *cls-den* and *cls-div* pair. After the conflict is solved, the generated samples mitigate the trade-off. This is the evidence for solving the trade-off by solving the conflicts between gradients. More samples are shown in Appendix D.

From both qualitative and quantitative improvement, the proposed framework has solved the main question of the work in modeling and alleviating the trade-off between image quality, conditional information, and diversity.

6.2. State-of-the-art image synthesis

Classification guidance is not only used for providing knowledge for a conditional generation. The method is also used for supporting a conditional diffusion model to achieve better performances. In (Dhariwal & Nichol, 2021), the authors significantly improve using classification guidance on a pretrained conditional diffusion model. This section will show that our proposed PxP can improve the existing methods on different numerical metrics for generative tasks.

The experimental results in Table 2 show the state-of-the-art. Although the guidance is not necessary for providing conditional knowledge to the denoising process anymore, it still helps improve the numerical values in most measures. The most significant improvement is in IS score. While ADM and ADM-G (Dhariwal & Nichol, 2021) are still significantly lower than most other GAN-based methods, ADM-PxP achieves the highest score for this measure for the first time. We achieve the state-of-the-art for most of the measures except the Precision value, where this figure is still lower than BigGAN on most resolutions. In contrast, We achieve a much higher Recall score.

Table 2. Comparison with other state-of-the-art generative models. Using PxP in Algorithm 2 helps achieve the best scores on most measures. Some values in the precision measure of ADM-PxP are lower than BigGAN, yet the algorithm provides a much better recall value. † means the score is evaluated from the samples provided by the paper. ∗ means the pretrained diffusion is conditionally trained. ‡ means the values taken directly from the paper due to the lack of available inference samples or pretrained models. Bold models are our proposes.

MODEL	IS	FID	SFID	PREC	REC
IMAGENET 64X64					
BIGGAN†	44.99	4.06	3.96	0.79	0.48
IDDPM*	46.31	2.90	**3.78**	0.73	0.62
ADM*	53.79	2.07	4.29	0.73	**0.63**
ADM-G*	75.98	2.47	4.88	**0.80**	0.57
ADM-PXP*	**78.31**	**1.84**	3.97	0.76	0.60
IMAGENET 128X128					
BIGGAN†	145.93	6.02	7.18	**0.86**	0.35
LOGAN‡	148.2	3.36			
ADM*	92.53	5.91	5.08	0.69	**0.65**
ADM-G*	141.55	2.98	5.10	0.77	0.59
ADM-PXP*	**191.38**	**2.64**	**4.97**	0.79	0.57
IMAGENET 256X256					
BIGGAN†	202.77	7.03	7.29	**0.87**	0.27
DCTRANS‡	-	36.51	8.24	0.36	0.67
VQ-VAE-2‡	-	31.11	17.38	0.36	0.57
IDDPM‡	-	12.26	5.42	0.70	0.62
ADM*	100.98	10.94	6.02	0.69	**0.63**
ADM-G*	188.91	4.58	5.23	0.81	0.52
ADM-PXP*	**216.11**	**4.00**	**5.19**	0.81	0.53

6.3. Extension to other types of guidance

Text-to-Image guidance: We set up similar experiments to CLIP guidance in GLIDE (Nichol et al., 2021). 30k captions are sampled from the CocoCaption dataset for Image guidance. The noise-aware CLIP and pretrained diffusion are taken from the paper. Without any modification to the Eq. 12, PXP can be applied directly to the GLIDE guidance with CLIP. Table 3 shows the improvement over GLIDE using PxP on Zero-shot FID. Our proposed method can work on conditional image and text-to-image generation tasks.

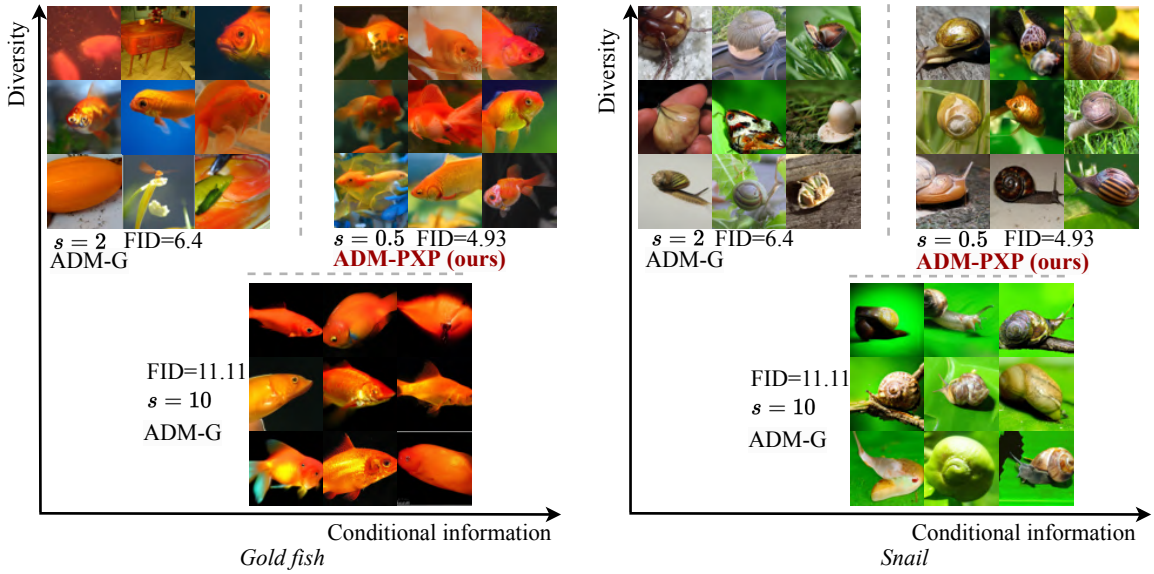


Figure 3. The left figure is the samples with "Goldfish" condition, and the right figure is the set of samples with the "Snail" condition. Without PxP, the diversity is sacrificed for conditional information. PxP achieves both diversity and conditional information.

Table 3. PxP helps to improve on both MSCoco 64x64 and MSCoco 256x256 significantly

Model	Zero-shot FID	Model	Zero-shot FID
MSCoco 64x64		**MSCoco 256x256**	
GLIDE	24.75	GLIDE	34.78
GLIDE-PxP	**23.78**	**GLIDE-PxP**	**32.83**

Classifier-free guidance: To extend the PxP for application over the classifier-free guidance method. We re-formulate the classifier-free guidance method as follows:

$$\tilde{\epsilon}_t = (1+w)\epsilon_\theta(\mathbf{x}_t, c) - w\epsilon_\theta(\mathbf{x}_t) \tag{17}$$

$$= \epsilon_\theta(\mathbf{x}_t, c) - w(\epsilon_\theta(\mathbf{x}_t, c) - \epsilon_\theta(\mathbf{x}_t)) \tag{18}$$

Eq. 17 is the original equation for updating the noise prediction of classifier-free guidance. The equation is transformed to Eq. 18 to form classification information. The term $\epsilon_\theta(\mathbf{x}_t, c) - \epsilon_\theta(\mathbf{x}_t)$ is interpreted as classification information. We denote this term as C. Rewrite the Eq. 18, we have:

$$\tilde{\epsilon}_t = \epsilon_\theta(\mathbf{x}_t, c) + wC. \tag{19}$$

Combine Eq. 19 with sampling Eq. 9, we have:

$$\mathbf{x}_{t-1} = \underbrace{\frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t}_{initial\ parameters} - \underbrace{\frac{-(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}\tilde{\mathbf{x}}_0^{cfg}}_{denoising\ gradient} - \underbrace{(-\sigma_t z)}_{diversity\ gradient} - \underbrace{\frac{(1-\alpha_t)w}{\sqrt{(1-\bar{\alpha}_t)\bar{\alpha}_t}}C}_{\text{classification gradient}} \tag{20}$$

Where $\tilde{\mathbf{x}}_0^{cfg} = (\frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\tilde{\epsilon}_\theta(\mathbf{x},t)}{\sqrt{\bar{\alpha}_t}})$. Note that, the classification gradient term, $\frac{(1-\alpha_t)}{\sqrt{(1-\bar{\alpha}_t)\bar{\alpha}_t}}C$ is treated similarly to $\sigma_t^2\nabla_{\mathbf{x}_t}\log p_\phi(y|\mathbf{x}_t)$, and w is equivalent to the guidance scale s in Eq. 12.

Algorithm 2 is applied to the defined terms in Eq. 20 to achieve results in Table 4. The results show a significant improvement over different metrics, and the observation is similar to Table 2.

Table 4. Application of PxP on Classifier-free guidance. CFree represents classifier-free guidance, and CFree-PxP is the application of the proposed PxP on CFree. Note: Due to the lack of pretrained joint unconditional and conditional diffusion provided by (Ho & Salimans, 2022), we utilize the unconditional and conditional diffusion models separately, suggested by the authors.

Model	IS	FID	sFID	Prec	Rec
ImageNet 64x64					
CFree*	58.76	1.92	4.32	0.75	**0.62**
CFree-PxP*	**59.17**	**1.84**	**4.28**	**0.76**	**0.62**
ImageNet 256x256					
CFree*	191.31	3.76	4.87	0.81	**0.55**
CFree-PxP*	**206.94**	**3.45**	**4.82**	**0.83**	0.541

6.4. Ablation study

The ablation study is conducted on three aspects that can influence the model's performance: the choice of g_{den}, the pairs needed conflict solving, and the effect of gradient scale s on the performance of the model.

Choosing g_{den}: As discuss in section 4.2, there are several ways to define the g_{den}. In this part, we will investigate the effect of solving conflicts between different definitions of gradients. The choice of g_{den} significantly affects the

*Table 5. The results shown the $\mathbf{x}_0$ prediction scheme (Eq. 12) achieves the best solution compared to others. The performance of $\mathbf{x}_0$ sampling and $\mathbf{x}_0$ prediction are similar to each other due to the conflicts between terms being solved. However, the $\mathbf{x}_0$ prediction offers a better solution due to the decomposition of the diversity and denoising, which helps reduce the conflicts between conditional information and diversity (**ImageNet 64x64**)*

MODEL	IS	FID	SFID
NO PXP	25.64	9.95	6.58
noise prediction	45.03	23.55	22.84
$\mathbf{x}_0$ *sampling*	50.41	5.19	**6.39**
$\mathbf{x}_0$ *prediction*	**61.88**	**4.93**	6.57

*Table 6. For each pair cls-den or cls-div, we see clear improvement by solving conflict at each pair. The combination of conflict solving on both pairs achieves the best result. (**ImageNet 64x64**)*

cls-den	*cls-div*	*den-div*	IS	FID	SFID
×	×	×	46.90	6.40	9.67
√	×	×	59.22	5.83	8.19
×	√	×	57.98	5.09	7.64
×	×	√	20.87	26.97	24.77
√	√	√	52.88	12.24	13.30
√	√	×	**61.88**	**4.96**	**6.57**

performance of the PxP Algorithm 2 due to the difference in degrees of conflict given by varying g_{den} and g_{div}.

The experimental results of $\mathbf{x}_0$ *prediction*, $\mathbf{x}_0$ *sampling*, and *noise prediction* are presented in Table 5. The performance of $\mathbf{x}_0$ *prediction* is better than the *noise prediction* due to the objective consistency in predicting the original image $\mathbf{x}_0$ instead of matching the noise prediction with are unclear objective. Furthermore, the conflict solving with noise prediction causes redundant information with the *cls-div* pair given optimally trained noise prediction. The performance of $\mathbf{x}_0$ *prediction* and $\mathbf{x}_0$ *sampling* do not differ much. The $\mathbf{x}_0$ *prediction* performs better due to the breakdown of gradient terms to detect more conflicts in *cls-div* and *cls-den*.

Effects of each pair of gradients conflict: This experiment shows the effect of each pair of gradients in Eq. 12 on the sampling performance. The results are shown in Table 6. We achieve the most significant improvement over the baseline based on the two pairs *cls-den* and *cls-div*. With all pairs considered simultaneously, this is similar to PC-grad (Yu et al., 2020) pair-wise conflict solving on the denoising sampling process. This results in abysmal performance due to the violation in solving conflicts between two non-conflicting items: diversity gradient and denoising gradients. We have verified our hypothesis in the section 4.3.

Classification gradient scale effects: The effects of using the classification gradient scale are shown in Figure 4. Conflict-solving pairs are affected differently when the classification gradient scale s increases. In all cases, using the PxP sampling technique with the two pairs *cls-den* and *cls-div*, we achieve a much more significant improvement in IS

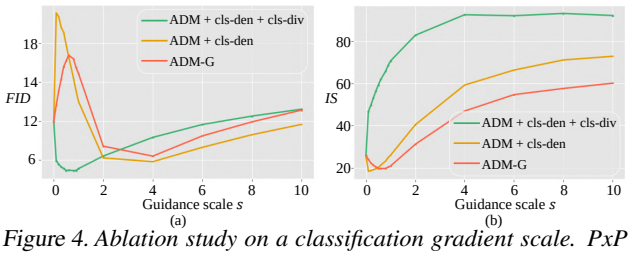


*Figure 4. Ablation study on a classification gradient scale. PxP achieves stable and significant improvement over ADM-G by conflict-solving between the gradient pairs. **ImageNet64x64***

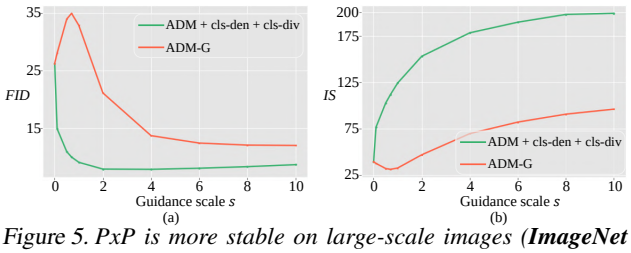


*Figure 5. PxP is more stable on large-scale images (**ImageNet 256x256**) when increasing the guidance scale.*

score. For the FID/sFID score, the combination between *cls-den* and *cls-div* achieves significant improvement compared to others with minimal values of classification gradient scale s, then starts to worsen when s increases. This might result from suppression over diversity, given extensive conditional information. Nevertheless, the higher resolutions result in more stable results, as in Figure 5.

7. Conclusion

This paper solves the problem of trade-offs for the DGMs sampling process with guidance. The whole denoising process is viewed through the gradient perspective, where the pixels of the image are turned into optimized parameters. Besides, diffusion terms with guidance signals are treated as gradients or update directions. Based on this view, conflicts between gradients are hypothesized to be the reason behind the trade-off problem. We solve the conflict problems by a projection technique, significantly improving image quality, diversity and conditional information simultaneously.

Acknowledgements

This work was supported in part by the Australian Research Council under Project DP210101859 and the University of Sydney Research Accelerator (SOAR) Prize. The authors acknowledge the use of the National Computational Infrastructure (NCI) which is supported by the Australian Government, and accessed through the NCI Adapter Scheme and Sydney Informatics Hub HPC Allocation Scheme.

Besides, the AI training platform supporting this work were provided by High-Flyer AI (Hangzhou High-Flyer AI Fundamental Research Co., Ltd.)

References

Bao, F., Li, C., Zhu, J., and Zhang, B. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022.

Batzolis, G., Stanczuk, J., Schönlieb, C.-B., and Etmann, C. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021.

Brock, A., Donahue, J., and Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Chao, C.-H., Sun, W.-F., Cheng, B.-W., Lo, Y.-C., Chang, C.-C., Liu, Y.-L., Chang, Y.-L., Chen, C.-P., and Lee, C.-Y. Denoising likelihood score matching for conditional score-based data generation. *arXiv preprint arXiv:2203.14206*, 2022.

Chen, S., Sun, P., Song, Y., and Luo, P. Diffusiondet: Diffusion model for object detection. *arXiv preprint arXiv:2211.09788*, 2022.

Chen, Z., Badrinarayanan, V., Lee, C.-Y., and Rabinovich, A. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pp. 794–803. PMLR, 2018.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

Dung, D. A. and Binh, H. T. T. Gdegan: Graphical discriminative embedding gan for tabular data. In *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–11. IEEE, 2022.

Gong, M., Xu, Y., Li, C., Zhang, K., and Batmanghelich, K. Twin auxilary classifiers gan. *Advances in neural information processing systems*, 32, 2019.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Hou, L., Cao, Q., Shen, H., Pan, S., Li, X., and Cheng, X. Conditional gans with auxiliary discriminative classifier. In *International Conference on Machine Learning*, pp. 8888–8902. PMLR, 2022.

Javaloy, A. and Valera, I. Rotograd: Gradient homogenization in multitask learning. *arXiv preprint arXiv:2103.02631*, 2021.

Kang, M., Shim, W., Cho, M., and Park, J. Rebooting acgan: Auxiliary classifier gans with stable training. *Advances in Neural Information Processing Systems*, 34:23505–23518, 2021.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., and Aila, T. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019.

Liu, B., Liu, X., Jin, X., Stone, P., and Liu, Q. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021a.

Liu, L., Li, Y., Kuang, Z., Xue, J.-H., Chen, Y., Yang, W., Liao, Q., and Zhang, W. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=IMPnRXEWpvr.

Liu, N., Li, S., Du, Y., Torralba, A., and Tenenbaum, J. B. Compositional visual generation with composable diffusion models. *arXiv preprint arXiv:2206.01714*, 2022.

Liu, X., Park, D. H., Azadi, S., Zhang, G., Chopikyan, A., Hu, Y., Shi, H., Rohrbach, A., and Darrell, T. More control for free! image synthesis with semantic diffusion guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 289–299, 2023.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Nash, C., Menick, J., Dieleman, S., and Battaglia, P. W. Generating images with sparse representations. *arXiv preprint arXiv:2103.03841*, 2021.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

Odena, A., Olah, C., and Shlens, J. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pp. 2642–2651. PMLR, 2017.

Preechakul, K., Chatthee, N., Wizadwongsa, S., and Suwajanakorn, S. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10619–10629, 2022.

Qiu, Z., Yang, H., Fu, J., and Fu, D. Learning spatiotemporal frequency-transformer for compressed video super-resolution. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVIII*, pp. 257–273. Springer, 2022.

Qiu, Z., Yang, Q., Wang, J., Feng, H., Han, J., Ding, E., Xu, C., Fu, D., and Wang, J. Psvt: End-to-end multi-person 3d pose and shape estimation with progressive video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21254–21263, 2023.

Razavi, A., Van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., and Norouzi, M. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Sener, O. and Koltun, V. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

Shan, W., Liu, Z., Zhang, X., Wang, Z., Han, K., Wang, S., Ma, S., and Gao, W. Diffusion-based 3d human pose estimation with multi-hypothesis aggregation. *arXiv preprint arXiv:2303.11579*, 2023.

Sinha, A., Song, J., Meng, C., and Ermon, S. D2c: Diffusion-decoding models for few-shot conditional generation. *Advances in Neural Information Processing Systems*, 34:12533–12548, 2021.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Song, Y. and Ermon, S. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Tran, N.-T., Bui, T.-A., and Cheung, N.-M. Dist-gan: An improved gan using distance constraints. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 370–385, 2018.

Vandenhende, S., Georgoulis, S., Van Gansbeke, W., Proesmans, M., Dai, D., and Van Gool, L. Multi-task learning for dense prediction tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2021.

Wang, Y., Xu, C., Du, B., and Lee, H. Learning to weight imperfect demonstrations. In *International Conference on Machine Learning*, pp. 10961–10970. PMLR, 2021.

Wang, Y., Wang, X., Dinh, A.-D., Du, B., and Xu, C. Learning to schedule in diffusion probabilistic models. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023.

Wu, Y., Donahue, J., Balduzzi, D., Simonyan, K., and Lillicrap, T. Logan: Latent optimisation for generative adversarial networks. *arXiv preprint arXiv:1912.00953*, 2019a.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. Detectron2. https://github.com/facebookresearch/detectron2, 2019b.

Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., and Finn, C. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33: 5824–5836, 2020.

Zhao, Y., Li, C., Yu, P., Gao, J., and Chen, C. Feature quantization improves gan training. *arXiv preprint arXiv:2004.02088*, 2020.

Table 7. All hyper-parameters required for reproducing the results. * denotes for conditionally trained diffusion model.

MODEL	DATASET	δ_1	δ_2	δ_3	δ_4	s	TIME-STEPS
TABLE 1							
ADM, IDDPM	IMAGENET64X64, 256X256, CIFAR 32X32	-	-	-	-	0.0	250
ADM-G	IMAGENET64X64	0.0	0.0	0.0	0.0	2.0	250
ADM-PXP	IMAGENET64X64	1.0	1.0	1.0	1.0	0.5	250
IDDPM-G	IMAGENET64X64	0.0	0.0	0.0	0.0	2.0	250
IDDPM-PXP	IMAGENET64X64	1.0	1.0	1.0	1.0	0.5	250
ADM-G	IMAGENET256X256	0.0	0.0	0.0	0.0	10.0	250
ADM-PXP	IMAGENET256X256	1.0	1.0	1.0	1.0	1.0	250
ADM-G	CIFAR 32X32	0.0	0.0	0.0	0.0	0.3	250
ADM-PXP	CIFAR 32X32	0.1	$3e^{-2}$	$3e^{-2}$	$7e^{-3}$	0.2	250
TABLE 2							
ADM*	IMAGENET64X64, 128X128, 256X256	-	-	-	-	0.0	250
ADM-G*	IMAGENET64X64	-	-	-	-	2.0	250
ADM-PXP*	IMAGENET64X64	1.0	1.0	$1e^{-3}$	$1e^{-3}$	0.5	250
ADM-G*	IMAGENET128X128	0.0	0.0	0.0	0.0	0.5	250
ADM-PXP*	IMAGENET128X128	1.0	1.0	$1e^{-3}$	$1e^{-3}$	0.5	250
ADM-G*	IMAGENET256X256	0.0	0.0	0.0	0.0	1.0	250
ADM-PXP*	IMAGENET256X256	1.0	1.0	$1e^{-3}$	$1e^{-3}$	0.7	250
TABLE 5							
NO PXP	IMAGENET64X64	0.0	0.0	0.0	0.0	2.0	250
ALL MODELS	IMAGENET64X64	1.0	1.0	1.0	1.0	0.5	250
TABLE 6							
cls-den	IMAGENET64X64	1.0	1.0	0.0	0.0	0.5	250
cls-div	IMAGENET64X64	0.0	0.0	1.0	1.0	0.5	250
den-div	IMAGENET64X64	0.0	0.0	0.0	0.0	0.5	250
cls-div + *cls-den*	IMAGENET64X64	1.0	1.0	1.0	1.0	0.5	250
cls-div + *cls-den* + *den-div*	IMAGENET64X64	1.0	1.0	1.0	1.0	0.5	250

A. Implementation details

All the hyperparameters for reproducing all the results are available in Table 7. For all sampling processes with conditionally trained diffusion models that have PxP (Algorithm 2), we see that the conflict solving between the pair *cls-div* is relatively tiny. This is because the diffusion model has been conditionally trained in which the diversity conflict with conditional information has been alleviated during training. Further conflict resolution will cause redundant steps and remove essential features.

All the experiments are run with cluster nodes with 8GPUs NVIDIA A-100. The code for the proposed sampling process is available here: https://github.com/dungdinhanh/pxpguided-diffusion

B. Hyperparameter selection

We verify the hyper-parameters sensitivity on Diffusion ImageNet64x64 with two cases. The first case is the unconditional diffusion model, and the second case is the conditional diffusion model.

Unconditional diffusion model Table 8 show the hyper-parameter sensitivity. Mean $\pm$ STD is calculated based on every metric when tuning each hyper-parameter. STD represents how varying the performance is associated with the change in the hyper-parameter. The results indicate that $\gamma_1, \gamma_2, \gamma_3$ are not so sensitive since we observe very minor variances in performance. The most sensitive value is γ_4. Increasing γ_4 to 1 benefits the IS/FID and sFID. We hypothesize that the conflict-solving between diversity and classification plays the most crucial role in this scenario. Thus, it is safe to set as default values $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1.0$, although slight improvement is achieved by tuning γ_1, γ_3.

Conditional diffusion model

Since the $\gamma_1, \gamma_2, \gamma_3$ does not vary significantly on both unconditional and conditional diffusion models, Table 9 shows γ_4

Table 8. Default values are set as $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1.0$, at one block, we vary the value of one hyperparameter while keeping other hyperparameters the same as the default. $\gamma_1, \gamma_2, \gamma_3$ are not so sensitive since they have minor variances in performance. The most sensitive hyperparameter is γ_4.

IS	FID	sFID	γ_1	γ_2	γ_3	γ_4	MEAN ± STD
IMAGENET 64X64							
59.41	4.96	6.52	0.2	1.0	1.0	1.0	IS = 59.35 ± 0.36, FID = 4.91 ± 0.035, sFID= 6.46 ± 0.055
58.87	4.89	6.44	0.4	1.0	1.0	1.0	
59.75	4.91	6.40	0.6	1.0	1.0	1.0	
59.38	4.88	6.50	0.8	1.0	1.0	1.0	
49.63	5.14	6.90	1.0	0.2	1.0	1.0	IS = 54.46 ± 3.7, FID = 5.02 ± 0.10, sFID= 6.7 ± 0.14
53.74	5.04	6.74	1.0	0.4	1.0	1.0	
56.37	5.00	6.66	1.0	0.6	1.0	1.0	
58.12	4.88	6.55	1.0	0.8	1.0	1.0	
58.78	4.91	6.37	1.0	1.0	0.2	1.0	IS = 59.07 ± 0.47, FID = 4.96 ± 0.03, sFID= 6.50 ± 0.09
59.57	4.99	6.56	1.0	1.0	0.4	1.0	
58.56	4.96	6.55	1.0	1.0	0.6	1.0	
59.38	4.98	6.54	1.0	1.0	0.8	1.0	
23.65	14.44	6.35	1.0	1.0	1.0	0.2	IS = 33.99 ± 9.85, FID = 9.62 ± 3.83, sFID= 6.18 ± 0.12
29.01	10.77	6.15	1.0	1.0	1.0	0.4	
37.06	7.54	6.05	1.0	1.0	1.0	0.6	
46.24	5.73	6.20	1.0	1.0	1.0	0.8	
59.18	4.93	6.50	1.0	1.0	1.0	1.0	(DEFAULT CASE)

sensitivity. In contrast to Table 8, the increase in the γ_4 leads to the trade-off between IS and FID. We hypothesize that this results from the impact on the diversity of terms (Eq. 12) by the output of the conditional diffusion model. The conditional diffusion model output contains σ_t, which indicates the diversity within a predefined condition (class). On the other hand, the classification gradient term only provides information to distinguish between classes. As a result, the conflict-solving between these terms will remove the diversity inside each class (leading to an FID increase) and enhances the diversity between classes (leading to better IS). To balance these two types of diversity, it is recommended that γ_4 should be set to a very small value.

Table 9. γ_4 sensitivity in the conditional diffusion model. Other hyper-parameters are set as default.

IS	FID	sFID	γ_1	γ_2	γ_3	γ_4
IMAGENET 64X64						
78.13	184	4.00	1.0	1.0	1.0	0.01
82.53	1.93	4.05	1.0	1.0	1.0	0.05
87.74	2.16	4.12	1.0	1.0	1.0	0.1
114.34	3.79	4.67	1.0	1.0	1.0	0.4

From the results in Table 8 and 9, we have a clear strategy to select the hyper-parameters as follows:

- For $\gamma_1, \gamma_2, \gamma_3$, these are not sensitive hyper-parameters. It is safe to set these values as default $\gamma_1 = \gamma_2 = \gamma_3 = 1.0$
- Although γ_4 varies the performance significantly, we offer a strategy to select the γ_4:
  - In unconditional diffusion case, set $\gamma_4 = 1.0$ as default value.
  - In the conditional diffusion case, if the application does not require diversity but requires the conditional information to be precisely generated, we can set $\gamma_4 = 1.0$. If diversity inside the condition is required, we can consider lowering the γ_4 into a small value.

C. Mathematical clarification

Guidance background: The guidance aims to provide the DGMs with conditional information during the sampling process so that the output image satisfies the predefined conditions. From Eq. 6, we denote $\mu_t := \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t))$, we have:

$$\begin{aligned} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= \mathcal{N}(\mu_t, \sigma_t) \\ \log p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= -\frac{1}{2}(\mathbf{x}_t - \mu_t)^T \sigma_t^{-1}(\mathbf{x}_t - \mu_t) + C \end{aligned} \tag{21}$$

with C as the constant.

$\log_\phi p(y|\mathbf{x}_t)$ is the conditional distribution of class labels. Due to the assumption that $\log_\phi p(y|\mathbf{x}_t)$ has low curvature compared to the σ_t^{-1}, $\log p_\theta(y|\mathbf{x}_t)$ can be approximated using a Taylor expansion around $\mathbf{x}_t = \mu_t$. We have:

$$\log p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) p_\phi(y|\mathbf{x}_{t-1}) \approx \log p(z) + C \tag{22}$$

Where $z \sim \mathcal{N}(\mu_t + s\sigma_t^2 g, \sigma_t)$ and $g = \nabla_{x_t} \log p_\phi(y|\mathbf{x}_t)$. Finally we have sampling equation for $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ as:

$$\mathbf{x}_{t-1} \sim \mathcal{N}(\mu_t + s\sigma_t^2 \nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t), \sigma_t) \tag{23}$$

Derive equation for sampling: We start with the sampling process of the DDPM model with guidance as in Eq. 7. By utilizing parameterize trick, the Eq. 7 and 6 can be re-written as:

$$\mathbf{x}_{t-1} = \mu_t + s\sigma_t^2 \nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t) + \sigma_t z \tag{24}$$

$$= \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)) + s\sigma_t^2 \nabla_{\mathbf{x}_t} \log p_\phi(y|\mathbf{x}_t) + \sigma_t z \tag{25}$$

The Eq. 25 is matched with Eq. 8.

Derive equation for likelihood objective: We can re-write Eq. 6 in the form with $\mathbf{x}_0$ prediction (Eq. 9) as below:

$$\begin{aligned} \mathbf{x}_{t-1} &= \frac{1}{\sqrt{\alpha_t}}\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t) + \sigma_t z \\ &= (\frac{1-\alpha_t}{(1-\bar{\alpha})\sqrt{\alpha_t}}\mathbf{x}_t + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t) - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t) + \sigma_t z \\ &= \frac{1-\alpha_t}{1-\bar{\alpha}_t}(\frac{\mathbf{x}_t}{\sqrt{\alpha_t}} - \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\alpha_t}}\epsilon_\theta(\mathbf{x}_t, t)) + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t z \\ &= \frac{(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}(\frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}) + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\mathbf{x}_t + \sigma_t z \end{aligned} \tag{26}$$

The Eq. 26 is matched with Eq. 9.

D. Sampling images

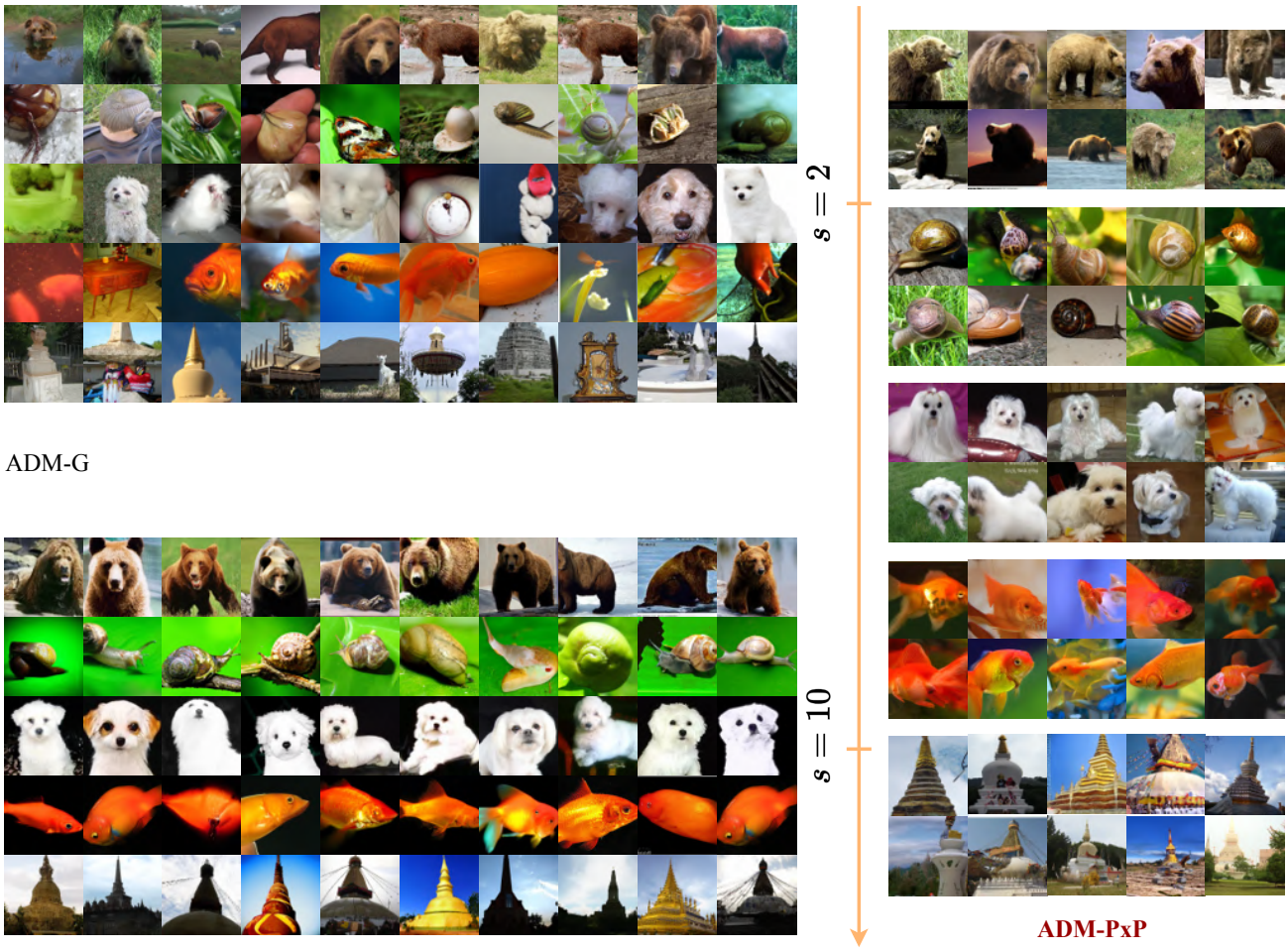


Figure 6. Improvement over baselines. The left figure shows the trade-off between conditional information and the diversity of the vanilla classifier guidance when guidance scale s is increased. The right figure shows the proposed PxP solution to the problem where We both achieve conditional information as well as image quality and diversity. The classes from top to bottom respectively are "Brown bear", "snail", "Maltese dog", "Gold fish" and "Stupa".

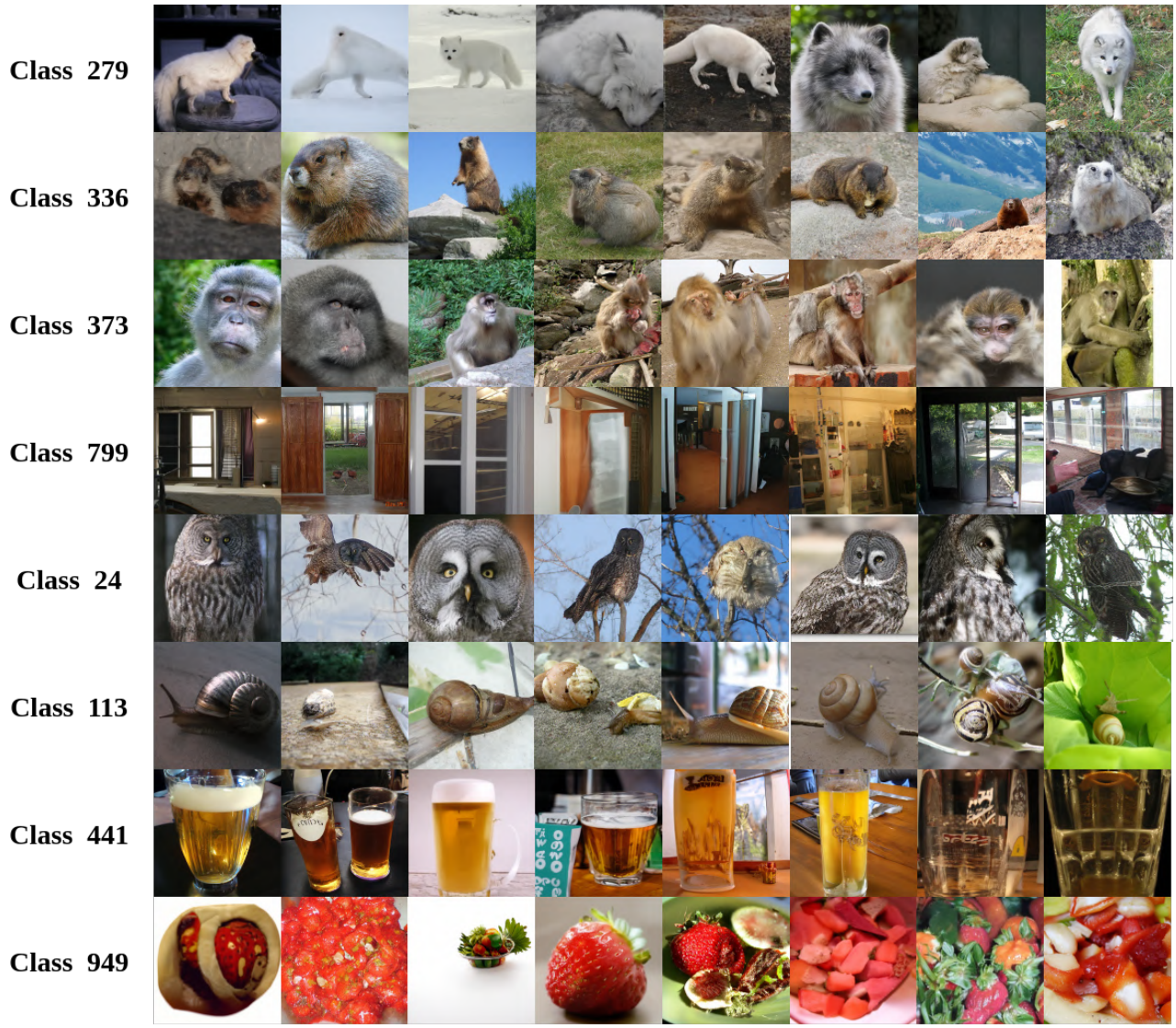


Figure 7. ADM-G on ImageNet128x128 with conditionally pretrained ADM (FID=2.98)

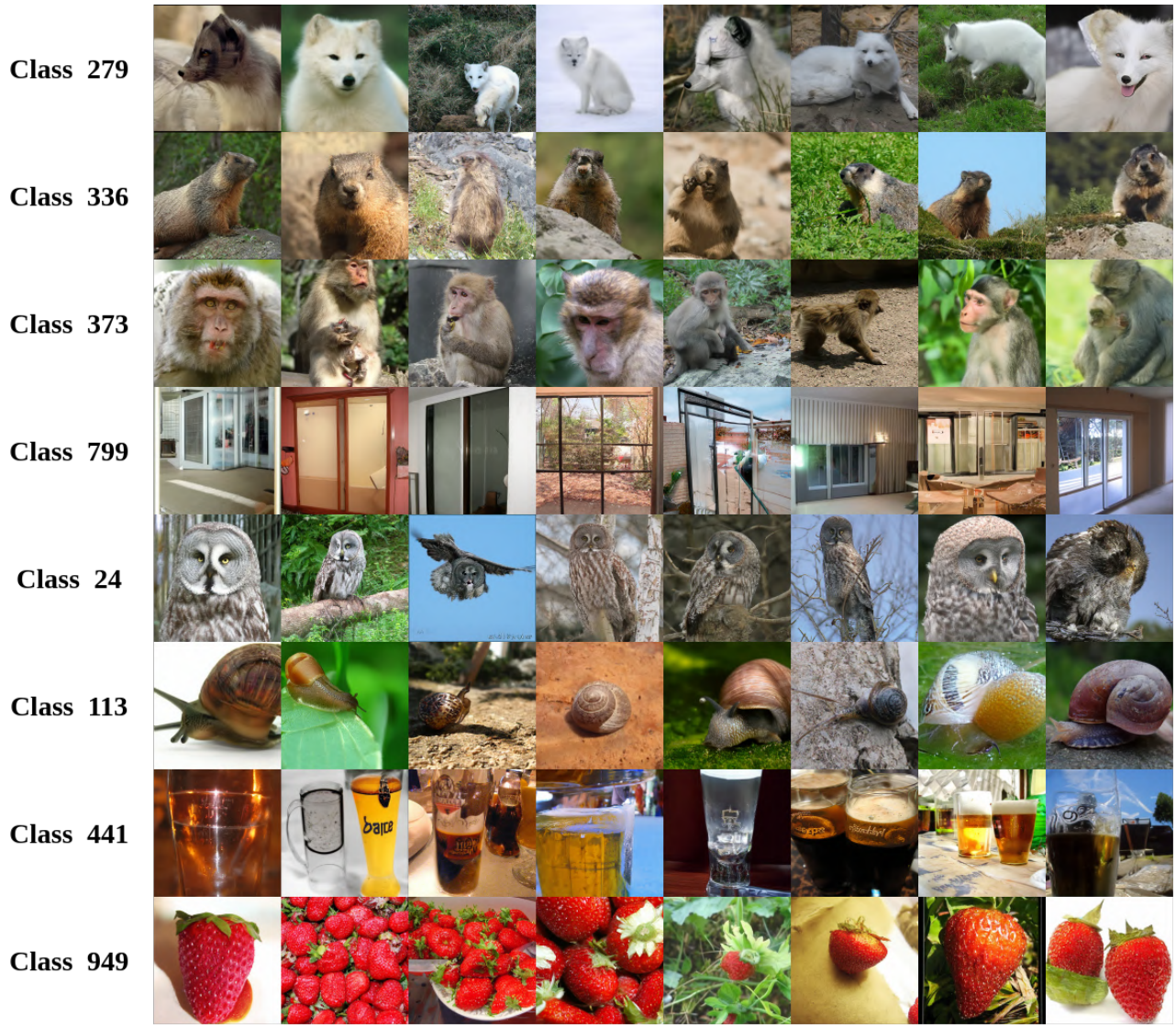


Figure 8. Improvement over Figure 7:ADM-PXP on ImageNet128x128 with conditionally pretrained ADM (FID=2.64)

Figure 9. ADM-PxP on ImageNet256x256 with conditionally pretrained ADM (FID=4.00)