Equilibrium Matching: Generative Modeling with Implicit Energy-Based Models

Runqian Wang
MIT

Yilun Du
Harvard University

Abstract

We introduce *Equilibrium Matching* (EqM), a generative modeling framework built from an equilibrium dynamics perspective. EqM discards the non-equilibrium, time-conditional dynamics in traditional diffusion and flow-based generative models and instead learns the equilibrium gradient of an implicit energy landscape. Through this approach, we can adopt an optimization-based sampling process at inference time, where samples are obtained by gradient descent on the learned landscape with adjustable step sizes, adaptive optimizers, and adaptive compute. Empirically, EqM achieves the best FID among the tested methods, reaching 1.90 on ImageNet 256×256, while remaining competitive on other metrics. Theoretically, we show under standard assumptions that real images are attracting critical points of the learned field and the energy interpretation applies when the field is conservative. Beyond generation, EqM is a flexible framework that naturally handles tasks including partially noised image denoising, OOD detection, and image composition. By replacing time-conditional velocities with a unified equilibrium landscape, EqM offers a tighter bridge between flow and energy-based models and a simple route to optimization-driven inference.

1 Introduction

Generative modeling has advanced rapidly with diffusion and flow-based methods [35, 14, 36, 20, 22], which map simple noise distributions to complex data by defining a forward noising process and learning its reverse. While they achieve state-of-the-art sample quality [28, 5, 15], these models employ non-equilibrium dynamics at both training and inference. Diffusion/flow models are conditioned on the input timestep and learn distinct dynamics for inputs at different noise levels. This imposes practical constraints such as a noise-level schedule and a fixed integration horizon.

Existing approaches that attempt to learn equilibrium dynamics suffer from different problems. Sun et al. [39] have shown that forcing diffusion models to learn equilibrium dynamics by simply removing time (noise[1]) conditioning leads to worse generation quality. Energy-based models (EBMs) [19, 6] directly learn equilibrium energy landscapes, but they often suffer from training instabilities [8] and poor sample quality [6, 29]. More recent approaches such as Energy Matching [2] involve separate training stages and are difficult to scale on large-scale datasets.

In this work, we introduce *Equilibrium Matching* (EqM), a generative modeling framework from an equilibrium perspective. EqM replaces the time-conditional non-equilibrium dynamics of diffusion/flow models with a single time-invariant equilibrium gradient over an implicit energy landscape. We hypothesize that the quality degradation in noise-unconditional diffusion models originates from the incompatibility between the target gradient and equilibrium dynamics. To this end, we introduce a new family of target gradients that align with an implicit energy function. We also provide model variants that explicitly learn this energy function.

By learning a single equilibrium dynamics, Equilibrium Matching supports *optimization-based sampling*, where samples are obtained by gradient descent on the learned landscape. Unlike

[1] In the context of this work, time conditioning and noise conditioning are equivalent.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

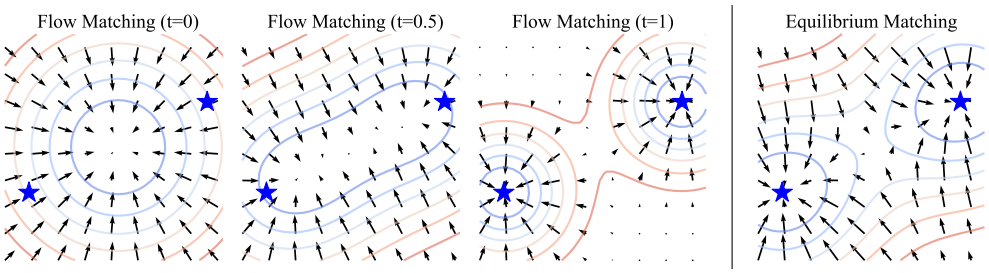

Figure 1: **Conceptual 2D Visualization.** We compare the conceptual 2D dynamics of Equilibrium Matching and Flow Matching under 2 ground truths (marked by stars). **Left**: Flow Matching learns *non-equilibrium* velocity that *varies* over time. **Right**: Equilibrium Matching learns an *equilibrium* gradient that is time-invariant.

existing diffusion samplers that integrate along a prescribed trajectory, optimization-based sampling supports different step sizes and adaptive optimizers. Existing gradient optimization techniques such as Nesterov Accelerated Gradient can be naturally adopted to achieve better generation quality. Equilibrium Matching can also allocate inference-time compute adaptively. It adjusts the sampling steps for each sample independently based on gradient norm and can save up to 60% of function evaluations while retaining similar generation quality.

We validate Equilibrium Matching through both theoretical analysis and empirical evidence. Under standard assumptions, we establish that training samples are local attraction points, and that gradient descent converges under a conservative field. Empirically, EqM achieves the best FID among the tested methods, 1.90 on ImageNet 256×256 generation. EqM also exhibits strong scaling behavior, exceeding the flow-based counterpart at all tested scales. These results suggest that EqM is a promising approach for generative modeling.

Beyond generation, EqM demonstrates additional properties that are not naturally supported by standard diffusion/flow formulations. EqM can generate high-quality samples directly from partially noised inputs, whereas flow-based models without the correct noise level only perform well when starting with pure noise. Moreover, EqM can perform out-of-distribution (OOD) detection without relying on any external module. We also show that different EqM models can be added together to generate compositional images in a similar way as EBMs. These experiments provide evidence for useful equilibrium-style capabilities. We hope that Equilibrium Matching provides a principled way to unify flow-based and energy-based generative models and can inspire better inference-time optimizers for generation in the future.

We summarize our contributions as follows:

- We propose Equilibrium Matching (EqM), a generative framework that learns a time-invariant gradient over an implicit energy landscape.
- We introduce optimization-based sampling, where generation is performed by gradient descent rather than integration, offering more flexibility, better performance, and higher efficiency.
- We analyze EqM under idealized population-optimal assumptions, establishing training point attraction and energy-descent guarantees on conservative fields.
- We empirically validate EqM on image generation, achieving the best FID among the tested methods and competitive on other metrics.
- We show that EqM unlocks capabilities not naturally supported by diffusion/flow models.

2 Preliminaries: Flow Matching

Diffusion and Flow Matching models learn non-equilibrium dynamics. Flow Matching (FM), for example, learns to match the conditional velocity along a linear path connecting noise and image samples. During sampling, Flow Matching starts from pure Gaussian noise and iteratively denoises the current sample using the velocity predicted by f. This process is governed by a differential equation framework, in which the predicted velocity is treated as the time derivative of the desired sampling path and integrated over a total length of 1. Formally, let f denote the generative model, let ϵ be Gaussian noise, let x be a real image sample from the training dataset, and let t be a timestep

sampled uniformly between 0 and 1. The training objective of FM can be written as:

$$L_{\text{FM}} = \big(f(x_t, t) - (x - \epsilon)\big)^2. \tag{1}$$

Here $x_t = tx + (1-t)\epsilon$ is the interpolated sample and $x - \epsilon$ is the target velocity. The model f takes both x_t and t as inputs, where t is the corresponding noise level for x_t.

Noise-Unconditional Model. One approach to make Flow Matching learn equilibrium dynamics is to directly remove t as a conditioning input for f [39]. In this noise-unconditional version, the model is no longer conditioned on the timestep or noise level, giving a new objective:

$$L_{\text{uncond-FM}} = \big(f(x_t) - (x - \epsilon)\big)^2. \tag{2}$$

The only difference is the absence of conditioning on t in f. The sampling and training procedures are otherwise identical to those of the original Flow Matching. However, removing noise conditioning like the above degrades generation quality, and this unconditional variant does not exhibit additional properties that differ from those of the original model.

3 Equilibrium Matching

Equilibrium Matching (EqM) learns a time-invariant gradient field that is designed to be compatible with an underlying energy function, eliminating time/noise conditioning and fixed-horizon integrators. Conceptually (Figure 1), EqM's gradient vanishes on the data manifold and increases toward noise, yielding an equilibrium landscape in which ground-truth samples are stationary points. Flow Matching learns a varying velocity that only converges to ground truths at the final timestep, whereas EqM attempts to learn a time-invariant gradient landscape.

3.1 Training

To train an Equilibrium Matching model, we aim to construct an energy landscape in which the target gradient at ground-truth samples is zero. To do so, we first define a corruption scheme that provides a transition between data and noise. Let γ be an interpolation factor sampled uniformly between 0 and 1, let ϵ be Gaussian noise, and let x be a sample from the training set. Denote $x_\gamma = \gamma x + (1-\gamma)\epsilon$ as an intermediate corrupted sample. Unlike t in FM, our γ is implicit and not seen by the model. Our goal is to define a target gradient at these intermediate samples x_γ that matches an implicit energy landscape. Using a gradient direction that descends from noise to data, we write the Equilibrium Matching training objective as:

$$L_{\text{EqM}} = \big(f(x_\gamma) - (\epsilon - x)c(\gamma)\big)^2, \tag{3}$$

where $c(\gamma)$ controls the gradient magnitude. The target gradient $(\epsilon - x)c(\gamma)$ has direction $(\epsilon - x)$ and magnitude $c(\gamma)$ that vanishes as we get closer to the data manifold. We explicitly enforce $c(1) = 0$, ensuring that the energy landscape has vanishing gradients at real samples. Together, the direction and magnitude result in a target gradient that supports an implicit energy landscape. When $c(\gamma) = 1$, the EqM target is exactly the negation of FM's target.

Compared with Flow Matching, EqM's objective is derived from an EBM perspective rather than a normalizing flow's perspective. More specifically, the EBM interpretation suggests that generation is an optimization procedure where we can use an arbitrary amount of compute to find the minima, as opposed to flow matching, where we have to integrate along a fixed trajectory from the noise to the sample. This difference in perspectives results in different fundamental constraints. EqM requires $c(1) = 0$ to construct an energy landscape with local minima at the data manifold, whereas FM requires $\int_{\gamma=0}^{1} c(\gamma) = 1$ to construct a valid integration path. Our interpretation also results in a different direction of target, where EqM learns the gradient $\epsilon - x$ and FM learns the velocity $x - \epsilon$. We next discuss several simple choices for c in EqM.

Linear Decay. A natural choice for c is a linear function that decays from 1 to 0. In the energy landscape, this is equivalent to assigning noise a high gradient and making the gradient decay linearly to 0 toward the ground-truth image:

$$c_{\text{linear}}(\gamma) = 1 - \gamma. \tag{4}$$

Truncated Decay. Beyond linear decay, we may want the gradient to remain constant when far away from data. This leads to a truncated decay, where the target gradient stays at 1 when $\gamma \leq a$ ($a \in [0, 1)$) and then decays linearly to 0 when approaching the data manifold:

$$c_{\text{trunc}}(\gamma) = \begin{cases} 1, & \gamma \leq a \\ \frac{1-\gamma}{1-a}, & \gamma > a \end{cases}. \tag{5}$$

Piecewise. We can also vary the constant segment of the truncated decay function and set its starting value to b, with $b \in [0, \infty)$. This gives a piecewise function that starts at b, changes linearly to 1 at $\gamma = a$, and then decays linearly down to 0 at $\gamma = 1$:

$$c_{\text{piece}}(\gamma) = \begin{cases} b - \frac{b-1}{a}\gamma, & \gamma \leq a \\ \frac{1-\gamma}{1-a}, & \gamma > a \end{cases}. \tag{6}$$

Gradient Multiplier. The above choices for c have varying magnitudes. Thus, we introduce an additional gradient multiplier λ on top of these gradient fields to control the overall scale. Using linear decay as an example, the final function becomes $c(\gamma) = \lambda c_{\text{linear}}(\gamma) = \lambda(1 - \gamma)$.

3.2 Learning Explicit Energy

Previously, we treated the energy landscape as an implicit underlying structure and learned the gradient of this implicit energy function. We can also modify the Equilibrium Matching model to learn explicit energy values, which we use primarily as a diagnostic. For an explicit energy model g, we match the gradient $\nabla g(x_\gamma)$ at corrupted samples and take $E = g(x_\gamma)$ as the energy at x_γ. This naturally constructs an energy landscape in which real samples are assigned low energy while noisy samples are assigned high energy. The Equilibrium Matching with Explicit Energy (EqM-E) objective can be written as:

$$L_{\text{EqM-E}} = (\nabla g(x_\gamma) - (\epsilon - x)c(\gamma))^2. \tag{7}$$

$\nabla g(x_\gamma)$ is the derivative of g with respect to input x_γ. To obtain a scalar function g, we follow [9] and discuss two different ways to construct g from an existing Equilibrium Matching model f without having to introduce new parameters.

Dot Product. The first approach uses the dot product between the input x_γ and the output $f(x_\gamma)$, defined as $g(x_\gamma) = x_\gamma \cdot f(x_\gamma)$. The corresponding derivative with respect to x_γ is $\nabla g(x_\gamma) = f(x_\gamma) + J_f(x_\gamma)^T x_\gamma$, where $J_f \in \mathbb{R}^{d \times d}$ is the Jacobian of f.

Squared L_2 Norm. The second approach uses the squared L_2 norm of the output $f(x_\gamma)$ with a factor of one half to simplify coefficients: $g(x_\gamma) = -\frac{1}{2}||f(x_\gamma)||_2^2$. The corresponding derivative is $\nabla g(x_\gamma) = -J_f(x_\gamma)^T f(x_\gamma)$.

3.3 Sampling

Because of its equilibrium nature, EqM generates samples via optimization on the learned landscape. In contrast to diffusion/flow models that integrate over a fixed time horizon, EqM decouples sample quality from a prescribed trajectory. It formulates sampling as a gradient descent procedure and supports adaptive step sizes, optimizers, and compute.

Gradient Descent Sampling (GD). A simple way to sample from an EqM model is to apply vanilla gradient descent. Let x_k denote the sample after k steps, then an update step with step size η is:

$$x_{k+1} \leftarrow x_k - \eta \nabla E(x_k), \tag{8}$$

where $\nabla E(x_k)$ is the predicted gradient at x_k and E may be learned implicitly ($\nabla E(x) = f(x)$) or explicitly ($\nabla E(x) = \nabla g(x)$).

Sampling with Nesterov Accelerated Gradient (NAG-GD). Building on gradient descent sampling, we can adopt existing optimization techniques in our sampling procedure. As an example, we use Nesterov Accelerated Gradient [27], which applies a look-ahead step at each update and evaluates the gradient at that look-ahead point:

$$x_{k+1} \leftarrow x_k - \eta \nabla E(x_k + \mu(x_k - x_{k-1})), \tag{9}$$

Algorithm 1 EqM Training

The loss function takes as input model f, noise ϵ (eps), image x, and an interpolation factor γ (g), and returns the EqM loss.

```
def training_loss(f, eps, x, g):
  xg = (1-g) * eps + g * x
  target = (eps-x) * c(g)
  loss = (f(xg) - target)**2
  return loss
```

Algorithm 2 EqM Sampling

The sampling function takes as input pretrained model f, initialization st, step size η, and total steps N, and returns the generated sample.

```
def generate(f, st, eta, N):
  xn = st
  for i in range(N):
    xn = xn - eta*f(xn)
  return xn
```

where μ is the look-ahead factor controlling how far to look ahead at each step.

Sampling with Differential Equations. EqM also naturally supports integration-based samplers. ODE-based diffusion samplers can be viewed as a special case of our gradient-based method. We discuss this further in Section D.

Sampling with Adaptive Compute. Another advantage of gradient-based sampling is that instead of a fixed number of sampling steps, we can allocate adaptive compute per sample by stopping when the gradient norm drops below a certain threshold g_{min}. For step size η and threshold g_{min}, we perform sampling steps $x_{k+1} \leftarrow x_k - \eta \nabla E(x_k)$ until $||\nabla E(x_k)||_2 < g_{\text{min}}$. This mechanism allows the sampler to adaptively adjust the number of steps for each sample and terminate automatically when the current sample is already close enough to the data manifold.

3.4 Implementation

Equilibrium Matching is simple to implement. We provide example pseudocode for training in Algorithm 1 and sampling in Algorithm 2. During training, we first compute an interpolated corrupted sample x_γ from noise ϵ, image x, and factor γ, then train the model f to predict the target gradient $(\epsilon - x)c(\gamma)$ by minimizing a mean squared error objective. At inference time, Equilibrium Matching uses the predicted gradients to iteratively optimize via gradient descent.

We adopt the transformer-based backbone from [25] for our EqM model, using the exact implementation to ensure that no architectural differences influence the results. To remove conditioning on t from this backbone, we set the input t to 0. For further details on model configurations, see Section A.

4 Analysis

We give an idealized population-optimal analysis of EqM and characterize critical points of the learned field.

Assumptions. Let f^* denote the continuous version of the population-optimal EqM field on a finite training set $\mathcal{X} \subset \mathbb{R}^d$ with $d \geq 3$. We assume $c(1) = 0$ and $c(\gamma) \leq C(1-\gamma)$ near $\gamma = 1$. For local-stability statements we specialize to linear decay $c(\gamma) = \lambda(1-\gamma)$. An exact energy-descent result additionally assumes a conservative field $f = \nabla E$, which holds by construction for EqM-E; for implicit EqM we only verify approximate conservativity empirically.

Under these assumptions, we can derive the following statements (see Appendix for derivations and scope):

Statement 1 (Training samples are critical points)**.** *For every $x^{(i)} \in \mathcal{X}$, the population-optimal field satisfies $f^*(x^{(i)}) = 0$. (Derivation in Section C.1)*

Statement 2 (Local attraction and scope of non-data equilibria)**.** *For linear decay, training samples are locally attracting critical points and every zero of f^* lies in $\text{conv}(\mathcal{X})$. A non-data zero is provably unstable under the explicit (k, κ)-resolvability assumption of Section C.2. (Derivation in Section C.2)*

Statement 3 (Gradient descent on an explicit energy converges to stationarity)**.** *For an explicit-energy EqM-E field $f = \nabla E$, if E is L-smooth and bounded below, gradient descent with $\eta \in (0, 1/L]$ satisfies*

$$\min_{0 \leq k < K} \|f(x_k)\|^2 \leq \frac{2\big(E(x_0) - E_{inf}\big)}{\eta K}.$$

(Derivation in Section C.3)



Figure 2: **Curated Samples.** We present curated samples generated by our EqM-XL/2 model.

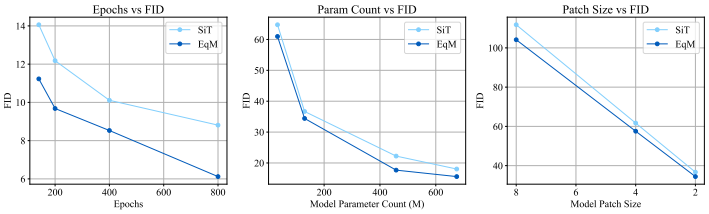

Figure 3: **Scalability of Equilibrium Matching.** EqM scales across training epochs (left), parameter count (middle), and patch size (right), and outperforms Flow Matching at all tested scales by a significant margin.

model	method	FID
StyleGAN-XL	GAN	2.30
VDM++	Diffusion	2.12
DiT-XL/2	Diffusion	2.27
SiT-XL/2	FM	2.06
EqM-XL/2	**EqM**	**1.90**

Table 1: **Class-Conditional ImageNet 256×256 Generation.** EqM-XL/2 achieves the best FID among the tested methods (1.90).

model	sampler	η	μ	FID
SiT-XL/2	Euler (ODE)	0.0040	-	2.10
SiT-XL/2	Heun (SDE)	0.0040	-	2.06
EqM-XL/2	Euler (ODE)	0.0017	-	1.93
EqM-XL/2	GD	0.0017	-	1.93
EqM-XL/2	NAG-GD	0.0017	0.3	**1.90**

Table 2: **Sampler Comparison.** EqM achieves lower FID than the tested Flow Matching baseline using both integration-based ODE sampler and gradient-based samplers.

Statements 1 and 2 analyze the population-optimal field, while Statement 3 relies on a conservative or explicit-energy field. For the trained implicit EqM network, we verify empirically later that the gradient field is roughly conservative in practice.

Empirical Interpretation. We emphasize that these results *do not* imply EqM only memorizes training samples in practice. Similar theoretical conclusions hold for diffusion and Flow Matching, whose generalizations are established empirically. We likewise demonstrate generalization for EqM empirically in Figure 5 of Section 5.1. Complementary precision/recall, density/coverage, and DINOv2-feature metrics are given in Section B.2. The implicit-field association with an energy is approximately true in practice, as quantified in Table 6.

5 Experiments

We validate the practical performance of EqM from four major perspectives. First, we demonstrate the advantages in image generation quality and generalization in practice. Then, we examine the properties and performance of our gradient-based sampling methods, compared against traditional integration-based methods. Next, we illustrate the effectiveness and validity of our gradient landscape design via ablation studies. Finally, we show additional properties of EqM that are not inherently supported by diffusion/flow methods. For details of the training and sampling settings, see Section A.

5.1 Image Generation

ImageNet Results. We report performance on class-conditional ImageNet [4] 256×256 image generation. We compare Equilibrium Matching with prior generative methods, including StyleGAN [34], VDM++ [16], DiT [30], and SiT [25]. Results are shown in Table 1. EqM achieves an FID of 1.90, the best FID among the tested methods. On additional metrics, EqM-XL/2 obtains sFID/IS $4.54/275.7$ versus $4.49/277.5$ for SiT-XL/2. Additional precision/recall, density/coverage, and DINOv2-feature results are reported in Section B.2. We also plot the evolution of FID over time while training our EqM-XL/2 model and compare it with the training FID curve of SiT-XL/2 in Figure 3. EqM consistently improves over the Flow Matching baseline throughout training, showing a consistent FID advantage in the tested ImageNet settings.

Visualizations. We present curated samples from our EqM-XL/2 model in Figure 2 and visualizations of the sampling process in Figure 4. For both EqM and FM, we use XL/2 models trained for 1400 epochs to produce the visualizations. In Figure 4, we observe that EqM converges faster than its FM counterpart at inference. To control for effective update scale, we match the default mean per-step displacement for EqM and FM, so the qualitative difference is not explained only by update

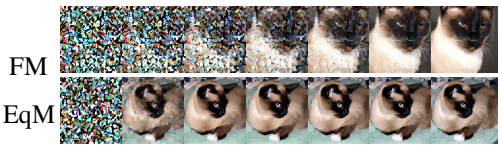

Figure 4: **Sampling Process Visualization.** We present intermediate samples from XL/2 models using matched average step displacement. EqM (bottom) produces realistic images earlier than FM (top).

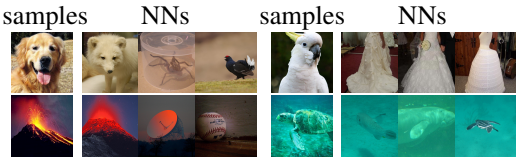

Figure 5: **Nearest Neighbors (NNs) of EqM Samples in the Training Set.** EqM produces samples that are not in the training set, suggesting that it generalizes and does not only memorize the training images.

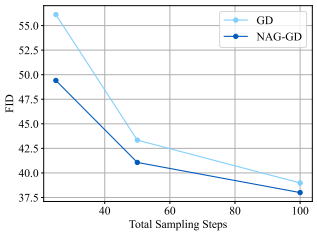

Figure 6: **Sampling with Nesterov Accelerated Gradient.** NAG-GD achieves better sample quality than GD, with the gap being more significant when using fewer steps.

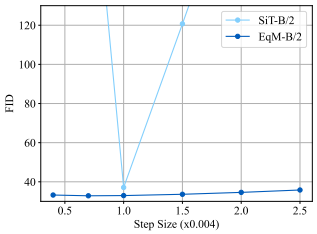

Figure 7: **Different Sampling Step Sizes.** EqM is robust to a wide range of step sizes, whereas Flow Matching only functions properly at one specific step size.

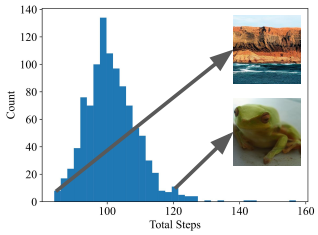

Figure 8: **Total Steps Under Adaptive Compute.** EqM assigns different numbers of steps for different samples, adaptively adjusting compute at inference time.

magnitude. In Figure 5, we show the top-3 nearest neighbors (measured by mean squared distance) in the training set for EqM-generated samples. The nearest neighbors differ from the generated samples, indicating that EqM does not memorize or overfit to the training dataset and can generalize to produce previously unseen samples at inference time.

Scalability. To assess scalability, we vary training length, model size, and patch size. We report the evolution of FID over training using the XL/2 training curves. For model-size scaling, we fix the patch size to 2 and the number of training epochs to 80, and evaluate all four model sizes: S, B, L, and XL. For patch-size scaling, we fix the model size to B and training to 80 epochs, and evaluate patch sizes 8, 4, and 2. As shown in Figure 3, EqM scales well along all axes and consistently outperforms Flow Matching under all tested configurations. These results suggest that EqM has strong scaling potential and is a promising alternative to Flow Matching.

5.2 Inference-Time Experiments

Different Gradient Samplers. We evaluate our proposed gradient-based samplers on ImageNet generation using the EqM-B/2 model. For NAG-GD, we use $\mu = 0.35$, which we found empirically to work best. As shown in Figure 6, NAG-GD yields improved FID across all tested numbers of steps. Moreover, the quality gap increases as the total number of steps decreases. This aligns with our intuition that with fewer total steps, gradient descent requires more assistance to reach a desirable local minimum, making NAG more effective. In Table 2, we also compare traditional integration samplers with the proposed gradient descent samplers on EqM-XL/2. Euler ODE sampler, plain gradient descent, and NAG-GD all exceed the Flow Matching baseline by a large margin, with NAG-GD performing the best. These results show that the NAG technique, commonly used in optimization, is also effective during sampling in Equilibrium Matching, providing further evidence that our approach enables new opportunities at inference time. We also investigate adaptive and stochastic optimizers in Section B.4.

Flexible Step Size. Viewing sampling through an optimization perspective implies that the step size can be adjusted freely. We evaluate this claim by varying the sampling step size on EqM-B/2. For comparison, we also report the performance of the FM baseline, where we use η to replace the ODE update length at each step. We use a total of $N = 250$ sampling steps. From Figure 7, we observe that our EqM model's generation quality remains high and exceeds the Flow Matching baseline across all tested step sizes. By contrast, Flow Matching requires a specific step size of $\eta = 0.004 = \frac{1}{N}$ to function properly, and small fluctuations in step size lead to significantly worse performance. This suggests that EqM constructs a fundamentally different landscape than FM, which enables new sampling schemes not supported by FM models.

$c(\gamma)$	constant (uncond FM [39])	linear	truncated			piecewise	
a	-	-	0.5	0.8	0.9	0.8	0.8
b	-	-	-	-	-	0.8	1.4
FID	40.81	50.47	38.98	**38.34**	41.22	38.84	38.75

Table 3: **Different Training Targets.** Several settings exceed the noise unconditional Flow Matching baseline in performance. Best performance achieved with the truncated decay c_{trunc} and hyperparameter $a = 0.8$.

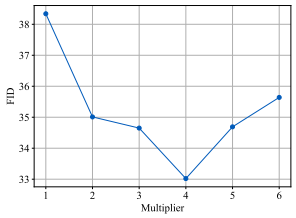

Figure 9: **Different Gradient Multipliers.** $\lambda = 4$ performs the best.

	FM	FM w/ EqM target	uncond FM [39]	EqM
t-cond	yes	yes	no	no
$c(\gamma)$	const	trunc	const	trunc
FID	36.68	41.89	40.81	**32.85**

Table 4: **Noise Conditioning Ablation.** Our truncated decay c_{trunc} improves generation only when not conditioned on noise level, matching our hypothesis.

model	FM	FM	EqM	EqM
sampler	Euler	Heun	GD	NAG-GD
steps	250	250	250	250
wall-clock time	70s	272s	59s	55s
FID	2.10	2.06	1.93	1.90

Table 5: **Sampling Time Comparison.** We compare the wall-clock time for different sampling methods and find EqM is time-efficient against FM.

Adaptive Compute. We test our adaptive compute sampling using a gradient norm threshold of 10 on the EqM-B/2 model. We observe that EqM is able to generate reasonably good samples by adaptive compute, achieving a reasonable FID of 33.79 (32.85 without adaptive compute) using the EqM-B/2 model. We present the distribution of total sampling steps for 1024 samples in Figure 8, which suggests that EqM assigns different inference-time compute for different samples and manages to lower the total compute to 40% of the original compute (original sampling uses fixed 250 steps). Our results offer promising evidence that EqM can enable new inference-time improvements.

5.3 Ablation Study

Hyperparameter Choices. To determine the best target landscape, we search over choices and hyperparameters for $c(\gamma)$. We use our EqM-B/2 model and 80 epochs of training for all hyperparameter experiments. We tune the step size of GD sampler and report the best result for each setting in Table 3. The best-performing gradient landscape is truncated decay with $a = 0.8$. Our results suggest that it is helpful to keep a constant target gradient at the start of the trajectory before decaying to 0. We then sweep the gradient multiplier λ on this best-performing gradient field. As shown in Figure 9, a multiplier of 4 significantly improves FID. We use this setting (c_{trunc} with $a = 0.8$ and $\lambda = 4$) as default in our experiments.

Noise Conditioning. We compare our new target gradient with the standard Flow Matching target under both noise-conditional and noise-unconditional settings in Table 4. We note that two specific ablation settings in Table 4 correspond exactly to FM and unconditional FM [39]. We adopt the 80 epochs B/2 model for both the FM baseline and EqM. Our target gradient improves performance only in the noise-unconditional case, which aligns with our expectation that an energy landscape with zero gradient at real samples is more favorable under equilibrium dynamics.

Sampling Time. For a more quantitative analysis on sampling compute, we report the wall-clock time required for each sampling approach in Table 5. We use the XL/2 model to generate 50,000 samples using identical batch size for all methods, and we report the averaged time per batch of 64 samples on a single H100 GPU. We find that EqM not only achieves better FID but also samples faster than typical differential-equation-based samplers. At equal NFE, part of this difference comes from EqM's time-independent conditioning path (the timestep embedding/adaLN modulation is constant) and from avoiding differential-equation solver overhead.

Energy Formulations. We evaluate two EqM-E variants, the dot product and the L_2 norm, on the EqM-B/4 model. Since L_2 norm measures gradient norm and is intuitively consistent with the EqM objective, we also report L_2 norm results when finetuned from a pretrained EqM model. From Table 7, we find that both energy formulations degrade performance when trained from scratch. We attribute this to the inherent difficulty of optimization, since training EqM-E would require calculating the Hessian and is numerically unstable. We also observe that finetuning an EqM-E L_2 model from a pretrained EqM model closes part of the gap to the implicit baseline and exceeds the performance

noise level γ	scale	0.00	0.25	0.50	0.75	1.00
J asymmetry	$\times 10^{-4}$	2.99	3.59	3.63	2.60	3.00
curl magnitude	$\times 10^{-3}$	2.4	2.9	2.9	2.1	2.4

Table 6: **Conservativity of the Learned Field.** Both Jacobian asymmetry and curl magnitude of the trained EqM model are close to 0 across all noise levels, indicating that the EqM field is approximately curl-free.

model	EqM	EqM-E	EqM-E	EqM-E
objective	-	dot	L_2	finetuned
FID	57.54	73.40	75.53	63.93

Table 7: **EqM-E Variants.** Finetuned EqM-E closes the gap to the baseline and outperforms other energy objectives, suggesting that EqM is inherently consistent with energy training.

model	SVHN	Textures	constant	avg.
Pixel-CNN++	0.32	0.33	0.71	0.45
GLOW	0.24	0.27	-	0.26
IGEBM	**0.63**	0.48	0.39	0.50
EqM	0.55	**0.49**	**1.00**	**0.68**

Table 8: **OOD Detection.** EqM achieves reasonable AUROC under all tested OOD datasets and has the best average result among all tested models.



Figure 10: **Image Composition.** We present compositional samples from EqM-XL/2 each using two ImageNet labels: panda and valley (leftmost), car mirror and volcano (second left), ice cream and chocolate syrup (second right), and broccoli and cauliflower (rightmost).

of both from-scratch results. We therefore view EqM-E primarily as a diagnostic that provides an exactly conservative reference field, rather than as a practical alternative to the implicit EqM model.

Empirical Conservativity of the Learned Field. Our EBM interpretation requires f to be a conservative vector field. To verify whether this holds in practice, we do not materialize the full high-dimensional Jacobian. Instead, we use forward-mode Jacobian-vector products with 256 probe pairs and 64 points per noise level and report Jacobian asymmetry and curl diagnostics. A perfectly conservative field would give zero on both metrics. As shown in Table 6, the raw Jacobian asymmetry remains below 4×10^{-4} and the curl magnitude below 3×10^{-3}. Thus, EqM is approximately conservative and supports an approximate implicit-energy interpretation.

5.4 Additional Properties

In this subsection, we investigate additional properties of Equilibrium Matching that are not naturally supported by standard diffusion/flow formulations.

Partially Noised Image Denoising. By learning equilibrium dynamics, EqM can directly start with and denoise a partially noised image. Existing diffusion/flow models require an explicit noise level as input to process such images, but EqM does not have this limitation. We evaluate EqM's generation quality from partially noised inputs using noised samples from the ImageNet validation set. As shown in Figure 11, EqM behaves very differently from Flow Matching when the latter is not given the noise level. EqM-B/4's FID improves significantly when fed less noisy samples, whereas the Flow-based SiT-B/4 cannot handle partially noised images as raw input and its generation quality drops quickly when not starting from pure noise. An oracle FM/SiT baseline given the true noise level is reported in Section B.2. EqM matches this oracle without requiring the noise level as input.

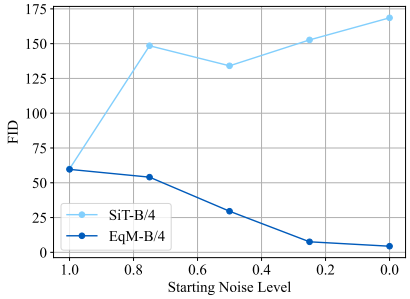

Figure 11: **Partially Noised Image Denoising.** EqM's generation quality improves with less noise while an FM baseline without the noise level drops.

Out-of-Distribution Detection. An additional property of the EqM model is its ability to perform out-of-distribution (OOD) detection using the energy value. In-distribution (ID) samples typically have lower energies than OOD samples. To this end, we use our dot product variant of the EqM-E-B/4 model and perform OOD detection with CIFAR-10 as ID. We use PixelCNN++ [33], GLOW [17], and IGEBM [6] as baselines and adopt the numbers reported by [43]. We report the area under the ROC curve (AUROC) in Table 8. Compared with these baselines, EqM provides reasonable OOD detection across all tested datasets and achieves the best overall performance, which is consistent with the implicit energy field interpretation.

Composition. EqM also naturally supports the composition of multiple models by adding energy landscapes together (corresponding to adding the gradients of each model). We test composition by

combining models conditioned on different ImageNet class labels. We use our EqM-XL/2 model with GD sampler and add two conditional gradients together as the update gradient at each sampling step. We present the generation results in Figure 10, where we observe semantic composition in produced images. We also report quantitative measurement of compositionality in Section B.5. This is similar to the composability of EBMs [7], while the composition of diffusion is significantly more complex to accurately implement [9].

6 Related Work

Diffusion, Flow Matching, and Energy-Based Models. Diffusion models [35, 14, 36, 28, 5, 15] generate images from pure noise through noise-conditioned noising and denoising steps, with sampling typically formulated as integrating a predicted velocity along the noise schedule [14, 37, 24, 15]. Flow Matching [20, 22, 1] replaces the noise schedule with a linear interpolation between noise and data. EBMs [13, 19, 41, 6, 8, 29, 10] learn an energy landscape representing the unnormalized log-density. Their equilibrium structure makes them versatile across modalities and downstream tasks [6, 11], but they usually suffer from training instabilities [3, 38, 12] and are hard to scale [8].

Hybrid Flow/EBM Methods. Several prior works combine diffusion and energy-based training. Sun et al. [39] remove noise conditioning from diffusion models without changing the objective; Energy Matching [2] uses a two-stage scheme that first trains a flow objective and then refines with Langevin dynamics near the manifold. Along with VAPO [23] and Action Matching [26], these methods all retain time- or noise-conditioned objectives over time-dependent trajectories or actions. EqM differs by using a single objective that learns a stationary equilibrium gradient field describing the dynamics both near and away from data. We compare EqM against EBM-style methods in Table 15.

7 Conclusion

We propose Equilibrium Matching, a generative modeling framework that learns equilibrium dynamics in a simple and effective way. EqM achieves competitive generation quality while providing an optimization-based perspective that connects flow and energy-based modeling. Our method is easy to train, achieves strong generation quality, and provides an approximately conservative implicit field (and an exact explicit-energy variant) that supports a wide range of sampling methods. We hope that the equilibrium dynamics learned by EqM will inspire more flexible and scalable learning and inference algorithms in the future.

Acknowledgments

We thank the Harvard Kempner Institute for granting us access to compute resources. We thank Michael Albergo, Hansen Lillemark, and Haldun Balim for helpful discussions. We thank Boyuan Chen for help on distributed training.

References

[1] Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations (ICLR)*, 2023. 10

[2] Michal Balcerak, Tamaz Amiranashvili, Antonio Terpin, Suprosanna Shit, Lea Bogensperger, Sebastian Kaltenbach, Petros Koumoutsakos, and Bjoern Menze. Energy matching: Unifying flow matching and energy-based models for generative modeling. *arXiv preprint arXiv:2504.10612*, 2025. 1, 10, 14

[3] Miguel A Carreira-Perpinan and Geoffrey Hinton. On contrastive divergence learning. In *International workshop on artificial intelligence and statistics*, pages 33–40. PMLR, 2005. 10

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009. 6

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1, 10

[6] Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in neural information processing systems*, 32, 2019. 1, 9, 10, 14

[7] Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation with energy based models. *Advances in Neural Information Processing Systems*, 33:6637–6647, 2020. 10

[8] Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy based models. *arXiv preprint arXiv:2012.01316*, 2020. 1, 10

[9] Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International conference on machine learning*, pages 8489–8510. PMLR, 2023. 4, 10

[10] Ruiqi Gao, Yang Song, Ben Poole, Ying Nian Wu, and Diederik P Kingma. Learning energy-based models by diffusion recovery likelihood. *arXiv preprint arXiv:2012.08125*, 2020. 10

[11] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019. 10

[12] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010. 10

[13] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. 10

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 10

[15] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35: 26565–26577, 2022. 1, 10, 14

[16] Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023. 6

[17] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018. 9

[18] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL `https://www.cs.toronto.edu/~kriz/cifar.html`. 14

[19] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006. 1, 10

[20] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 10

[21] Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024. 14

[22] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 10

[23] Junn Yong Loo, Michelle Adeline, Arghya Pal, Vishnu Monn Baskaran, Chee-Ming Ting, and Raphaël C.-W. Phan. Learning energy-based generative models via potential flow: A variational principle approach to probability density homotopy matching. *Transactions on Machine Learning Research*, 2025. 10

[24] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 10

[25] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024. 5, 6, 14

[26] Kirill Neklyudov, Rob Brekelmans, Daniel Severo, and Alireza Makhzani. Action matching: Learning stochastic dynamics from samples. In *International conference on machine learning*, pages 25858–25889. PMLR, 2023. 10

[27] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983. 4

[28] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning (ICML)*. PMLR, 2021. 1, 10

[29] Erik Nijkamp, Mitch Hill, Tian Han, Song-Chun Zhu, and Ying Nian Wu. On the anatomy of mcmc-based maximum likelihood learning of energy-based models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5272–5280, 2020. 1, 10

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 6

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention (MICCAI)*, 2015. 14

[32] Tim Salimans and Jonathan Ho. Should ebms model the energy or the score? In *Energy based models workshop-ICLR 2021*, 2021. 14

[33] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017. 9

[34] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022. 6

[35] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 1, 10

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1, 10

[37] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 10

[38] Yunfu Song and Zhijian Ou. Learning neural random fields with inclusive auxiliary generators. *arXiv preprint arXiv:1806.00271*, 2018. 10

[39] Qiao Sun, Zhicheng Jiang, Hanhong Zhao, and Kaiming He. Is noise conditioning necessary for denoising generative models? *arXiv preprint arXiv:2502.13129*, 2025. 1, 3, 8, 10

[40] Runqian Wang and Kaiming He. Diffuse and disperse: Image generation with representation regularization. *arXiv preprint arXiv:2506.09027*, 2025. 14

[41] Jianwen Xie, Yang Lu, Song-Chun Zhu, and Yingnian Wu. A theory of generative convnet. In *International conference on machine learning*, pages 2635–2644. PMLR, 2016. 10

[42] Xiulong Yang, Qing Su, and Shihao Ji. Towards bridging the performance gaps of joint energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15732–15741, 2023. 14

[43] Sangwoong Yoon, Young-Uk Jin, Yung-Kyun Noh, and Frank Park. Energy-based models for anomaly detection: A manifold diffusion recovery approach. *Advances in Neural Information Processing Systems*, 36:49445–49466, 2023. 9

Appendix

In Section A, we show the detailed experimental settings. In Section B, we present additional experiments including CIFAR-10 experiments, additional evaluation metrics, sampling dynamics visualization, additional adaptive and stochastic optimizers, and quantitative compositionality measurements. In Section C, we show the derivations of the statements made. In Section D, we demonstrate the relationship between ODE-based integration sampler and our gradient descent sampler.

A Experimental Setting

A.1 Training Setting

We present the training setting of our Equilibrium Matching model in Table 14.

A.2 Inference Setting

We present the majority of our sampler settings in Table 14. More specifically, in Section 5.1, we adopt the SiT results reported by Ma et al. [25] and Wang and He [40], and use our own NAG-GD sampler for EqM results. In Section 5.2, Section 5.3, and Section 5.4, we adopt the Euler sampler for SiT experiments and the vanilla GD sampler for EqM experiments.

B Additional Experiments

B.1 CIFAR-10 Experiments

We also evaluate our approach on non-transformer architectures on the CIFAR-10 dataset [18], where the commonly used network architecture is U-Net [31]. The experiments are based on the publicly available code of Flow Matching [21]. We use the same hyperparameters as the original codebase, with the only difference being that we do not use EDM scheduling or skewed timesteps, which are tricks designed specifically for flow matching training [15]. Table 15 presents the FID results. We used the truncated decay function with $a = 0.4, \lambda = 2.0$ for our EqM model. Equilibrium Matching outperforms Flow Matching baseline, similar to our observations on SiT. This demonstrates that EqM is a general generative modeling approach applicable across different datasets and model backbones.

Since it has been hard to directly train EBMs on ImageNet and most existing literature only reports CIFAR-10 numbers, we compare other related methods' performance on CIFAR-10 in Table 15, including IGEBM [6, 42], Energy-based U-net [32], and Energy Matching [2]. EqM outperforms all listed methods, making it a promising approach for generative modeling. As an additional cross-dataset/backbone check, on Fashion-MNIST with a U-Net and 600 epochs, EqM obtains FID-10k 7.48 versus 11.32 for FM under the same reference implementation.

B.2 Additional Metrics

We report additional evaluation metrics, including sFID and Inception Score (IS), in Table 16. Equilibrium Matching achieves competitive sFID and IS compared with other generative methods. We further evaluate distributional quality, adaptive stopping, and partially noised oracle controls below.

Distributional metrics. Using the identical XL/2 protocol, EqM improves or matches SiT on several complementary distributional metrics, as summarized in Table 9. In DINOv2 feature space, EqM also achieves lower Fréchet distance and higher precision and coverage. These results complement FID rather than establish a distribution-level theorem.

Adaptive quality–compute curve. For EqM-B/2, adaptive stopping produces the quality–compute trade-off shown in Table 10. A fixed 250-step EqM baseline obtains an FID-10k of 36.4. These results support a principled per-sample stopping criterion rather than a claim that adaptive stopping uniformly improves FID at every compute budget.

metric	EqM	SiT
Precision ($k = 3$)	0.806	0.802
Recall ($k = 3$)	0.609	0.597
Density ($k = 5$)	1.130	1.143
Coverage ($k = 5$)	0.9997	0.9997
Mean per-class coverage	0.943 (0.068)	0.943 (0.070)
DINOv2 Fréchet distance	101.3	117.3
DINOv2 precision	0.640	0.606
DINOv2 coverage	0.760	0.714

Table 9: **EqM achieves competitive additional metrics.** Distributional metrics for EqM and SiT under the identical XL/2 protocol. Parentheses denote the standard deviation of per-class coverage.

mean NFE	FID-10k
657	36.0
241	36.5
168	38.1
141	39.6
250 (fixed)	36.4

Table 10: **Quality–compute trade-off for adaptive stopping on EqM-B/2.**

Partially noised oracle control. Finally, we evaluate partially noised inputs at several starting noise levels. As shown in Table 11, EqM closely matches an oracle SiT model that is explicitly given the true noise level, while substantially outperforming SiT without this input. Thus, EqM approximately recovers oracle-level performance without requiring the noise level as an input.

B.3 Energy Evolution

We include an energy evolution plot over the sampling process for our EqM-E model in Figure 12. We used the B/4 model with dot product parameterization and report the average energy curve over 64 samples. We see that the energy value consistently and smoothly decreases over the course of sampling, which is consistent with the expected behavior of an EBM.

B.4 Adaptive and Stochastic Optimizers

Adaptive optimizer. We also attempt to directly adopt existing adaptive optimizers for EqM sampling. In particular, we use Adam without momentum ($\beta_1 = \beta_2 = 0$). We find that Adam

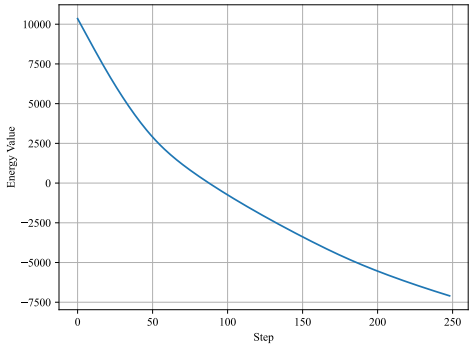

Figure 12: **Energy Evolution During Sampling.** We report the energy curve during sampling on EqM-E. The energy value decreases smoothly and consistently, matching our expectation.

	starting noise level		
method	0.8	0.4	0.0
EqM	59.2	20.2	6.2
SiT w/o noise-level input	125.0	164.2	138.1
Oracle SiT	58.2	20.2	5.4

Table 11: **EqM matches oracle FM without time input.** FID-10k for partially noised inputs. Oracle SiT is provided the true starting noise level, whereas EqM and standard SiT are not.

setting	top-5 accuracy
single-class control	0.2%
EqM composed samples	1.8%

Table 12: **EqM composes classes with higher accuracy.** To measure composition, we report ResNet-50 top-5 classification on both labels of EqM composed image and compare against a single-class control paired with a random second class. EqM's composed samples provide much stronger signal to classification on both classes.

performs decently well on EqM-B/2, as shown in Table 17. This opens exciting opportunities for future research by demonstrating that reasonable samples can still be produced when the gradients are normalized.

Stochastic sampling. We additionally test whether the learned EqM field is compatible with stochastic sampling. On EqM-B/2, NAG-GD slightly improves both FID-10k and recall over deterministic gradient descent, while annealed Langevin sampling attains the same recall with a modest FID increase (Table 13).

B.5 Measuring Compositionality

To measure compositionality of EqM models in a quantitative way, we report ResNet-50 top-5 classification on both labels of EqM composed image and compare against a single-class control paired with a random second class. As reported in Table 12, across 100 random class pairs with 20 samples each, 1.8% of composed samples contain both parent classes in a ResNet-50 top-5 prediction, compared with 0.2% for a single-class control paired with a random second class. EqM therefore composes class labels in a semantically meaningful way detectable by classifiers.

C Derivations

C.1 Learned Gradient at Ground-Truth Samples

Statement 1 (Population-Optimal Field at Training Samples)**.** *Let f^* be the population-optimal EqM field, let $x^{(i)} \in \mathcal{X} \subset \mathbb{R}^d$, and suppose $c(1) = 0$ with $c(\gamma) \le C(1-\gamma)$ near 1. For the continuous version of the conditional-mean field and $d \ge 3$, $f^*(x^{(i)}) = 0$.*

Derivation of Statement 1. At the population optimum,

$$f^*(z) = \mathbb{E}[(\epsilon - x)c(\gamma) \mid x_\gamma = z].$$

Because conditioning on an exact point is a zero-measure event, we make the argument through shrinking balls. Let $u = 1 - \gamma$ and condition on $B_r(x^{(i)})$. On the branch generated by $x = x^{(i)}$, the event implies $\|u(\epsilon - x^{(i)})\| \le r$. Since $c(1-u) \le Cu$,

$$\|(\epsilon - x^{(i)})c(1-u)\| \le Cr.$$

This branch has probability $\Theta(r)$ as $r \to 0$. For every other training point, the corrupted sample has a smooth d-dimensional density around $x^{(i)}$, so its probability of entering the ball is $O(r^d)$ and its conditional target first moment is bounded. Therefore

$$\left\| \mathbb{E}[(\epsilon - x)c(\gamma) \mid x_\gamma \in B_r(x^{(i)})] \right\| = O\big(C(r + r^{d-1})\big) \xrightarrow[r \to 0]{} 0.$$

sampler	FID-10k	Recall
GD	20.42	0.627
NAG-GD	20.34	0.634
Annealed Langevin	21.11	0.634

Table 13: **EqM supports stochastic sampling.** Stochastic sampling results for EqM-B/2. Annealed Langevin sampling uses a cosine schedule with $T_0 = 0.01$.

model	S/2	B/2	L/2	XL/2
model configurations				
params (M)	33	130	458	675
depth	12	12	24	28
hidden dim	384	768	1024	1152
patch size	2	2	2	2
heads	6	12	16	16
ImageNet training configurations				
epochs	80	80	80	80 - 1400
batch size	256			
optimizer	AdamW			
optimizer β_1	0.9			
optimizer β_2	0.999			
weight decay	0.0			
learning rate (lr)	1×10^{-4}			
lr schedule	constant			
lr warmup	none			
gradient field hparams				
choice of $c(\gamma)$	truncated decay			
$c(\gamma)$ multiplier λ	4.0			
a	0.8			
b	-			
integration sampler configurations				
sampler	dopri5	dopri5	dopri5	Euler
NFE	250			
η	-	-	-	0.0017
gradient sampler configurations				
sampler	GD/NAG-GD			
steps	250			
η	0.003	0.003	0.003	0.0017
μ	0.35	0.35	0.35	0.3

Table 14: **Equilibrium Matching Configurations.**

The continuous version of the population-optimal field hence satisfies $f^*(x^{(i)}) = 0$. This avoids assigning positive probability to the event $\gamma = 1$. □

C.2 Local Attraction and Non-Data Equilibria

Statement 2 (Scope of Population-Optimal Equilibria)**.** *For linear decay $c(\gamma) = \lambda(1-\gamma)$, the population-optimal field is*

$$f^*(z) = \lambda\big(z - \mathbb{E}[x \mid x_\gamma = z]\big).$$

Hence every zero lies in $\mathrm{conv}(\mathcal{X})$, and an isolated training point is a locally attracting critical point. For non-data zeros we make the resolvability assumption below; no claim is made for arbitrary non-resolvable zeros.

model	FM	IGEBM	Energy-based U-net	Energy Matching	EqM
FID	3.70	37.9	6.8	3.34	**3.32**

Table 15: **Image Generation on CIFAR-10.** We report FID on CIFAR-10 class-conditional image generation with U-Net. EqM improves upon Flow Matching and energy-based methods on CIFAR-10 generation.

model	method	FID ↓	sFID ↓	IS ↑
StyleGAN-XL	GAN	2.30	**4.02**	265.1
VDM++	Diffusion	2.12	-	267.7
DiT-XL/2	Diffusion	2.27	4.60	**278.2**
SiT-XL/2	FM	2.06	4.49	277.5
EqM-XL/2	**EqM**	**1.90**	4.54	275.7

Table 16: **Class-conditional Generation on ImageNet 256×256 with Additional Metrics.** Equilibrium Matching achieves the best FID and relatively good sFID and IS.

Derivation and scope of Statement 2. For $u = 1 - \gamma$, linear decay makes the target

$$(\epsilon - x)c(\gamma) = \lambda u(\epsilon - x) = \lambda(x_\gamma - x),$$

which gives the posterior-mean form above. At a zero $\hat{x}$, $\hat{x} = \mathbb{E}[x \mid x_\gamma = \hat{x}]$, so $\hat{x}$ lies in the convex hull of the training set. Near an isolated training point the posterior concentrates on that point, yielding a positive local Jacobian for f^* and therefore a contractive gradient-descent update for sufficiently small step size.

Resolvability assumption. A non-data zero $\hat{x}$ is (k, κ)-resolvable if, for some constant $c_0 > 0$, all but an $e^{-c_0 d}$ fraction of the posterior mass lies on k training points with distances $D_j = \|x^{(j)} - \hat{x}\|$ satisfying $\max_j D_j / \min_j D_j \le \kappa$.

Let v be the top spread direction of these supporting components and $a_j = v^T(x^{(j)} - \hat{x})$. The balance condition gives $\sum_j w_j a_j = 0$. A per-branch Laplace evaluation gives

$$\mathbb{E}[(1-u)/u^2 \mid j] = \frac{d}{D_j^2}\big(1 + O(1/d)\big),$$

and the spread bound gives $\sum_j w_j a_j^2 \ge (\sum_j w_j D_j^2)/(k-1)$. Consequently the directional Jacobian is bounded, up to the cross term from unequal D_j, by

$$v^T J_{f^*}(\hat{x}) v \;\lesssim\; \lambda\left(1 - \frac{d}{(k-1)\kappa^2}\right) + O(\kappa - 1).$$

For equidistant supporting components ($\kappa = 1$) the cross term vanishes, so any resolvable non-data zero with $k - 1 < d$ has a negative-curvature/expansive direction and is not a stable minimum; the two-component case is $k = 2$. For κ sufficiently close to one, the same conclusion holds by perturbation. Thus the theory rules out stable non-data equilibria only under this explicit resolvability condition. □

C.3 Convergence of Gradient-Based Sampling

Statement 3 (Convergence of Gradient-Based Sampling on an Explicit Energy)**.** *Let E be an explicit EqM-E energy and $f = \nabla E$. Suppose E is L-smooth and bounded below by $E(x) \ge E_{inf}$. Then, gradient descent with step size $\eta \in \big(0, \frac{1}{L}\big]$ satisfies:*

$$\min_{0 \le k < K} \|f(x_k)\|^2 \le \frac{2\big(E(x_0) - E_{inf}\big)}{\eta\, K},$$

where x_k is the iterate after k steps and K is the total number of optimization steps performed. For implicit EqM, this is an approximate energy-descent interpretation.

sampler	GD	NAG-GD	Adam
FID	32.85	32.97	36.35

Table 17: **EqM Sampling with Adam.** We find that Adam without momentum achieves decent generation quality on EqM-B/2.

Derivation of Statement 3. By L-smoothness of E, for any $x, y \in \mathbb{R}^d$,

$$E(y) \;\le\; E(x) + \nabla E(x)^\top (y - x) + \frac{L}{2}\,\|y - x\|^2.$$

Take $y = x_{k+1} = x_k - \eta\, f(x_k)$ and recall $\nabla E(x_k) = f(x_k)$. Then

$$\begin{aligned} E(x_{k+1}) &\le E(x_k) - \eta\, f(x_k)^\top f(x_k) + \tfrac{L}{2}\,\eta^2\,\|f(x_k)\|^2 \\ &= E(x_k) - \eta\Big(1 - \tfrac{L\eta}{2}\Big)\|f(x_k)\|^2 \;\;\le\;\; E(x_k) - \tfrac{\eta}{2}\|f(x_k)\|^2, \end{aligned}$$

where the last inequality holds because $\eta \le 1/L \implies 1 - \frac{L\eta}{2} \ge \frac{1}{2}$.

Summing from $k = 0$ to $k = K - 1$ gives

$$E(x_K) - E(x_0) \;\le\; -\tfrac{\eta}{2} \sum_{k=0}^{K-1} \|f(x_k)\|^2 \;\le\; -\tfrac{\eta}{2}\, K \min_{0 \le k < K} \|f(x_k)\|^2.$$

Since $E(x_K) \ge E_{\inf}$, rearrange to obtain

$$\min_{0 \le k < K} \|f(x_k)\|^2 \;\le\; \frac{2\big(E(x_0) - E_{\inf}\big)}{\eta\, K},$$

as required. □

D Relation between Integration-Based and Optimization-Based Sampling

D.1 ODE Sampling as a Special Case of Optimization

Consider the ODE

$$\dot{x} = v(x),$$

and its explicit (forward) Euler discretization with a uniform time grid on $[0, 1]$:

$$x_{k+1} = x_k + h\, v(x_k), \qquad h = \tfrac{1}{N},\; k = 0, \dots, N - 1.$$

When the velocity field is conservative with potential E, i.e., $v(x) = -\nabla E(x)$, the Euler step becomes:

$$x_{k+1} = x_k - h\, \nabla E(x_k),$$

which is exactly a gradient-descent update with step size $\eta = h = \frac{1}{N}$. Hence, with N steps on a unit time horizon, the gradient-descent sampler with $\eta = \frac{1}{N}$ coincides with the explicit Euler ODE sampler.

D.2 General Integration Samplers

The equivalence above suggests a broader correspondence: integration-based samplers can be interpreted as optimization-based methods by viewing the velocity as a descent direction. In particular, when $v(x) = -\nabla E(x)$, any time integrator induces an optimization update rule.

Practically, ODE samplers in generative modeling are often implemented on a uniform grid with step size $h = \frac{1}{N}$. Under the optimization view, we are not bound to this constraint. We can adopt adaptive step sizes while retaining the same underlying direction field. This perspective enables a direct adaptation of existing integration-based samplers within Equilibrium Matching.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We conduct extensive experiments in the Experiments section to validate our claims.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the assumptions on high dimensionality and conservativeness in Section 4 Analysis and verify that these assumptions hold in practice in Section 5 Experiments.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide the assumptions in Analysis and provide the full correct proof in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We disclose our code to reproduce the main experiments and provide all details needed for experiments in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide a well documented codebase with readme detailing reproduction procedures.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We include all details in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We conduct experiments across at least 3 scales to verify statistically consistent improvement.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We detail the compute instance we used for our wall-clock time experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We faithfully follow the NeurIPS code of ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our work focus on improving generative modeling method on image generation, which is a well established area and we strictly follow prior literature.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not involve high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We credit all assets used in this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide sufficient documentation for our codebase.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing or human subjects are involved.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.