A Pseudo Code of Proposed Algorithm

Algorithm 1: Energy-Calibrated VAE Algorithm.

input : Learning iterations T, number of MCMC steps K, observed examples $\{\mathbf{x}_i\}_{i=1}^N$, network optimizer $\mathcal{Q}$.
output: Estimated parameters $\phi = \{\alpha, \beta, \theta\}, \omega$.
for $t = 0 : T-1$ **do**

Primal-Step:
1. **Mini-batch**: Sample observed examples $\{\mathbf{x}_i\}_{i=1}^n$
2. **Mini-batch**: Sample generated examples $\{\hat{\mathbf{x}}_i\}_{i=1}^n$, where $\hat{\mathbf{x}}_i \sim p_{\beta,\theta}(\hat{\mathbf{x}})$
3. **Sample $\tilde{\mathbf{x}}$ by MCMC**: For each generated $\hat{\mathbf{x}}_i$, sample $\tilde{\mathbf{x}}_i$ using Eq. (3) for K steps
4. **Learning** E_ω: $\omega_{t+1} = \mathcal{Q}(\nabla_{\omega_t} \hat{\mathcal{L}}(\omega_t), \omega_t)$
5. **Learning and Calibrating VAE** ϕ_t: $\phi_{t+1} = \mathcal{Q}(\nabla_{\phi_t} \hat{\mathcal{L}}(\phi_t) + \lambda \nabla_{\phi_t} \hat{\mathcal{L}}_{\text{con}}(\phi_t), \phi_t)$

Dual-Step:
6. **Update** λ: update λ according to Eq. (4)

B More Details in Zero-Shot Image Restoration

Typical image restoration tasks usually have simple $\mathbf{A}$ and $\mathbf{A}^\dagger$, we give some examples following:

Colorization. $\mathbf{A} = [1/3, 1/3, 1/3]$ converts each RGB channel pixel $[r, g, b]^T$ into a grayscale value $[r/3 + g/3 + g/3]$. A simple pseudo-inverse $\mathbf{A}^\dagger$ is $\mathbf{A}^\dagger = [1, 1, 1]^T$, that satisies $\mathbf{A}\mathbf{A}^\dagger\mathbf{A} = \mathbf{A}$.

Super Resolution. For SR with scale n, we can set $\mathbf{A} \in \mathbb{R}^{1\times n^2}$ as a average-pooling operator $[1/n^2, \cdots, 1/n^2]$. A simple pseudo-inverse $\mathbf{A}^\dagger \in \mathbb{R}^{n^2\times 1}$ is $\mathbf{A}^\dagger = [1, \cdots, 1]^T$.

Inpainting. For $\mathbf{A}$ is a mask operator, simply let $\mathbf{A}^\dagger = \mathbf{A}$, we can have $\mathbf{A}\mathbf{A}^\dagger\mathbf{A} = \mathbf{A}$.

C Experiment Setting Details in Image Generation

Evaluation Metric. We employ the FID score as the metric in most experiments. We compute the FID score between 50k generated samples and training images. On CelebA HQ we compute 30k generated samples and training images due to the dataset only containing 30k images.

Dataset Details. For STL-10, we follow the procedure in AutoGAN and DC-VAE, and resize the STL-10 images to 32×32.

VAE architecture. The backbone similar to that proposed in FastGAN [35] is used in experiments.

Ablations in Tab. 1. The VAE and Flow use the same architecture as EC-VAE and EC-Flow, except the EBM is not needed in VAE and Flow.

Energy-Calibrated Variational Learning. We use five MCMC steps for calibrating the posterior. We employ exactly the same architecture of VAE and EBM (only need for EC-VAE and EC-VAE w/ calibrated posterior) for all variants. Both the MSE and ELBO are measured on the test set of CIFAR-10.

Hyper-parameters. Given a large number of datasets, heavy compute requirements, and limited computational resources, we don't optimize the hyper-parameters carefully. Even if the backbone is not carefully selected or designed, it's expected that we can use other well-designed backbones to easily get better results. Also, we just roughly choose the learning rate from 1e-4 or 2e-4 in all experiments. Following most previous work [10, 32, 57], we utilize the Langevin Dynamics with a low temperature (e.g., 1e-5) and select a small step size from 1e-6 or 1e-5. On all datasets, we train EC-VAE using the Adam optimizer. For a fair comparison, EMA is applied for the VAE component for all variants in our ablations. For all other training details (e.g., detailed model architecture), we refer readers to our full code, which will be released after published.

D Experiment details in Zero-Shot Image Restoration

We use the well-trained **Energy-Calibrated VAE** in zero-shot image restoration by the proposed application method in Sec. 4.3 with 50 MCMC steps.

E Experiment Details in Ablation Study

Setting Details. The same architectural design of VAE's encoder, decoder, and energy function is consistently applied across all variants of EC-VAE in our ablation studies. For VAE+GAN, the same architectural design of VAE is applied, and we directly use the architecture of energy function as the discriminator. For EBM the same architectural design of energy function is applied. For EBM nit w/ VAE samples, the same architectural design of all networks is applied. For EC-VAE w/ flow prior, we use a glow parameterized by MLPs as the architecture. For EC-VAE w/ LPIPS, we only use LPIPS in modeling $p_\beta(\mathbf{x}|\mathbf{z})$, as we found the benefit of using LPIPS in modeling calibration loss is negligible. We employed a replay buffer size of 10,000 for training the EBM, with a distribution of 5% sampling from noise and 95% sampling from the replay buffer. Regarding the EBM initialized with VAE samples, a similar approach was adopted, utilizing the same replay buffer size of 10,000, 5% sampling from the VAE, and 95% sampling from the replay buffer. Notably, in the absence of a replay buffer during the training of the EBM initialized with VAE samples, we were unable to produce reasonable generation outcomes.

E.1 Additional Ablations

Comparison to CoopVAEBM. We provide an extra comparison with CoopVAEBM [56] which is a cooperative learning approach for training VAE and

Table 11: Additional Quantitative results on CIFAR-10.

Model	FID↓	MSE↓	ELBO↑
CoopVAEBM w/ MCMC [56]	22.08	-	-
CoopVAEBM w/o MCMC [56]	26.17	0.0276	130.35
EC-VAE (ours)	**5.20**	**0.0193**	**652.59**

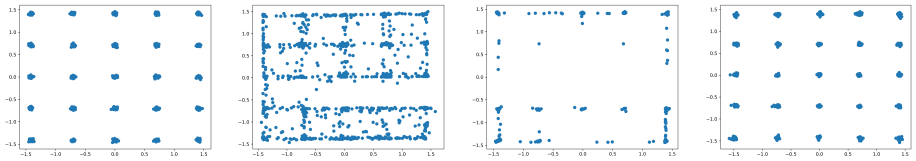
Table 12: Additional Quantitative results on CIFAR-10.

Model	FID↓
EC-VAE	**5.20**
EC-VAE w/o EMA	8.02
EC-VAE w/ unconditional EBM	5.89

EBM. As discussed in related work section (Sec. 2.1), the existing cooperative learning approach trains the base generative model (i.e., VAE) solely using generated samples, leading to biased learning. To further emphasize the necessity of training the base generative model (i.e., VAE) upon both real data and generated samples with adaptive weight as proposed in our **EC-VAE**, we train the CoopVAEBM using the same architecture as EC-VAE to ensure a fair comparison instead of directly using the reported results in their original paper which might be influenced by architectural differences. As shown in Tab. 11, our EC-VAE outperforms CoopVAEBM by a large margin in terms of both FID, MSE, and ELBO. Additionally, it is observed that the MCMC still performs an important role in improving the generation quality of CoopVAEBM. These results indicate that our proposed EC-VAE is more effective with more accurate likelihood learning and better generation quality.

Effect of Exponential Moving Average (EMA). The EMA technique has been extensively incorporated into prior generative models [25, 44, 46], which demonstrates its widespread applicability and effectiveness. In alignment with these findings, we found the EMA can also enhance the performance of our model, EC-VAE, as indicated by the improved FID scores. As detailed in Tab. 12, the application of EMA to EC-VAE results in a noteworthy reduction in the FID score from 8.02 to 5.20 on CIFAR-10.

Effect of Conditional Energy-Based Models. Unlike the previous cooperate approach [6, 54, 56] which utilize unconditional EBMs for obtaining MCMC-revised samples, we utilize conditional EBMs to produce calibrated samples. Conditional EBMs have the advantage of constraining the high-density regions to be localized around the condition $\mathbf{x}$. This not only simplifies the calibrating process for the VAE but also enables us to focus on maximizing the conditional likelihood of the calibrated samples given condition samples via the VAE's decoder. Otherwise, an additional inference step would be necessary to infer the related latent variables of the MCMC-revised samples, as discussed in CoopVAEBM [56]. Empirical evidence, as presented in Tab. 12, supports the efficacy



(a) Real Samples (b) Samples by VAE (c) Samples by GAN (d) Samples by EC-VAE

Fig. 4: Qualitative results on the 25-Gaussians dataset.

of conditional EBMs; the FID deteriorates from 5.20 to 5.89 when an unconditional EBM is employed, highlighting the superiority of the conditional approach.

F Extra Study on Mode Coverage

We evaluate the mode coverage of our model on the popular 25-Gaussians. This 2D toy dataset is also used in previous work [52]. We train our EC-VAE with 15 MCMC steps and compare it to other models in Fig. 4. The VAE's encoder, decoder, and energy function both consist of four linear layers with 256 hidden units. The latent dimension is 20. We observe that the vanilla VAEs can not produce good samples, lots of samples are significantly out of modes. In contrast, our EC-VAE successfully calibrated the sampling distribution with all modes covered and high-quality samples. We also train a GAN with similar networks for comparison. We observe that GAN suffers severely from mode collapse. Moreover, as the true distribution is available, we also evaluate the likelihood of 100000 generated samples by models with true data density. Our EC-VAE obtains **-1.07 nats** which significantly improves the likelihood obtained by VAE which is **-1.78 nats**. For reference, the likelihood of 100000 real data under real density is **-1.04 nats**.

G Additional Qualitative Results

We present additional qualitative results in this section. Note that all images in this section are **un-selected**.

Qualitative Comparison to Other models. We provide a qualitative comparison to other models on CIFAR-10 in Fig. 5. We use the official checkpoint from CoopFlow to generate its samples and use its official code to compute the corresponding FID score. For VAEBM w/o MCMC, we take the samples from its paper's appendix.

Please see if drop MCMC steps at test time sampling, how better our proposed method is compared to previous cooperative learning model (i.e., CoopFlow) and previous work that combines generative model and EBMs (i.e., VAEBM). And we note that CoopFlow is the most advanced model among existing cooperative learning models. Also, many existing works (e.g., CoopFlow and VAEBM)

related to EBMs need extra hyper-parameters tuning stage to achieve high performance or will lead to a significantly worse result. And it can be seen in Fig. 5d that after carefully tuning the hyper-parameters, although the FID is better than the untuned, the color of samples by CoopFlow tends to be over-saturated, which is not the property of real data. In contrast, our proposed Energy-Calibrated VAE can even drop MCMC steps at test time sampling while achieving high-quality samples.

Qualitative Results. From the interpolation results in Fig. 12, we conclude that our model has a good, smooth latent space. We show the nearest neighbors from the training dataset with our generated samples in Fig. 13. It is easy to see our generated images are significantly different from images from the training dataset, which concludes that our model does not simply remember the data.

Additional Zero-Shot Image Restoration samples are in Fig. 6.

Additional samples on CelebA 64 are in Fig. 7.

Additional samples on LSUN Church 64 are in Fig. 8.

Additional samples on STL-10 are in Fig. 9.

Additional samples on ImageNet 32 are in Fig. 10.

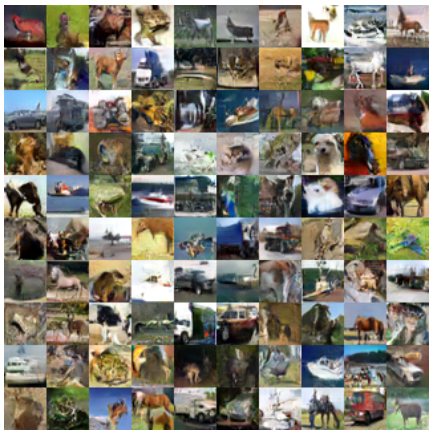
Additional samples on CelebA-HQ-256 are in Fig. 11.

Interpolation results on CelebA-HQ-256 are in Fig. 12.

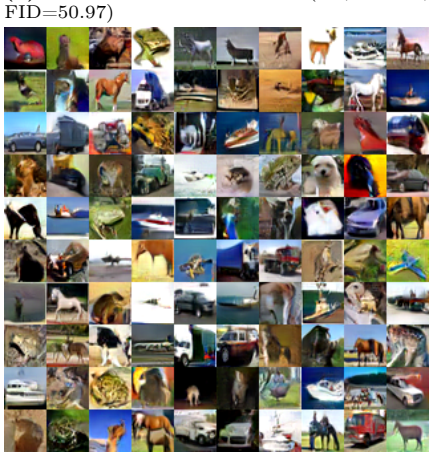
Samples on CelebA-HQ-256 and their nearest neighbors from the training dataset are in Fig. 13.

(a) CoopFlow without MCMC (FID=79.45)

(b) VAEBM without MCMC (i.e., NVAE, FID=50.97)



(c) CoopFlow w/o extra tuning hyper-parameters (FID=26.54)



(d) CoopFlow w/ carefully tuned hyper-parameters (FID=15.80)

(e) Real Data

(f) **EC-VAE** (FID=5.20)

Fig. 5: Qualitative comparison of Energy-Calibrated VAE (Ours) and other models on CIFAR-10. Samples are un-selected.

Fig. 6: Additional Zero-Shot Image Restoration (Colorization, 4× SR) Samples on CelebA-HQ-256. Samples are uncurated.

Fig. 7: Additional samples from CelebA 64. Samples are uncurated.

Fig. 8: Additional samples from LSUN Church 64. Samples are uncurated.

Fig. 9: Additional samples from STL-10 which are uncurated.

Fig. 10: Additional samples from ImageNet 32 which are uncurated.

Fig. 11: Additional samples from CelebA-HQ-256. Samples are uncurated.

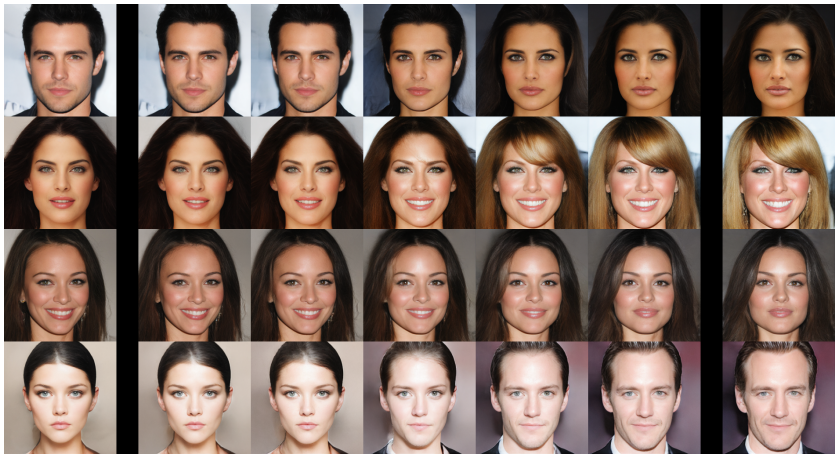


Fig. 12: Interpolation results in latent space on CelebA-HQ-256.

Fig. 13: Generated images (left) and their nearest neighbors in VGG's feature space from the CelebA-HQ-256 training dataset.