VAEBM: A SYMBIOSIS BETWEEN VARIATIONAL AUTOENCODERS AND ENERGY-BASED MODELS

Zhisheng Xiao, Karsten Kreis, Jan Kautz, Arash Vahdat




VARIATIONAL AUTO-ENCODERS (VAES)

A quick recap

- Assume data distribution is modeled by $p(x) = \int p(x|z)p(z)\,dz$
  - $p(x|z)$ is the decoder distribution, and $p(z)$ is the prior
  - Training would be easy if we have access to the posterior $p(z|x)$, but this is intractable in general
- Resort to use variational inference, where a variational posterior $q(z|x)$ is introduced as an approximation to the true posterior, resulting in the variational lower bound:

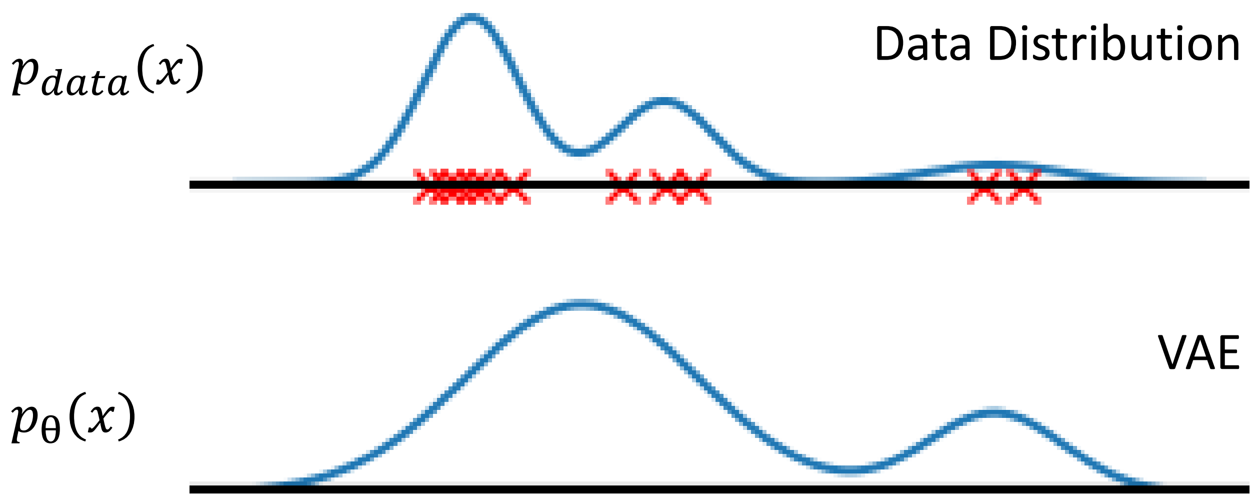
$$\log p(x) \geq \underbrace{\mathbb{E}_{q(z|x)}[\log p(x|z)]}_{\text{Reconstruction Term}} - \underbrace{KL(q(z|x)\,||\,p(z))}_{\text{KL Regularization Term}}$$

- Recently, large VAEs such as NVAE* and VDVAE** with carefully designed network structures and hierarchical latent variables achieve impressive results in likelihood modeling, but their sample qualities are still limited.

* NVAE: A Deep Hierarchical Variational Autoencoder, Vahdat and Kautz.
** Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images, Child

WHAT'S WRONG WITH VAES?

VAEs tend to assign high probabilities to non-data like regions!



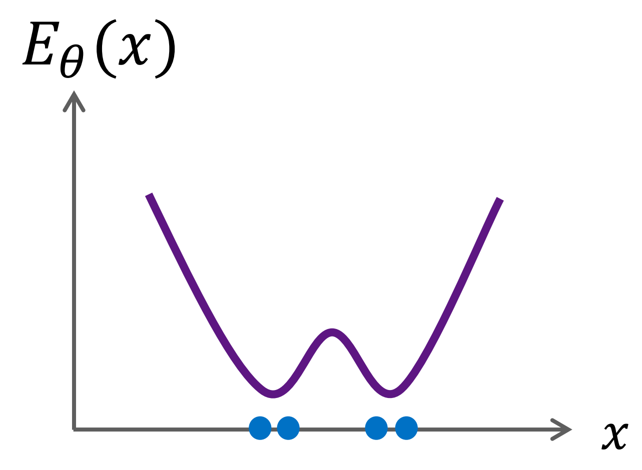
NVAE

$t = 1.$

ENERGY-BASED MODELS (EBMS)

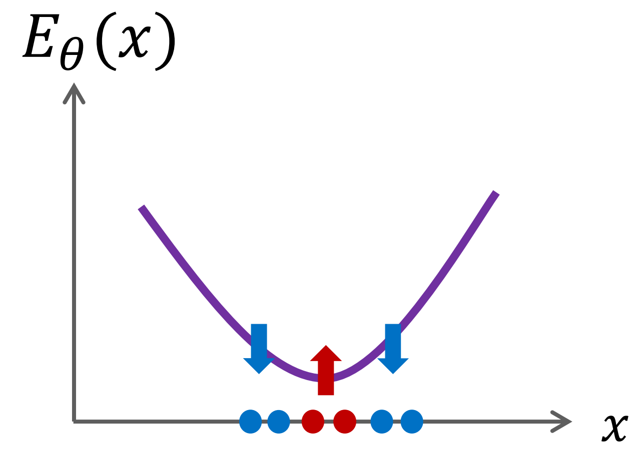
A quick introduction

- Assume data distribution is modeled by $p_{\text{EBM}}(x) = \frac{1}{Z} e^{-E_\theta(x)}$
  - where $E_\theta : \mathcal{X} \rightarrow \mathbb{R}$ is an energy function implemented by neural networks
  - Z is the normalization constant



- Maximum likelihood training:

$$\nabla_\theta \mathbb{E}_{x\sim p_{data}(x)} [\log p_{\text{EBM}}(x)] = -\underbrace{\mathbb{E}_{x\sim p_{data}(x)}[\nabla_\theta E_\theta(x)]}_{\text{Training Samples}} + \underbrace{\mathbb{E}_{x\sim p_{\text{EBM}}(x)} [\nabla_\theta E_\theta(x)]}_{\text{Model Samples}}$$



- Sampling from model is often done by Markov chain Monte Carlo (MCMC) sampling

VAES VS. EBMS

A comparison

Energy-based Models (EBMs):

- ☺ Explicitly push down the densities of non-data like regions
- ☺ Almost no constrain on the energy function (unlike normalizing flows)
- ☹ Slow sampling during training and test due to expensive MCMC steps

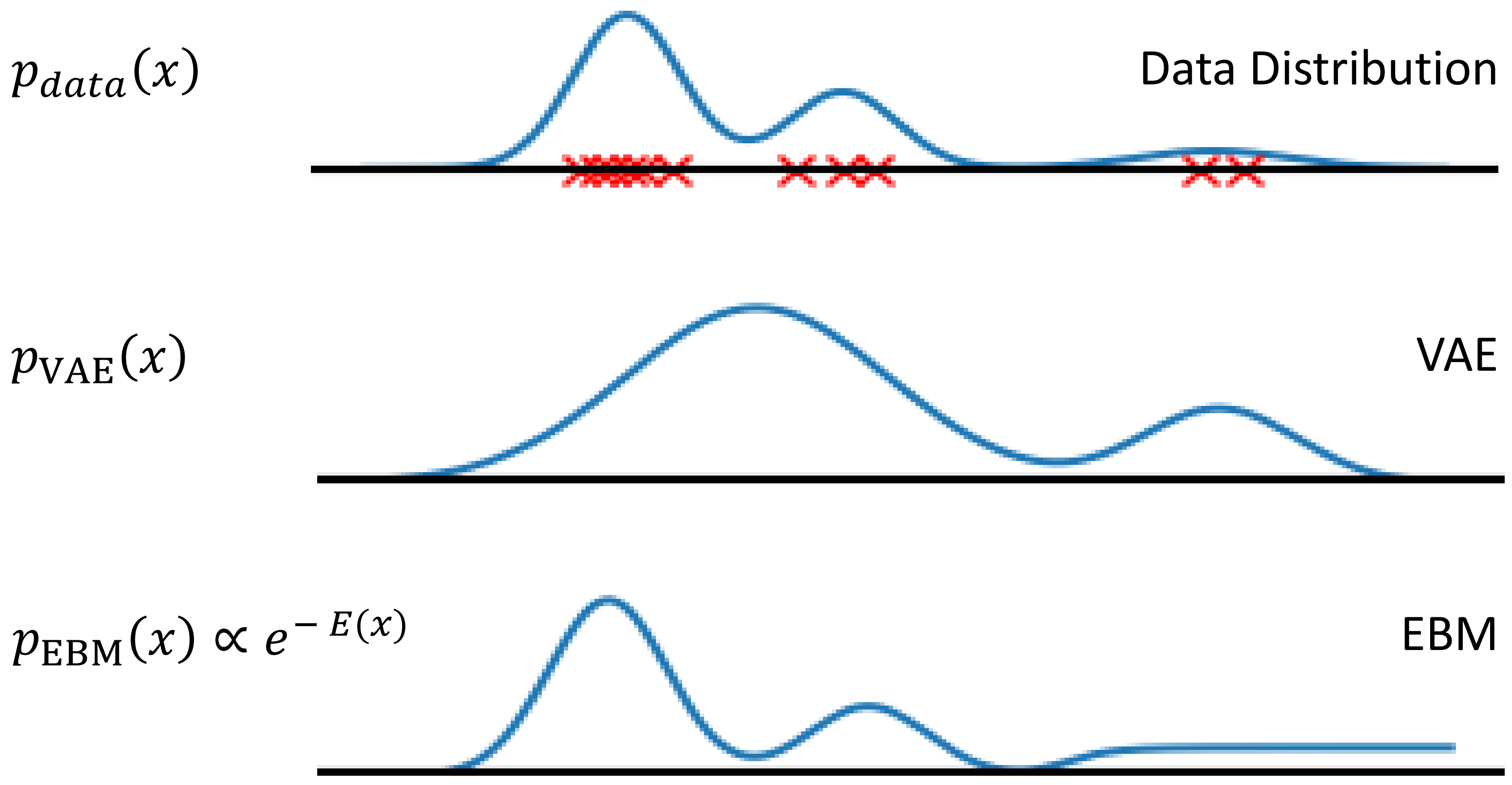
Variational Autoencoders (VAEs):

- ☺ Fast sampling, easy train
- ☺ Latent embedding allows fast traversal in data space
- ☹ High probabilities for non-data-like regions in the data space

VAEBM: A symbiotic composition of VAEs and EBMs

VAEBM

The basic idea



$$p_{\mathrm{VAEBM}}(x) \propto p_{\mathrm{VAE}}(x)\, p_{\mathrm{EBM}}(x)$$

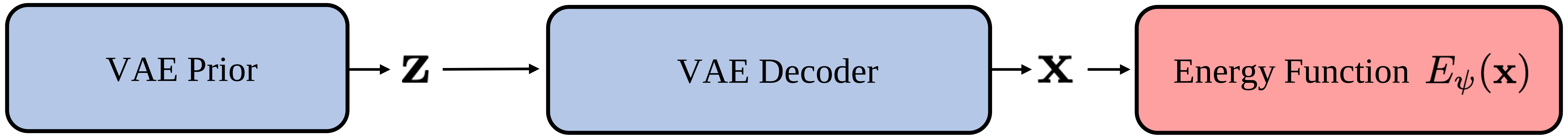
Global Structures

Fine-grained Details

By taking the product of the densities of a VAE and an EBM, we want the VAE to capture the global structures of data, and the EBM to refine the distribution by pushing down the densities of non-data-like regions.

VAEBM

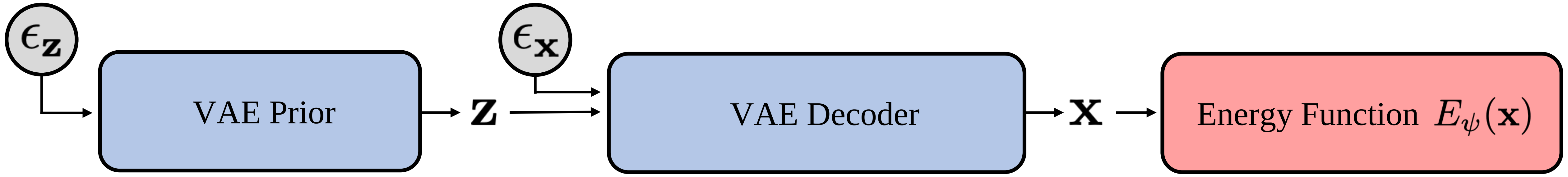
Conceptual Visualization



$$p_{\text{VAEBM}}(x) \propto p_{\text{VAE}}(x)\, p_{\text{EBM}}(x)$$

VAEBM

Training



$$p_{\mathrm{VAEBM}}(x) = \frac{1}{Z}\, p_{\mathrm{VAE}}(x)\, p_{\mathrm{EBM}}(x)$$

$$\log p_{\mathrm{VAEBM}}(x) = \underbrace{\log p_{\mathrm{VAE}}(x)}_{\text{Stage (1) train VAE}} + \underbrace{\log p_{\mathrm{EBM}}(x) - \log Z}_{\text{Stage (2) train EBM}}$$

Stage (1) train VAE: easy with reparam. trick ☺

Stage (2) train EBM: requires MCMC sampling ☹

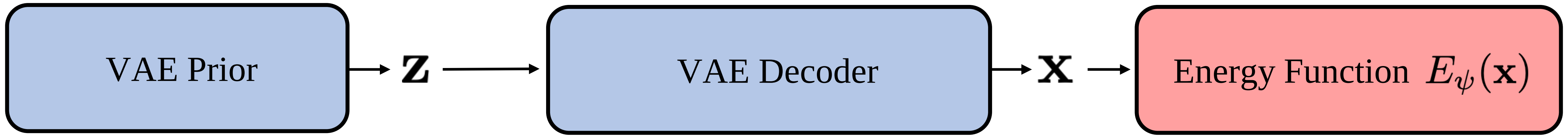
$x = T(\epsilon_z, \epsilon_x)$

Run MCMC in the ϵ space* ☺

* NeuTra-lizing Bad Geometry in Hamiltonian Monte Carlo Using Neural Transport, Hoffman et al.

VAEBM

Two stage training



Stage (1) train VAE

Stage (2) train EBM

A symbiotic composition:

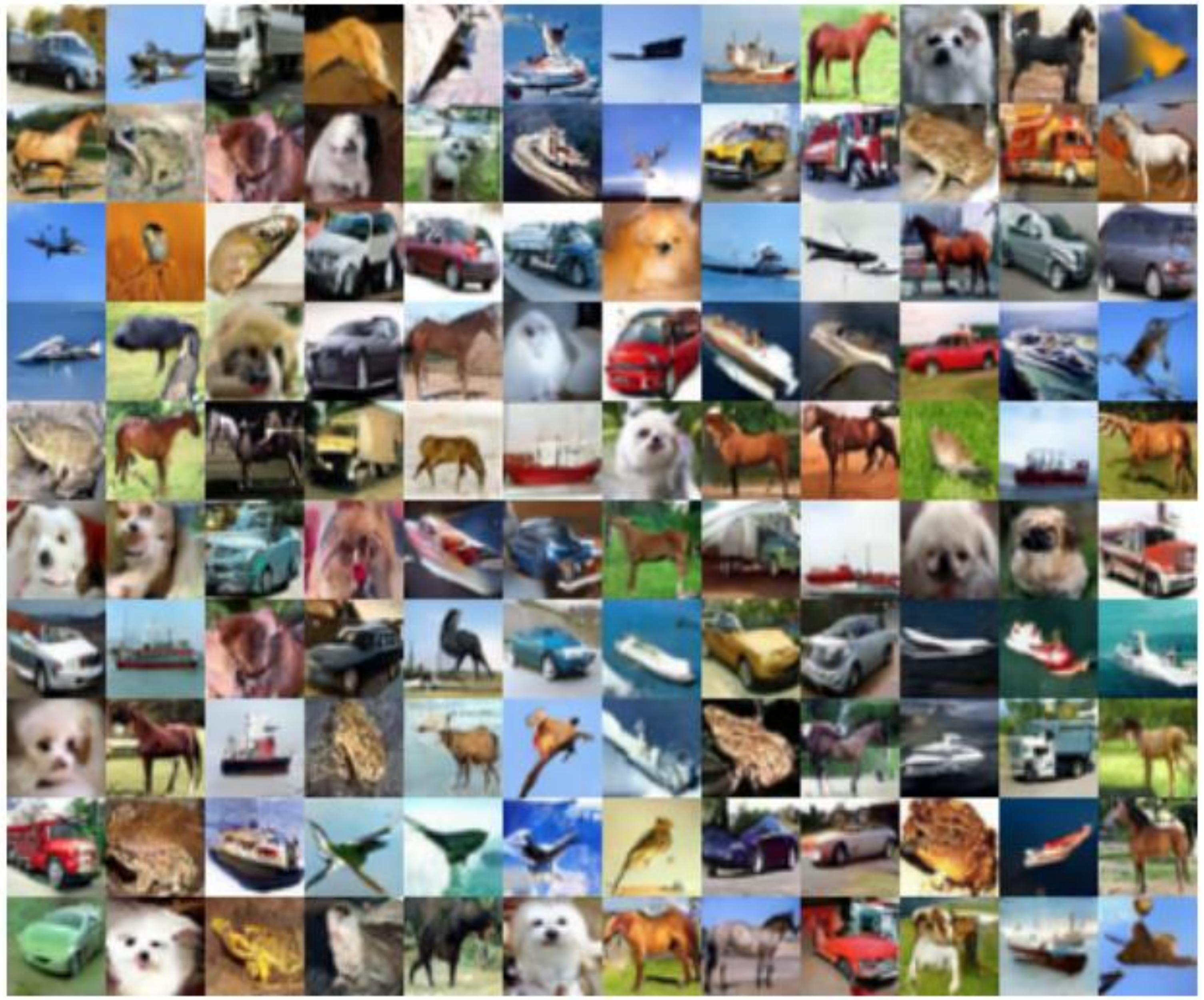
- VAE learns the overall mode structure
- VAE provides re-parametrization for MCMC sampling from EBM
- EBM helps VAE to exclude non-data-like regions
- MCMC steps are expensive, but VAEBM requires very few training epochs for EBM

CIFAR10

Use NVAE as the base VAE

NVAE (t = 1)

VAEBM

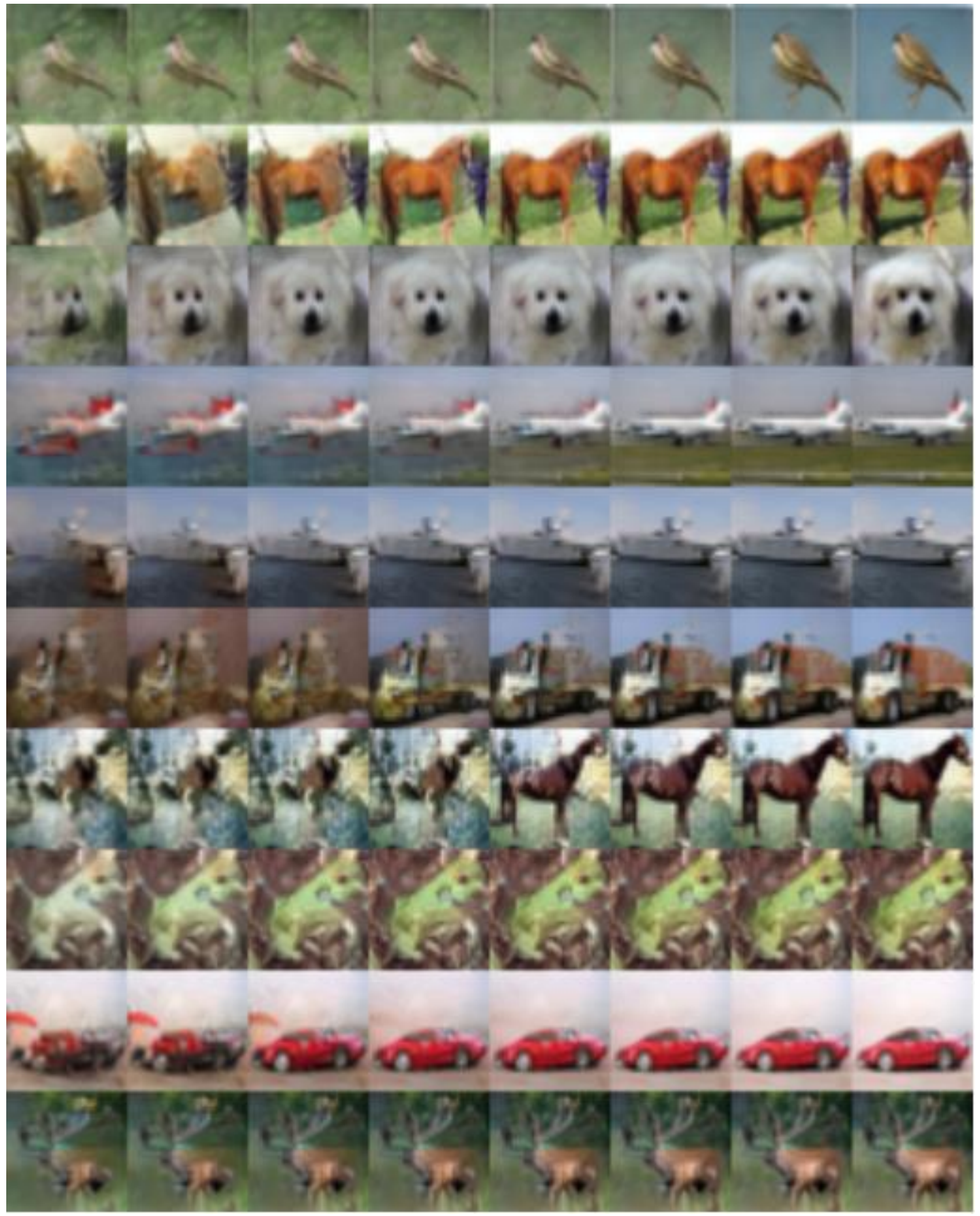


16 MCMC STEPS

CIFAR-10

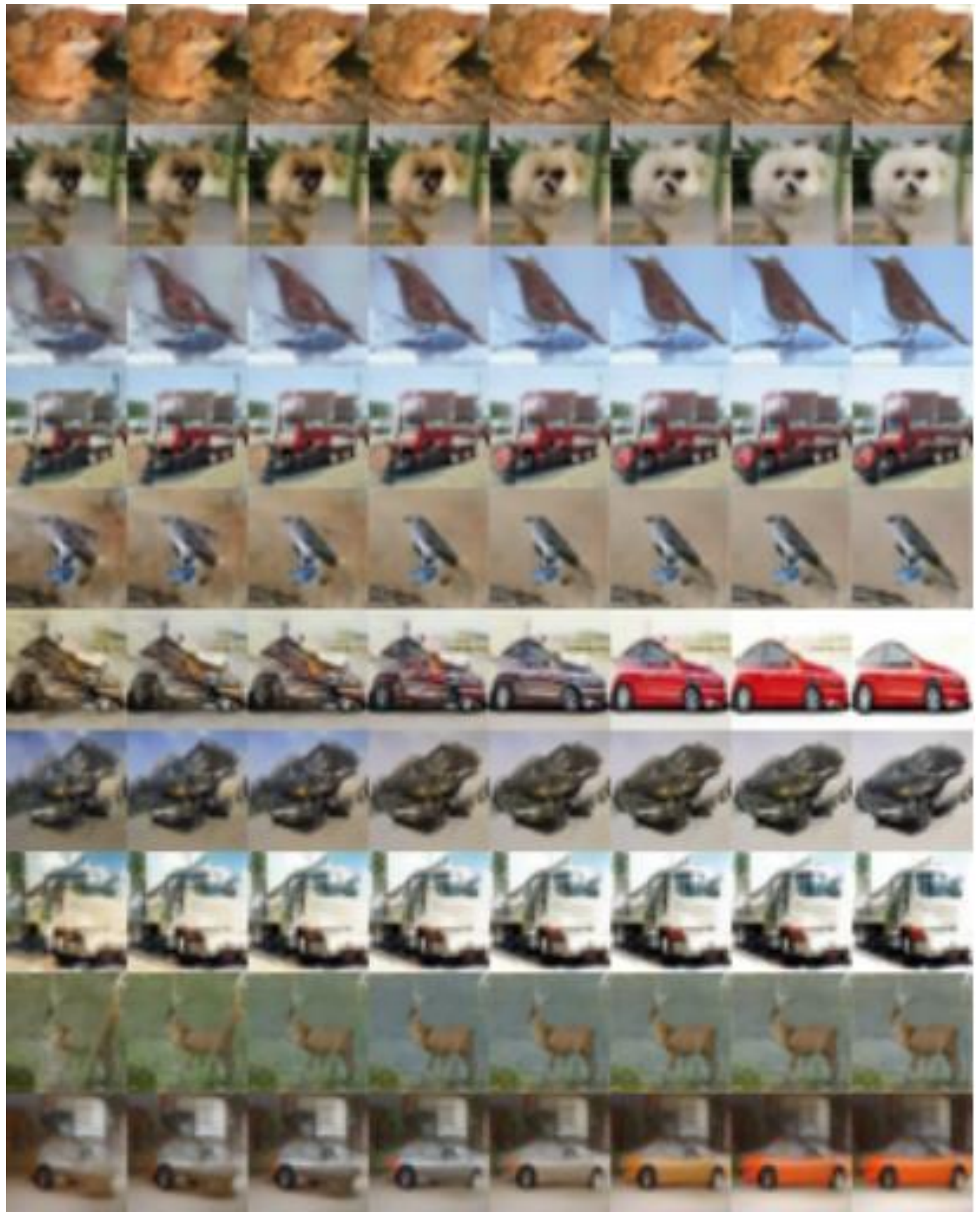
16-step MCMC

NVAE → VAEBM



16-step MCMC

NVAE → VAEBM



QUANTITATIVE RESULTS

CIFAR-10 (unconditional)

	Model	IS↑	FID↓
Ours	VAEBM w/o persistent chain	8.21	12.26
	VAEBM w/ persistent chain	8.43	12.19
EBMs	IGEBM (Du & Mordatch, 2019)	6.02	40.58
	EBM with short-run MCMC (Nijkamp et al., 2019b)	6.21	-
	F-div EBM (Yu et al., 2020a)	8.61	30.86
	FlowCE (Gao et al., 2020)	-	37.3
	FlowEBM (Nijkamp et al., 2020)	-	78.12
	GEBM (Arbel et al., 2020)	-	23.02
	Divergence Triangle (Han et al., 2020)	-	30.1
Other Likelihood Models	Glow (Kingma & Dhariwal, 2018)	3.92	48.9
	PixelCNN (Oord et al., 2016b)	4.60	65.93
	NVAE (Vahdat & Kautz, 2020)	5.51	51.67
	VAE with EBM prior (Pang et al., 2020)	-	70.15
Score-based Models	NCSN (Song & Ermon, 2019)	8.87	25.32
	NCSN v2 (Song & Ermon, 2020)	-	31.75
	Multi-scale DSM (Li et al., 2019)	8.31	31.7
	Denoising Diffusion (Ho et al., 2020)	9.46	3.17
GAN-based Models	SNGAN (Miyato et al., 2018)	8.22	21.7
	SNGAN+DDLS (Che et al., 2020)	9.09	15.42
	SNGAN+DCD (Song et al., 2020)	9.11	16.24
	BigGAN (Brock et al., 2018)	9.22	14.73
	StyleGAN2 w/o ADA (Karras et al., 2020a)	8.99	9.9

VAEBM is 12x faster than NCSN (Song & Ermon)

QUALITATIVE RESULTS

Other datasets

FID: NVAE 14.7 → VAEBM 5.3

(a) CelebA 64

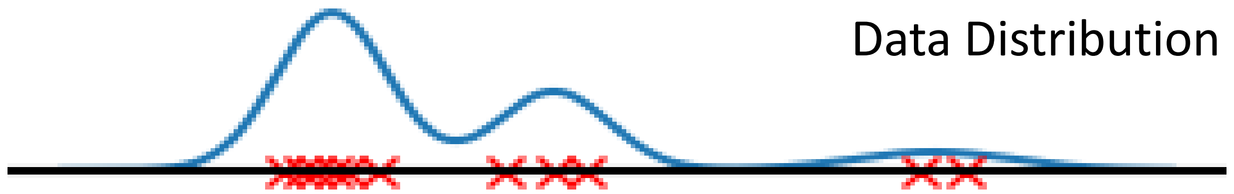
FID: NVAE 41.3 → VAEBM 13.5

(b) LSUN Church 64

FID: NVAE 45.1 → VAEBM 20.4

(c) CelebA HQ 256

OUT OF DISTRIBUTION DETECTION



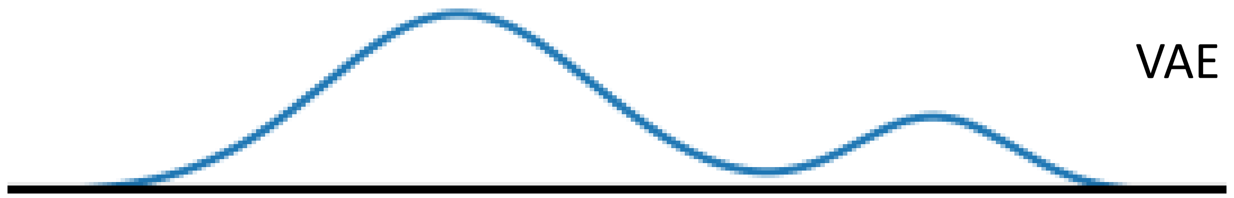
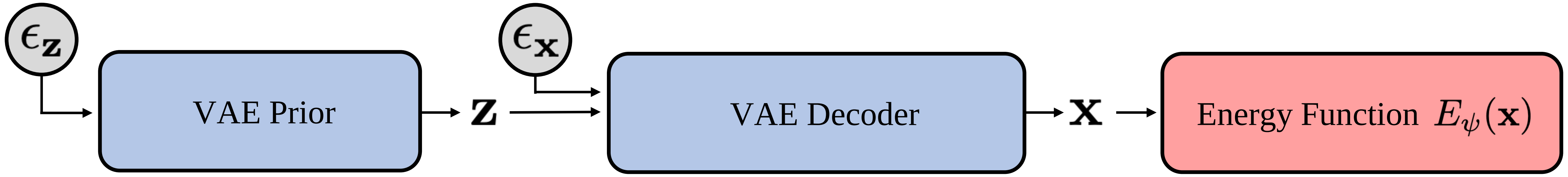


Table 6: Table for AUROC↑ of $\log p(\mathbf{x})$ computed on several OOD datasets. In-distribution dataset is CIFAR-10. Interp. corresponds to linear interpolation between CIFAR-10 images.

		SVHN	Interp.	CIFAR100	CelebA
Unsupervised Training	NVAE (Vahdat & Kautz, 2020)	0.42	0.64	0.56	0.68
	Glow (Kingma & Dhariwal, 2018)	0.05	0.51	0.55	0.57
	IGEBM (Du & Mordatch, 2019)	0.63	**0.7**	0.5	0.7
	Divergence Traingle (Han et al., 2020)	0.68	-	-	0.56
	VAEBM (ours)	**0.83**	**0.7**	**0.62**	**0.77**
Supervised Training	JEM (Grathwohl et al., 2020a)	0.67	0.65	0.67	0.75
	HDGE (Liu & Abbeel, 2020)	0.96	0.82	0.91	0.8

SUMMARY

VAEBM: A symbiotic composition of VAE & EBM



$$p_{\text{VAEBM}}(x) \propto p_{\text{VAE}}(x)\, p_{\text{EBM}}(x)$$

- A two-stage training is proposed
- The experimental results show that the EBM component can improve the generative quality of VAEs by a large margin
- VAE helps with MCMC sampling from the EBM component
- We showed out of distribution detection results and studied mode coverage properties
- Codes will be available at github.com/NVlabs/VAEBM