Friedrich-Alexander-Universität
Technische Fakultät

Advanced Deep Learning

Energy-based Models

Florian Kordon[1], Katharina Breininger[2], Vincent Christlein[1]

1 Pattern Recognition Lab, FAU
2 Artificial Intelligence in Medical Imaging, FAU

June, 7th 2023

Intuition: What is Energy-based Modeling?

Today‘s lecture

What is energy-based modeling?

Intuition

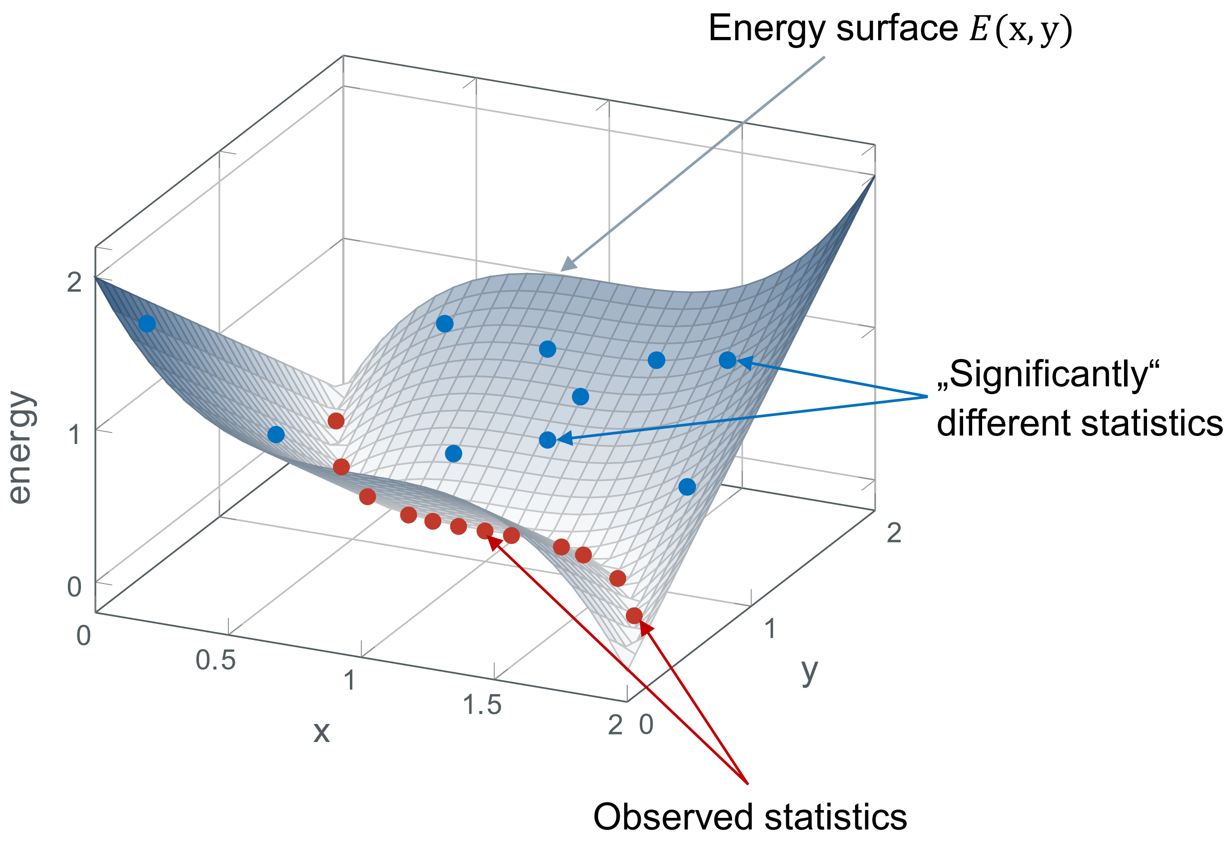
Assume some observations (red data points) that share a common set of characteristics.

These observations lie on a high-dimensional manifold $\mathcal{M}$.

→ Find an embedding function ϕ that maps $\mathcal{M}$ to a structure-preserving (potentially lower-dim.) representation $\mathcal{N}$.

$$\phi: \mathcal{M} \mapsto \mathcal{N}$$

An energy-based model (EBM) performs such mapping by minimizing the value of the energy function E for the observed statistics (red data points), while increasing it for all other statistics (blue data points).



[1] Chris G. Willcocks. Deep Learning Lecture 7: Energy-based models. https://cwkx.github.io/data/teaching/dl-and-rl/dl-lecture7.pdf

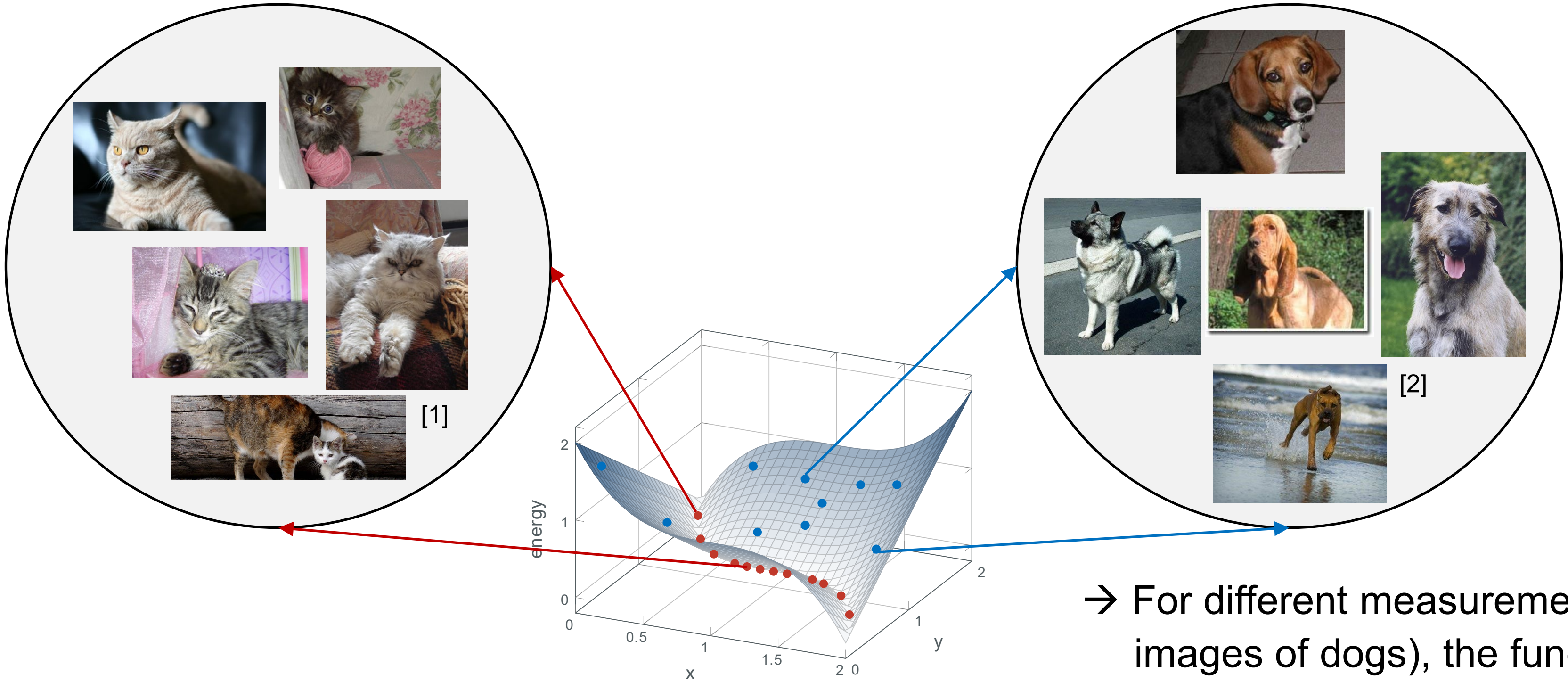
Energy-based models

Illustrative example

We define an energy function $E(\mathbf{x})$ that assigns a small scalar value to images of different breeds of cats.

Observed cat statistics

$$E(\mathbf{x}) \approx 0$$



Unknown dog statistics

$$E(\mathbf{x}') > 0$$

→ For different measurements (e.g., images of dogs), the function output should be significantly larger.

[1] https://www.kaggle.com/datasets/crawford/cat-dataset
[2] https://www.kaggle.com/datasets/jessicali9530/stanford-dogs-dataset

Background and Fundamentals

Energy-based models

Energy function E as the backbone of energy-based models

Fundamentals

The energy function $E(\mathrm{x},\mathrm{y})$ measures the compatibility between an input data point $\mathrm{x} \in \mathbb{R}^{D_x}$ and a target variable $\mathrm{y} \in \mathbb{R}^{D_y}$ by assigning a scalar value (=energy).

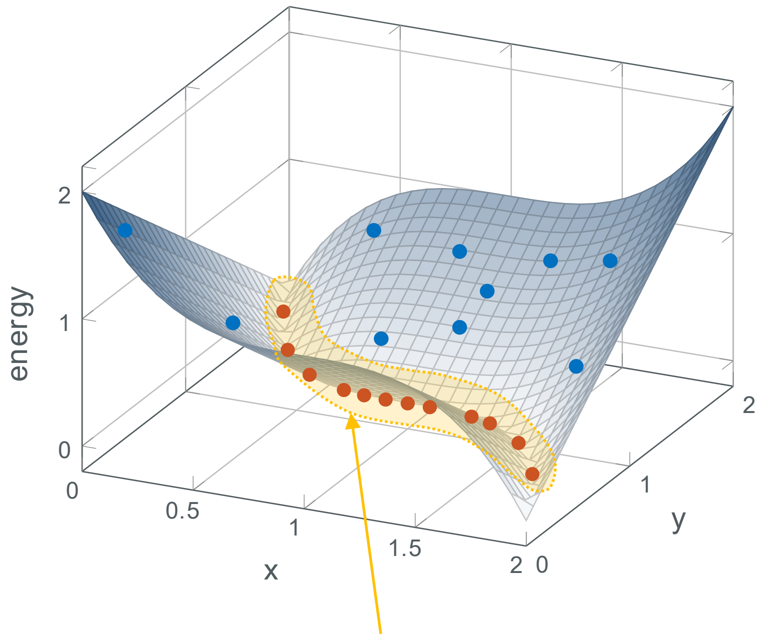
$$E(\mathrm{x},\mathrm{y}): \mathbb{R}^{D_x} \times \mathbb{R}^{D_y} \mapsto \mathbb{R}$$

→ Takes small values for compatible x and y, high values for less compatible x and y.

→ y can represent, e.g., a binary or categorial assignment, a target image, or an object‘s coordinates.

During inference, we search for configurations of x for which $E(\mathrm{x},\mathrm{y})$ is small.

$$\hat{\mathrm{y}} = \mathrm{argmin}_{\mathrm{y}}\, E(\mathrm{x},\mathrm{y})$$



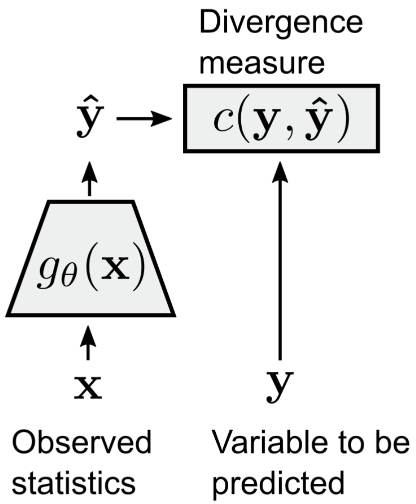
‚Good‘ solutions
(there most likely are multiple ones)

[1] Chris G. Willcocks. Deep Learning Lecture 7: Energy-based models. https://cwkx.github.io/data/teaching/dl-and-rl/dl-lecture7.pdf

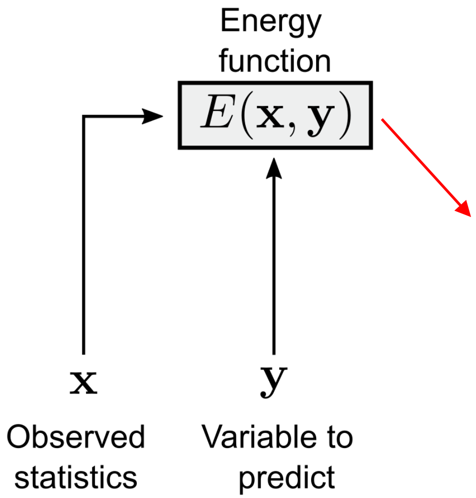
Energy-based models

EBMs as implicit functions

→ A feed-forward model represents an explicit function that predicts $\hat{y}$ based off x



→ An energy-based model (EBM) is an implicit function that models the relation and compatability between x and y



Parameterization is very flexible, this can be almost any kind of function.

→ e.g., $E_\theta(x, y)$ with parameters θ being the weights of a neural network

[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

Energy-based models

Inference and training

A) Inference procedure

During inference, we search for configurations of x for which the energy function $E_\theta(\mathrm{x},\mathrm{y})$ is small.

$$\hat{\mathrm{y}} = \mathrm{argmin}_{\mathrm{y}}\, E_\theta(\mathrm{x},\mathrm{y})$$

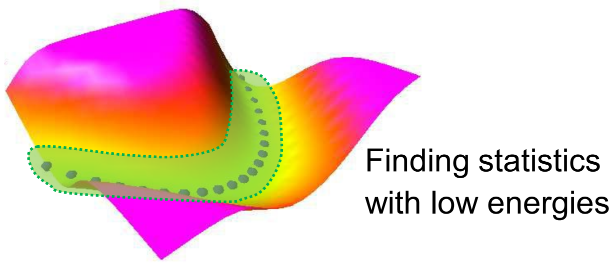
B) Training procedure

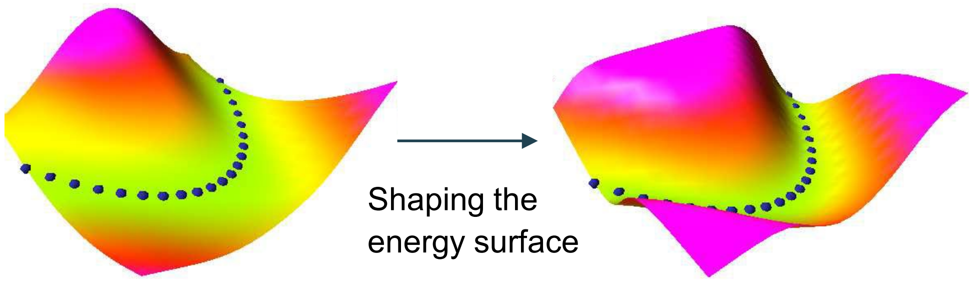
During training, we search for an energy function $E_\theta(\mathrm{x},\mathrm{y})$ from the space of functions $\mathcal{E}$ that yields the optimal y for any x.

This typically involves searching for the best set of parameters θ^* for a particular function E using a loss functional $\mathcal{L}$

$$\theta^* = \mathrm{argmin}_\theta\, \mathcal{L}\,(\theta, S)$$

, where S is the set of training statistics $S = \{\mathrm{x}_i, \mathrm{y}_i\}$.

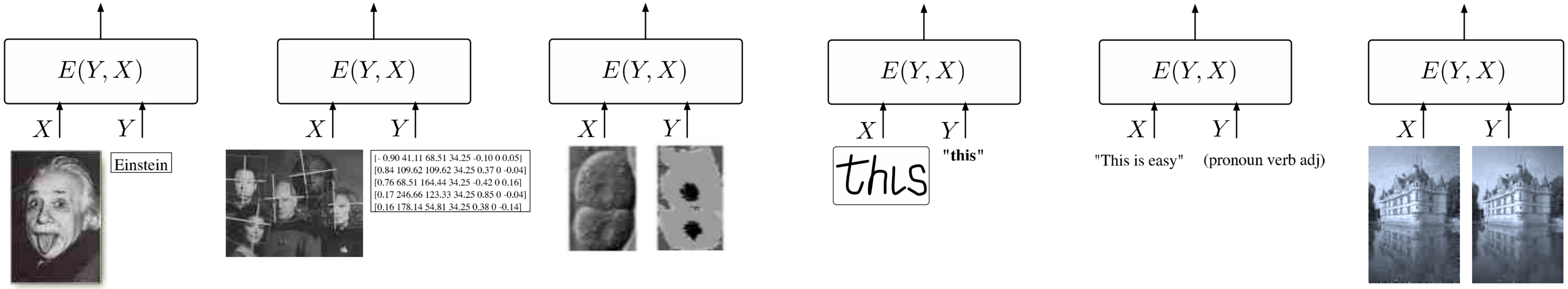




[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

Energy-based models

A) Inference strategies



Face recognition | Pose estimation | Segmentation | Handwritten text recognition | Sequence labeling | Image2Image Translation

Inference can be quite difficult for high-cardinality solution spaces $\mathcal{Y}$ (exhaustive search over all states is not feasible).

Suitable inference strategies include:

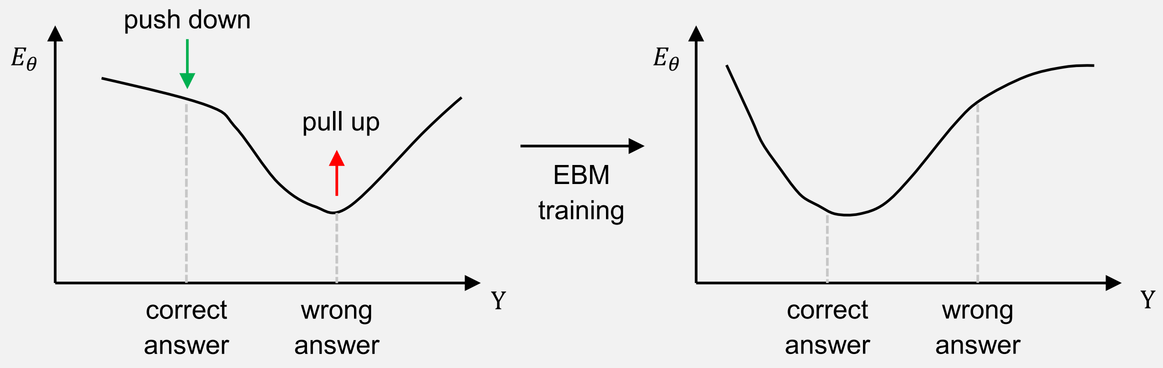
- gradient-decent (if y is continuous and the energy function is smooth and differentiable)
- belief propagation
- min-sum (factor graphs)
- dynamic programming (Viterbi, A$*$).
- variational approaches with surrogate energies
- …

[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

Energy-based models

B) Training strategies

Intuition on the training procedure (per-sample basis)



→ Push down the energy of the correct answer

→ Pull up the energies of the incorrect answers

Standard loss functions (amongst many)

Energy loss

$$\mathcal{L}_{\text{energy}}(E_\theta(\mathrm{x}, \mathcal{Y}), \mathrm{y}) = E_\theta(\mathrm{x}, \mathrm{y})$$

Perceptron loss

$$\mathcal{L}_{\text{perceptron}}(E_\theta(\mathrm{x}, \mathcal{Y}), \mathrm{y}) = E_\theta(\mathrm{x}, \mathrm{y}) \min_{\mathrm{y}' \in \mathcal{Y}} E(\mathrm{x}, \mathrm{y}')$$

Hinge loss

$$\mathcal{L}_{\text{hinge}}(E_\theta(\mathrm{x}, \mathcal{Y}), \mathrm{y}) = \max(0, \underset{\substack{\uparrow \\ \text{positive margin}}}{m} + E_\theta(\mathrm{x}, \mathrm{y}) - \underbrace{E_\theta(\mathrm{x}, \bar{\mathrm{y}})}_{\text{most offending incorrect answer}})$$

→ See [1] for a broad overview of standard loss functions and their pros and cons

[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

[2] Stefano Ermon, Yang Song. Energy-Based Models. Lecture Stanford University

From energies to probability

Limitations

The energies are uncalibrated (=un-normalized log-probability)!

What does that mean?

- Not a particular problem for low-cardinality decision making scenarios (just take the solution with the lowest energy)
- But: Two EBMs trained independently most likely have different energies scales. This renders model combination or comparison almost impossible.
- But: Without normalizing the energies, the likelihood of the observed statistics can change with different energy scales

→ We need to find a consistent way to embed EBMs in a framework of common energy units.

Straight-forward solution?*

We embed the energies in a probability distribution $p(\mathrm{x})$ with properties:

1. non-negative variables: $p(\mathrm{x}) \geq 0$ ⟵ easy to achieve
2. integrates to 1: $\int p(\mathrm{x})\, d\mathrm{x} = 1$ ⟵ essential, but hard for non-trivial problems

*We continue with a simpler notation of the energy function $E_\theta(\mathrm{x})$ that models distribution membership. The conditional $p(\mathrm{y}|\mathrm{x})$ can thus be simplified to $p(\mathrm{x})$

[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006
[2] Stefano Ermon, Yang Song. Energy-Based Models. Lecture Stanford University

From energies to probability

Boltzmann-Gibbs distribution

$$p(\mathrm{x}) = \frac{1}{Z} \exp\left(-\frac{E(\mathrm{x})}{T}\right)$$

$$Z = \int \exp\left(-\frac{E(\mathrm{x})}{T}\right) dx$$

- x: system state
- $E(x)$: system energy at state x
- T: system temperature
- Z: normalizing constant/partition function

Parameterized EBM

$$q_\theta(\mathrm{x}) = \frac{1}{Z_\theta} \exp(-E_\theta(\mathrm{x}))$$

$$\boxed{Z_\theta = \int \exp(-E_\theta(\mathrm{x}))\, dx}$$

- x: image, text, etc.
- $E_\theta(x)/T$: energy function (parameterized by a neural network)

What can we do to compute the partition function?

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.

From energies to probability

Approach 1: Is the task based on pair-wise comparison (e.g. denoising)?

For two data points x, x', calculating $q_\theta(\mathrm{x})$ and $q_\theta(\mathrm{x}')$ requires us to know Z_θ. If we can use their ratio, however, we can avoid computating Z_θ entirely: $\frac{q_\theta(\mathrm{x})}{q_\theta(\mathrm{x}')} = \exp(-E_\theta(\mathrm{x}) + E_\theta(\mathrm{x}'))$.

Approach 2: Choose energy function $E_\theta(\mathrm{x})$ such that we can compute $Z_\theta(\mathrm{x})$ analytically.

This approach is viable for trivial choices of the energy function. However, we can also build more complex methods by using products of individually normalized functions (autoregressive, product of experts) or mixtures of normalized objects (latent variables).

Approach 3: Approximate the partition function $Z_\theta(\mathrm{x})$ using a Monte Carlo (MC) estimate.

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Stefano Ermon, Yang Song. Energy-Based Models. Lecture Stanford University

Maximum Likelihood Estimation for EBMs

MCMC for model synthesis

Prerequisites

n observed data points: $\{\mathrm{x}_1, \dots, \mathrm{x}_n\} \sim p_{\text{train}}(\mathrm{x})$

Model: $q_\theta(\mathrm{x}) = \frac{1}{Z_\theta} \exp(-E_\theta(\mathrm{x}))$

MLE objective: $\mathcal{L}(\theta; p) = \frac{1}{n} \sum_{i=1}^{n} \log q_\theta(\mathrm{x}_\mathrm{i}) \doteq \mathbb{E}_{p(\mathrm{x})}[\log q_\theta(\mathrm{x})]$

Derivative of negative log-likelihood (NLL)

$$
\begin{aligned}
\frac{\partial \mathcal{L}(\theta; p)}{\partial \theta} &= \mathbb{E}_{p(\mathrm{x})}\left[\frac{\partial E_\theta(\mathrm{x})}{\partial \theta}\right] - \frac{\partial \log Z_\theta}{\partial \theta} \\
&\overset{*}{=} \mathbb{E}_{p(\mathrm{x})}\left[\frac{\partial E_\theta(\mathrm{x})}{\partial \theta}\right] - \boxed{\mathbb{E}_{q_\theta(\mathrm{x})}\left[\frac{\partial E_\theta(\mathrm{x})}{\partial \theta}\right]} \\
&\approx \frac{1}{n} \sum_{i=1}^{n} \frac{\partial E_\theta(\mathrm{x}_i)}{\partial \theta} - \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} \frac{\partial E_\theta(\boxed{\tilde{\mathrm{x}}_i})}{\partial \theta}
\end{aligned}
$$

This is analytically intractable:

Consider grayscale image $\mathrm{x} \in \mathbb{R}^{128 \times 128}$ (uint)

Solution space is $256^{16{,}384}$!

Approximate by Markov Chain Monte Carlo (MCMC)
$\{\tilde{\mathrm{x}}_1, \dots, \tilde{\mathrm{x}}_n\} \sim q_\theta(\mathrm{x})$

* [1] Oliver Woodford. Notes on Contrastive Divergence. Department of Engineering Science, University of Oxford, Tech. Rep 4 (2006).

Maximum Likelihood Estimation for EBMs

Contrastive Optimization

Recall the gradient of the NLL objective:

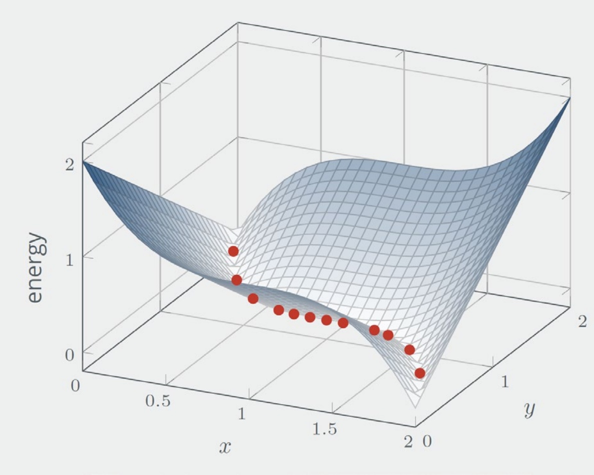
Contrastive Approximation

$$\frac{\partial \mathcal{L}(\theta; p)}{\partial \theta} \approx \underbrace{\frac{1}{n}\sum_{i=1}^{n}\frac{\partial E_\theta(\mathrm{x}_i)}{\partial \theta}}_{\text{Minimize energies for observed statistics}} - \underbrace{\frac{1}{\tilde{n}}\sum_{i=1}^{\tilde{n}}\frac{\partial E_\theta(\tilde{\mathrm{x}}_i)}{\partial \theta}}_{\text{Maximize energies for synthesized statistics from our current model}}$$

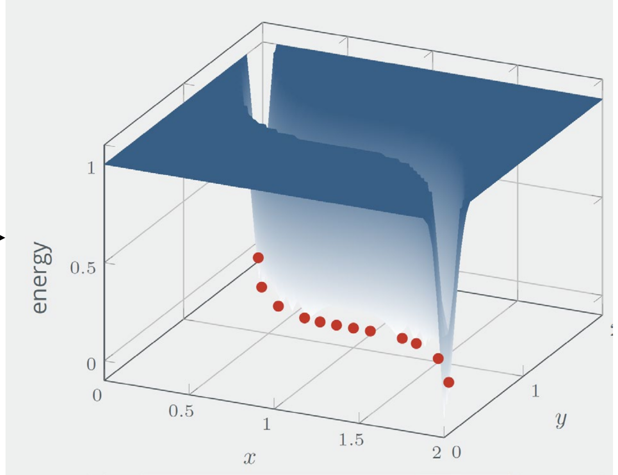
Optimum at equilibrium $q_\theta(\mathrm{x}) \simeq p(\mathrm{x})$

→ "The data samples from the training distribution should be more likely than a sample from our model."

Caution: The objective wants to strongly maximize the energy off the manifold, leading to non-smooth energies. → Consider loss regularization!



→



[1] Chris G. Willcocks. Deep Learning Lecture 7: Energy-based models. https://cwkx.github.io/data/teaching/dl-and-rl/dl-lecture7.pdf
[2] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

Gradient-based MCMC w. Langevin Dynamics

Synthesizing samples from $q_\theta(\mathrm{x})$ via vanilla MCMC works in theory but shows extremely long mixing times.

→ For distributions $q_\theta(\mathrm{x})$ that are continuous, we can exploit a score function $\frac{\partial \log q_\theta(\mathrm{x})}{\partial \mathrm{x}}$ for gradient-based synthesis.

→ Widely used approaches are, e.g., Langevin Dynamics (SGLD) or Hamiltonian Monte Carlo (HMC).

MCMC with Stochastic Gradient Langevin Dynamics

Stochastic gradient ascent*

Brownian motion

$$\mathrm{x}^0 \sim \pi(\mathrm{x})$$

$$\begin{aligned} \mathrm{x}^{k+1} &= \mathrm{x}^k \boxed{+ \frac{\eta}{2} \frac{\partial \log q_\theta(\mathrm{x}^k)}{\partial \mathrm{x}^k}} + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \eta) \\ &= \mathrm{x}^k - \frac{\eta}{2} \frac{\partial E_\theta(\mathrm{x}^k)}{\partial \mathrm{x}^k} - \underbrace{\frac{\partial \log Z_\theta}{\partial \mathrm{x}^k}}_{=0} + \epsilon \\ &= \mathrm{x}^k - \frac{\eta}{2} \frac{\partial E_\theta(\mathrm{x}^k)}{\partial \mathrm{x}^k} + \epsilon \end{aligned}$$

First order Euler discretation of a stochastic differential equation [2]

→ Gradient term (approximated on mini-batch) enforces dynamics to focus on regions of high probability

→ Brownian motion imposes noisy trajectory so that the dynamics explore the complete parameter space

*Recall that $f_\theta(\mathrm{x}) = -E_\theta(\mathrm{x})$ (per convention) denotes the negative energy function.

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Max Welling, Yee Whye The. Bayesian Learning via Stochastic Gradient Langevin Dynamics. ICML 2011. https://www.stats.ox.ac.uk/~teh/research/compstats/sgld.pdf

Gradient-based MCMC w. Langevin Dynamics

There exist different approaches for selecting the starting point of SGLD-based synthesis:

1. Contrastive Divergence: $\mathrm{x}^0 \sim p_{\text{train}}(\mathrm{x})$ → Run finite MCMC from observed samples from the training dataset.
2. Persistent Chain: $\mathrm{x}^0 \sim q_\theta^{\text{current epoch}-1}(\mathrm{x})$ → Run finite MCMC from synthesized examples from previous epoch.
3. Non-persistent Short-run MCMC [2]: $\mathrm{x}^0 \sim \pi(\mathrm{x})$ → Run finite MCMC from Gaussian noise (see previous slide).

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Erik Nijkamp, Mitch Hill, Song-Chun Zhu, Ying NianWu. On learning non-convergent non-persistent short-run MCMC toward energy-based model. NeurIPS, 2019.

Gradient-based MCMC w. Langevin Dynamics

Initial sample (white Gaussian noise)

High-quality sample with low energy/high likelihood under $q_\theta(\mathrm{x})$

x^0 → x^{50} (50 Langevin steps)

[1] Yang Zhao, Jianwen Xie, Ping Li. Learning Energy-Based Generative Models via Coarse-to-Fine Expanding and Sampling. ICLR, 2021.

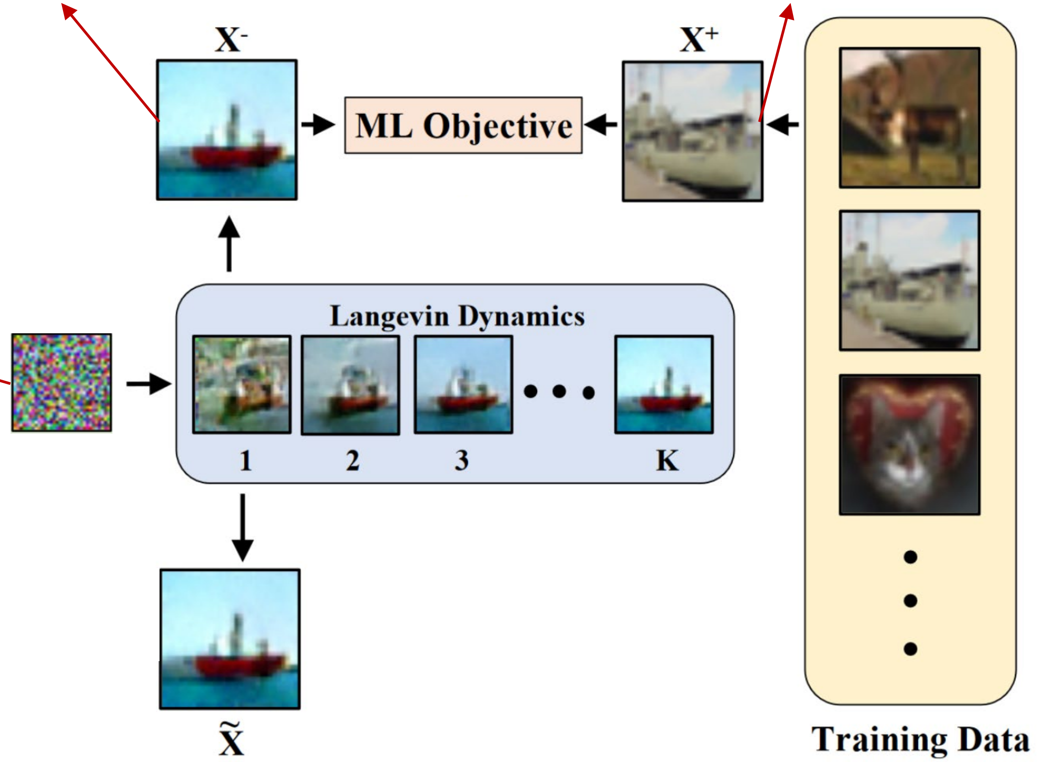
Training and Sampling Algorithm

Overview

Synthetic sample from $q_\theta(\mathrm{x})$ after K steps of MCMC
→ Maximize energy

Real sample from training distribution $p_{\mathrm{train}}(\mathrm{x})$
→ Minimize energy

Initial sample (white Gaussian noise)

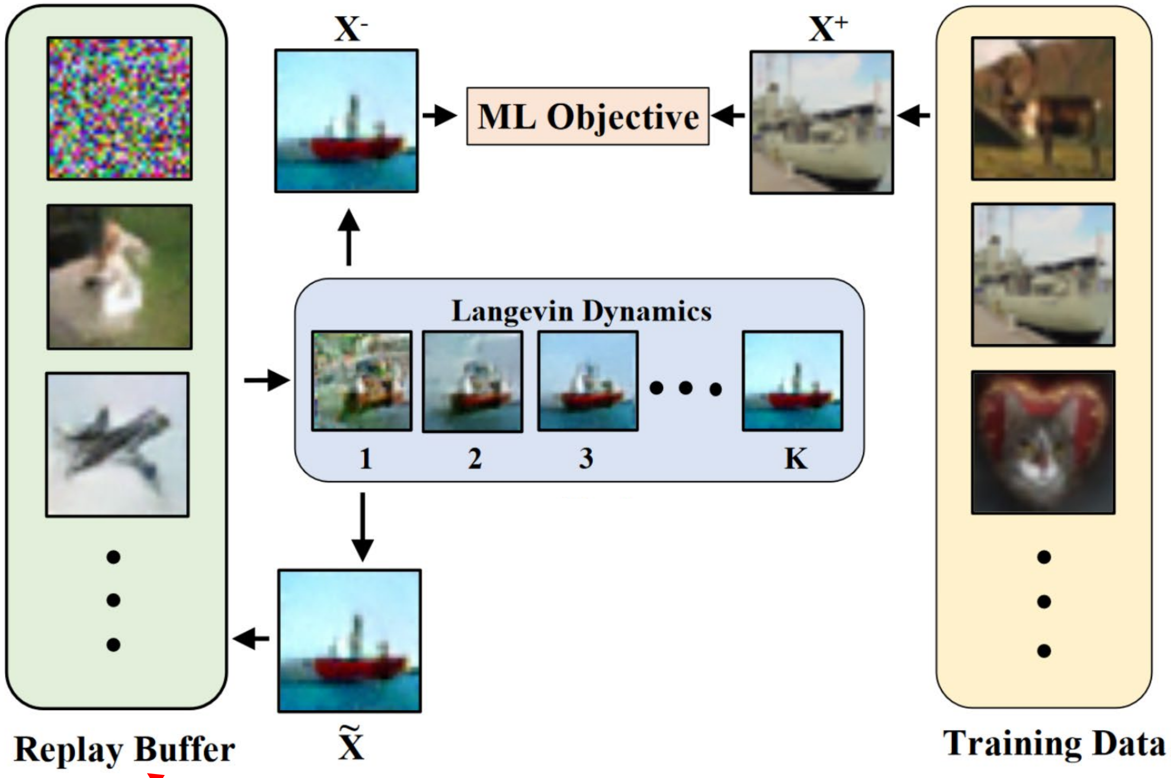


We can use the trained EBM to synthesize new (and quite realistic) samples. (Interpretation as a latent generative model).

[1] Yilun Du, Igor Mordatch. Implicit Generation and Generalization in Energy-Based Models. NeurIPS, 2019.

Training and Sampling Algorithm

General approach



Leads to mixture of a) non-persistent short-run MCMC and b) persistent MCMC

Standard training and sampling algorithm for EBMs

<u>**Input**</u> data distribution $p(x)$, MCMC mixing steps K

Initialize synthesis buffer $\mathcal{B} \leftarrow \emptyset$

while not converged **do:**

  sample from dataset: $x_i^+ \sim p_{\text{train}}$

  initial synthesis: $x_i^0 \sim \mathcal{B}$ with 95% probability, $x_i^0 \sim \mathcal{U}$ or $x_i^0 \sim \mathcal{N}$ otherwise

  for sample step $k = 1$ to K **do**:

$$\tilde{x}^k = \tilde{x}^{k-1} - \frac{\eta}{2}\frac{\partial E_\theta(\tilde{x}^{k-1})}{\partial \tilde{x}^{k-1}} + \epsilon, \;\; \epsilon \sim \mathcal{N}(0, \eta)$$

  end for

  $x_i^- \leftarrow \Omega(\tilde{\mathrm{x}}_i^k)$

  Contrastive Divergence: $\mathcal{L}_{\text{CD}} = \frac{1}{N}\sum_i E_\theta(x_i^+) - E_\theta(x_i^-)$

  L2 regularization: $\mathcal{L}_{\text{L2}} = \frac{1}{N}\sum_i E_\theta(x_i^+)^{\,2} + E_\theta(x_i^-)^2$

  Optimization with SGD/Adam/…: $\frac{\partial}{\partial\theta}(\mathcal{L}_{\text{CD}} + \lambda\mathcal{L}_{\text{L2}})$

  add to buffer: $\mathcal{B} \leftarrow \mathcal{B} \cup x_i^-$

end while

→ Alternating between sampling $q_\theta(\mathrm{x})$ and updating parameters θ of $q_\theta(\mathrm{x})$

[1] Yilun Du, Igor Mordatch. Implicit Generation and Generalization in Energy-Based Models. NeurIPS, 2019.
[2] Phillip Lippe. Tutorial 8: Deep Energy-Based Generative Models. https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/tutorial8/Deep_Energy_Models.html#Training-algorithm

Adversarial Interpretation of Training and Sampling

Recall the contrastive algorithm where we synthesize data points using MCMC.

$$\frac{\partial \mathcal{L}(\theta;p)}{\partial \theta} \approx \frac{1}{n}\sum_{i=1}^{n} \frac{\partial E_\theta(\mathrm{x}_i)}{\partial \theta} - \frac{1}{\tilde{n}}\sum_{i=1}^{n} \frac{\partial E_\theta(\tilde{\mathrm{x}}_i)}{\partial \theta}$$

$$= \frac{\partial}{\partial \theta}\left[\frac{1}{n}\sum_{i=1}^{n} E_\theta(\mathrm{x}_i) - \frac{1}{\tilde{n}}\sum_{i=1}^{n} E_\theta(\tilde{\mathrm{x}}_i)\right]$$

→ Define value function $V(\{\tilde{\mathrm{x}}_i\},\theta) = \frac{1}{n}\sum_{i=1}^{n} E_\theta(\mathrm{x}_i) - \frac{1}{\tilde{n}}\sum_{i=1}^{n} E_\theta(\tilde{\mathrm{x}}_i)$

→ The alternating i) Learning and ii) Sampling procedures play an adversarial minmax game:

$$\min_{\{\tilde{\mathrm{x}}_i\}} \max_{\theta} V(\{\tilde{\mathrm{x}}_i\},\theta)$$

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.

Non-persistent Short-Run MCMC

Problem If $q_\theta(\mathrm{x})$ is multi-modal, different MC chains might get trapped in distinct local modes → no mixing.

Approach Nijkamp et al. [2] propose to run a non-convergent, non-mixing, and non-persistent chain that starts from always the same initial noise distribution q_0 (uniform, Gaussian) and run it for a small fixed number of steps K towards q_θ. → Short-run MCMC

Short-run MCMC

M_θ: K-step MCMC transition kernel
$z \sim q_0$: starting point (i.e., latent variables)

$$\gamma_\theta(\mathrm{x}) = (M_\theta q_0)(\mathrm{z}) = \int q_0(\mathrm{z}) M_\theta(\mathrm{x}|\mathrm{z}) dz$$

Marginal distribution of sample x after K-step MCMC from q_0

$$\mathrm{x} = M_\theta(\mathrm{z}, \epsilon)$$

If training converges, EBMs tends to have low entropy and Langevin dynamics behaves like GD with disabled noise term ϵ → $\mathrm{x} = M_\theta(\mathrm{z})$

→ No longer maximum-likelihood estimation, but moment-matching estimation (MME) with the following relation to solve:

$$\mathbb{E}_{p(\mathrm{x})}\left[\frac{\partial E_\theta(\mathrm{x})}{\partial \theta}\right] = \mathbb{E}_{\gamma_\theta(\mathrm{x})}\left[\frac{\partial E_\theta(\mathrm{x})}{\partial \theta}\right]$$

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Erik Nijkamp, Mitch Hill, Song-Chun Zhu, Ying NianWu. On learning non-convergent non-persistent short-run MCMC toward energy-based model. NeurIPS, 2019.

Non-persistent Short-Run MCMC

Image synthesis using Short-run MCMC ($K = 100$, $q_0 \sim \mathcal{U}[-1,1]$) on CelebA (64x64)

Image synthesis using Short-run MCMC ($K = 100$, $q_0 \sim \mathcal{U}[-1,1]$) on CelebA (128x128)

[1] Erik Nijkamp, Mitch Hill, Song-Chun Zhu, Ying NianWu. On learning non-convergent non-persistent short-run MCMC toward energy-based model. NeurIPS, 2019.

Non-persistent Short-Run MCMC

$M_\theta(z_1)$

$M_\theta(z_2)$

Interpolation using Short-run MCMC as generative latent model. Transitions depits $M_\theta(z_p)$ with interpolated noise $z_p = \wp z_1 + \sqrt{1-\wp^2 z_2}$ where $\wp \in [0,1]$ on CelebA (64x64).

$M_\theta(z_{t=0})$

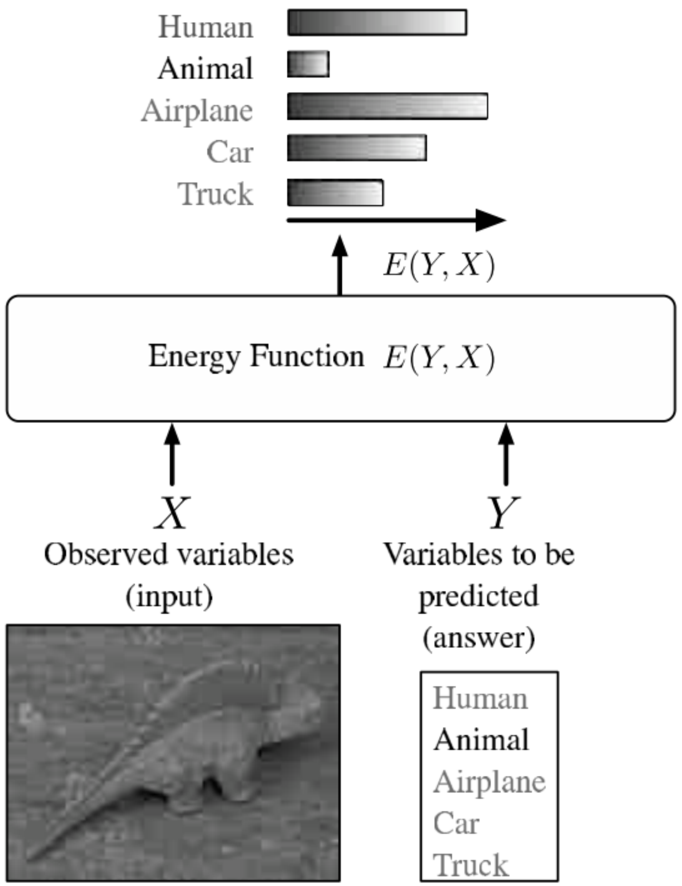
$M_\theta(z_{t=200})$

Reconstruction using Short-run MCMC as generative latent model. Transitions depits $M_\theta(z_t)$ over time t from random noise at $t = 0$ to reconstruction of observed example at $t = 200$ on CelebA (64x64).

[1] Erik Nijkamp, Mitch Hill, Song-Chun Zhu, Ying NianWu. On learning non-convergent non-persistent short-run MCMC toward energy-based model. NeurIPS, 2019.

Joint Energy-based Models (JEM)

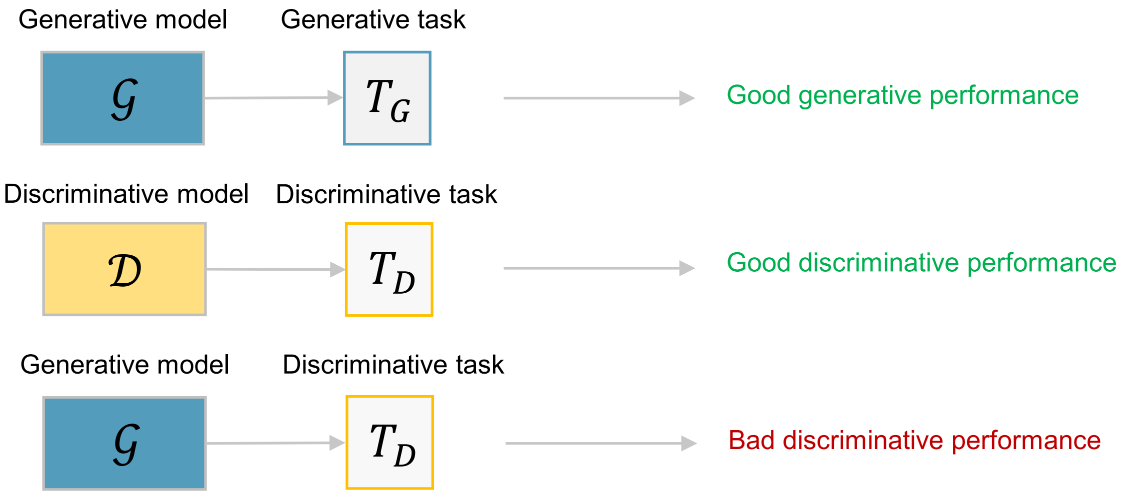
Joint Energy-based Models (JEM)



We can use the joint probability density $p(\mathrm{x}, \mathrm{y})$ for classification and recognition tasks: $\hat{\mathrm{y}} = \mathrm{argmin}_{\mathrm{y}}\, E(\mathrm{x}, \mathrm{y})$

General observation

Discriminative capabilities of generative models are inferior to discriminative models that specialize on classification tasks!



[1] Yann LeCun et al. A Tutorial on Energy-Based Learning. Predicting structured data. 2006

Joint Energy-based Models (JEM)

Gratwohl et al. made a fascinating observation

(1) Categorical distribution obtained by classifier: $q_\theta(y|\mathbf{x}) = \left. \frac{\exp(\overbrace{f_\theta(\mathbf{x})[y]}^{\text{logits for target class label}})}{\sum_{y'} \exp(f_\theta(\mathbf{x})[y'])} \right]$ Pseudo-probabilities via softmax over all y

(2) We can re-interpret the logits as unnormalized densities of the joint distribution $q_\theta(\mathrm{x}, y)$!

$$q_\theta(\mathbf{x}, y) = \frac{\exp(\overbrace{f_\theta(\mathbf{x})[y]}^{E_\theta(\mathrm{x},\mathrm{y}) = -f_\theta(\mathrm{x})[\mathrm{y}]})}{\underbrace{Z(\theta)}_{\text{Partition function}}} \xrightarrow{\text{Marginalization over y}} q_\theta(\mathbf{x}) = \sum_y q_\theta(\mathbf{x}, y) = \underbrace{\frac{\sum_y \exp(f_\theta(\mathbf{x})[y])}{Z(\theta)}}_{\text{Density model for } \mathrm{x}}$$

logits for target class label

(3) Energy function at point x: $E_\theta(\mathbf{x}) = -\mathrm{LogSumExp}_y(f_\theta(\mathbf{x})[y]) = -\log \sum_y \exp(f_\theta(\mathbf{x})[y])$

[1] Gratwohl et al. Your classifier is secretely an Energy-based model and you should treat it like one, ICLR 2020

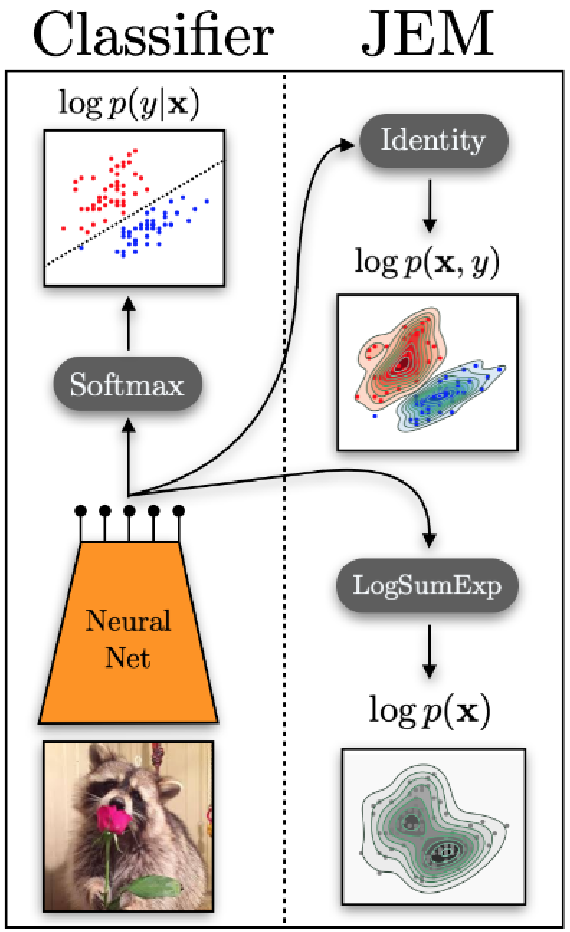
Joint Energy-based Models (JEM)

(4) Efficient training with factorization

→ Recall that the posterior is given by $p(\mathrm{y}|\mathbf{x}) = \frac{p(\mathbf{x},\mathrm{y})}{p(\mathbf{x})}$

→ Factor the likelihood to facilitate training a hybrid model (generative and discriminative aspects)

$$\log q_\theta(\mathbf{x}, y) = \underbrace{\log q_\theta(\mathbf{x})}_{\text{optimize density model with NLL + SGLD}} + \underbrace{\log q_\theta(y|\mathbf{x})}_{\text{optimize with standard cross-entropy}}$$



[1] Gratwohl et al. Your classifier is secretely an Energy-based model and you should treat it like one, ICLR 2020

Application Examples

Application examples

Energy-based inpainting

Can we recover missing information from incomplete training data in an unsupervised fashion?



Proposed solution: Learning + synthesis of new examples + recovery of incomplete training samples

→ Combination of two Langevin dynamics:

1. Start from white noise and synthesize new example a_i
2. Start from incomplete data and synthesize recovered data b_i

→ Update rule: $\theta_{t+1} = \theta_t - \eta_t \left[\frac{1}{n}\sum_{i=1}^{n} \frac{\partial E_\theta(a_i)}{\partial \theta} - \frac{1}{n}\sum_{i=1}^{n} \frac{\partial E_\theta(b_i)}{\partial \theta}\right]$

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

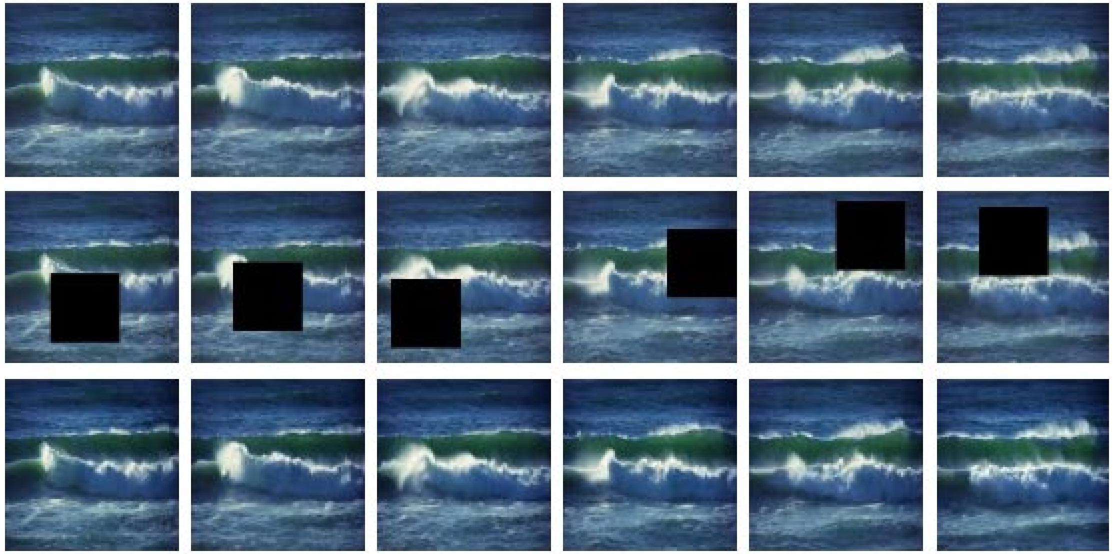
Energy-based inpainting

50% salt and pepper masking

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

Energy-based inpainting



ocean

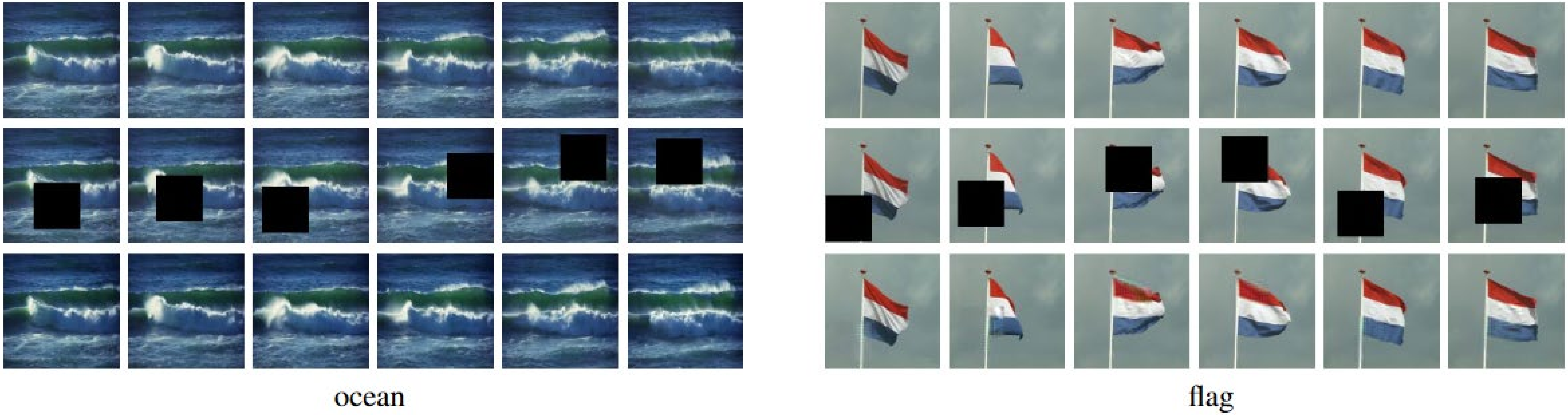
flag

50% salt and pepper masking

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

Energy-based inpainting

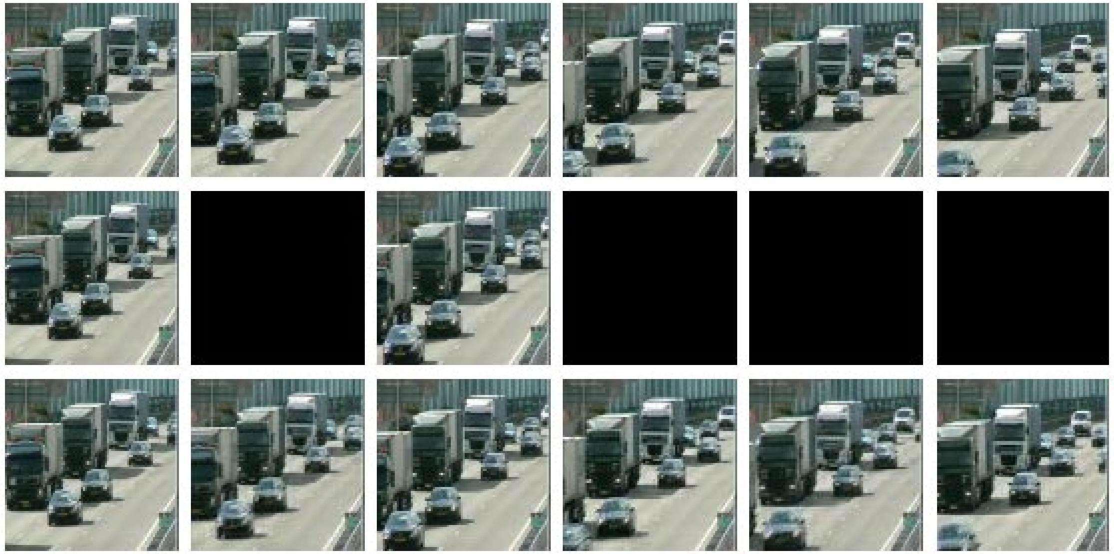


single region masks

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

Energy-based inpainting



traffic

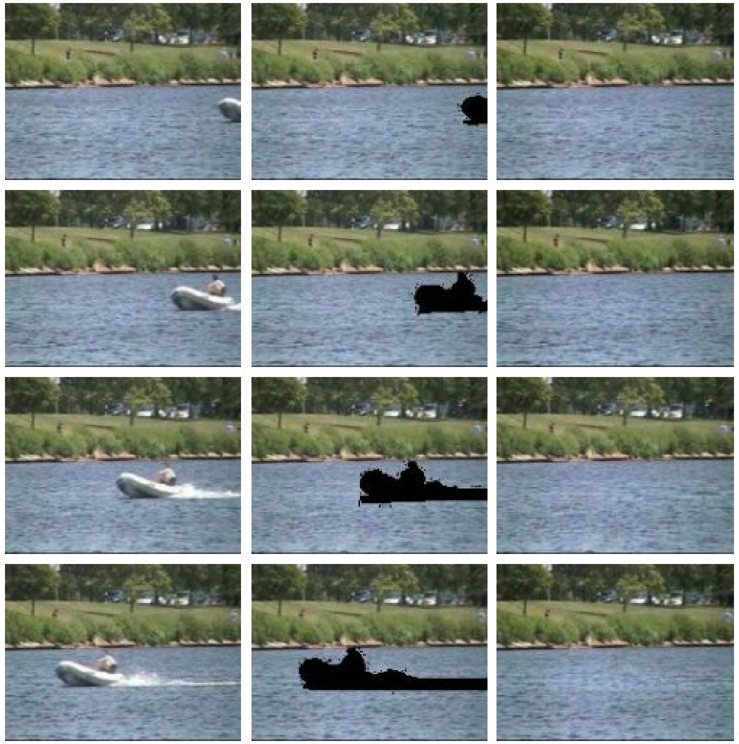
playing

50% missing frames

[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

Energy-based inpainting



(a) removing a moving boat in the lake

(b) removing a walking person in front of fountain

Background inpainting

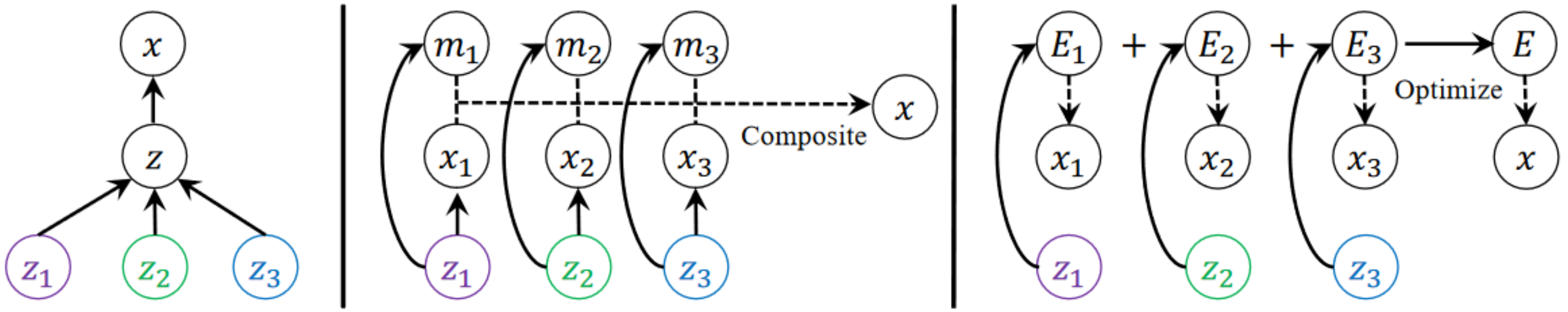
[1] Jianwen Xie, Ying Nian Wu. Theory and Applications of Energy-Based Generative Models. ICCV, 2021.
[2] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Synthesizing Dynamic Pattern by Spatial-Temporal Generative ConvNet. CVPR 2017.
[3] Jianwen Xie, Song-Chun Zhu, Ying Nian Wu. Learning Energy-based Spatial-Temporal Generative ConvNetfor Dynamic Patterns. PAMI 2019.

Application examples

Unsupervised Learning of Compositional Energy Concepts

Can we identify and discover visual concepts (local or global factor of variation) using EBMs?

Proposed solution: COMET[1] infers multiple energy functions individually with separate minimal energy states, each capturing a distinct factor of variation.



(a) β-VAE (b) MONET (c) COMET [1]

Global decoder, mapping factors to images

Composit disjoint segmentation masks

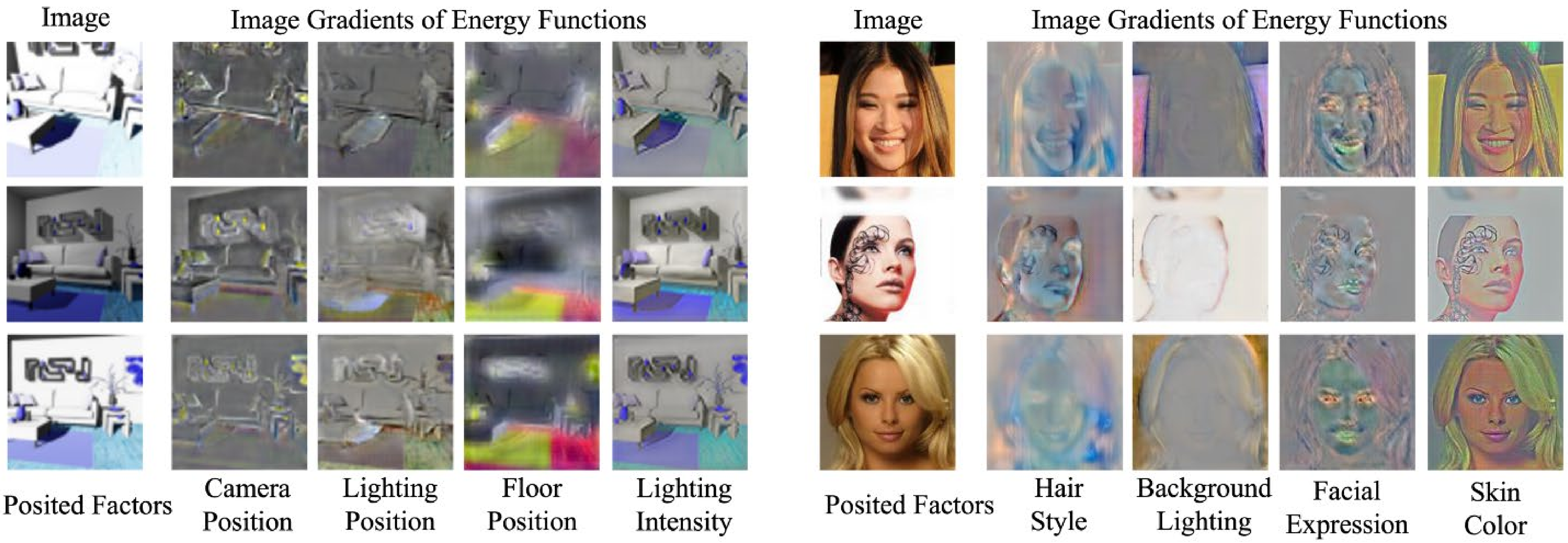
Infer individual energy functions per component, optimize the sum over all.

→ We obtain a set of composable energy functions!

[1] Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, Igor Mordatch. Unsupervised Learning of Compositional Energy Concepts. NeurIPS, 2021.

Application examples

Unsupervised Learning of Compositional Energy Concepts

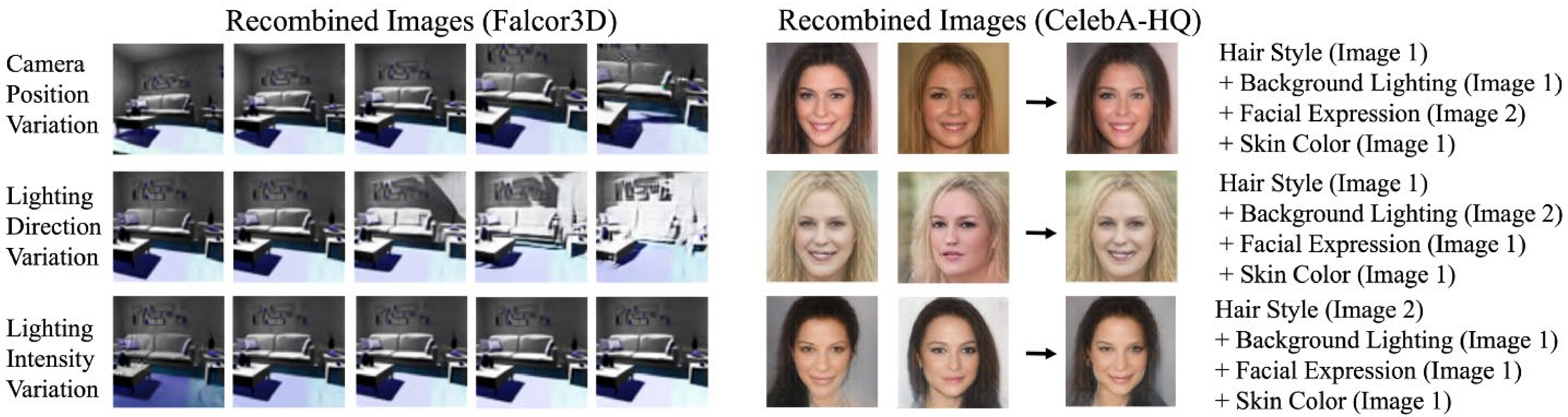


Gradients of the individual energy functions (k=4) wr.t. an image. The gradient images indicate those aspects of an image the respective energy function attends to. (Factor labels are assigned by visual inspection.)

[1] Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, Igor Mordatch. Unsupervised Learning of Compositional Energy Concepts. NeurIPS, 2021.

Application examples

Unsupervised Learning of Compositional Energy Concepts

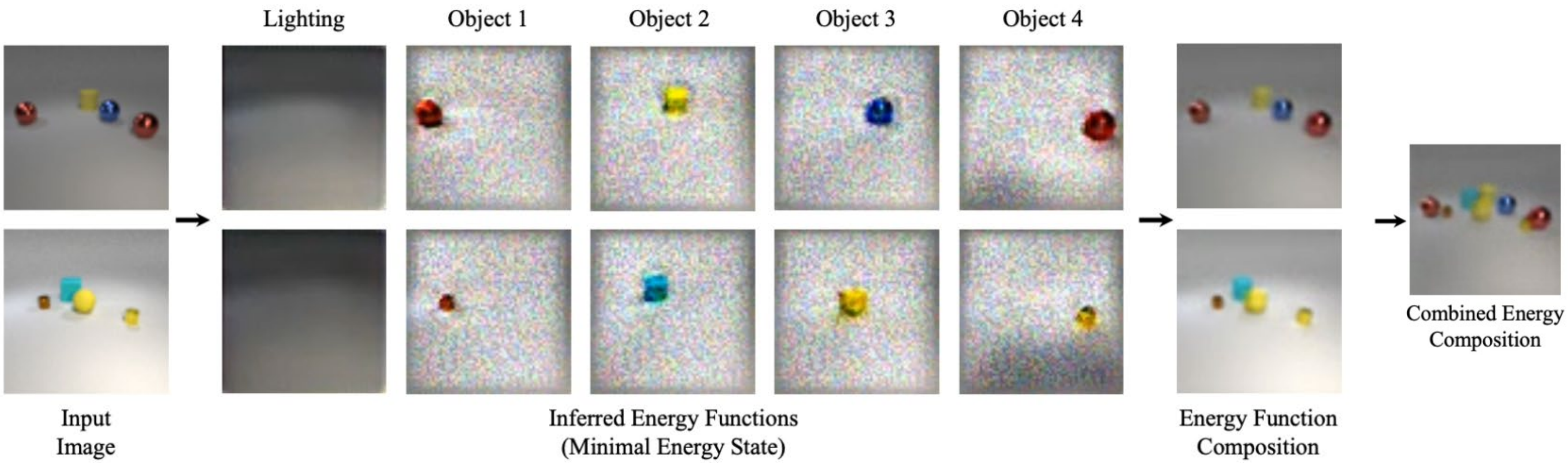


Recombination of individual energy functions.

[1] Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, Igor Mordatch. Unsupervised Learning of Compositional Energy Concepts. NeurIPS, 2021.

Application examples

Unsupervised Learning of Compositional Energy Concepts

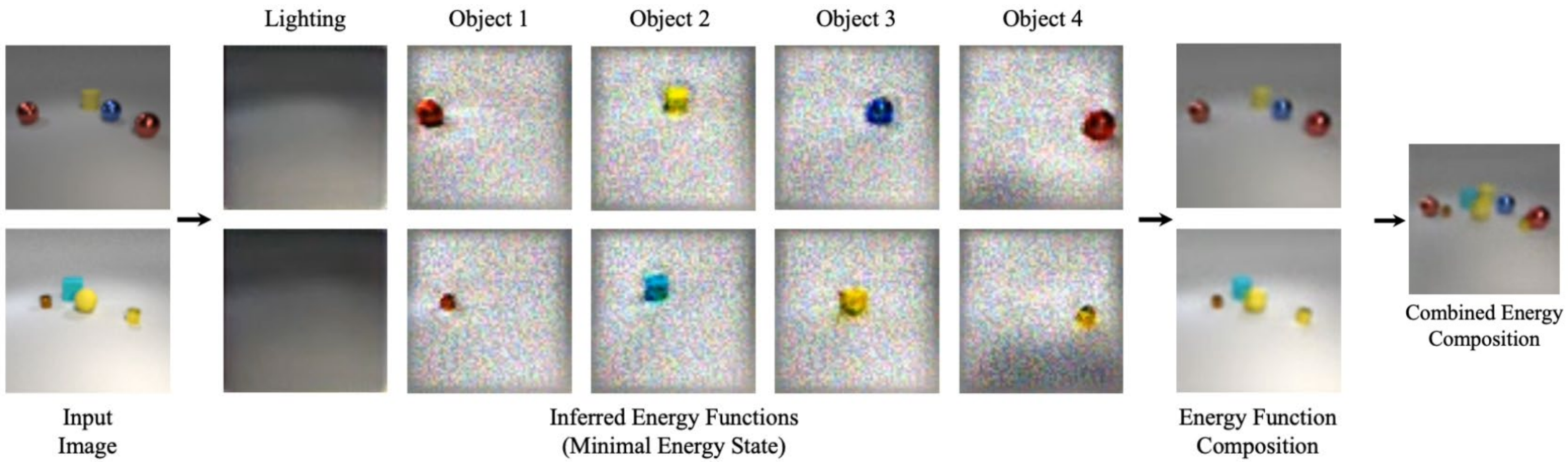


Decompose and recombine energy functions representing both local and global factors

[1] Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, Igor Mordatch. Unsupervised Learning of Compositional Energy Concepts. NeurIPS, 2021.

Application examples

Unsupervised Learning of Compositional Energy Concepts



Decompose and recombine energy functions representing both local and global factors

[1] Yilun Du, Shuang Li, Yash Sharma, Joshua B. Tenenbaum, Igor Mordatch. Unsupervised Learning of Compositional Energy Concepts. NeurIPS, 2021.

Conclusion and Take-Away Points

Conclusion and Take-Away Points

$$q_\theta(\mathrm{x}) = \frac{1}{\int \exp(-E_\theta(\mathrm{x}))\, d\mathrm{x}} \exp(-E_\theta(\mathrm{x})) = \frac{1}{Z_\theta} \exp(-E_\theta(\mathrm{x}))$$

→ Energy-based Models (EBMs) are powerful generative methods for capturing characteristics, regularities, and constraints of (high-dim.) data distributions

→ The energy function can be parameterized by almost any kind of model architecture!

→ Composition of energies is mostly straight-forward (product of experts, etc.)

→ There exist lots of fundamental connections to other model families (Diffusion Modeling, GANs, Normalizing Flows, Graphical Models, etc.)

But:

→ Likelihood-based learning is not trivial and oftentimes computationally expensive

→ Sampling is hard and training is oftentimes instable

Thank you for your attention!