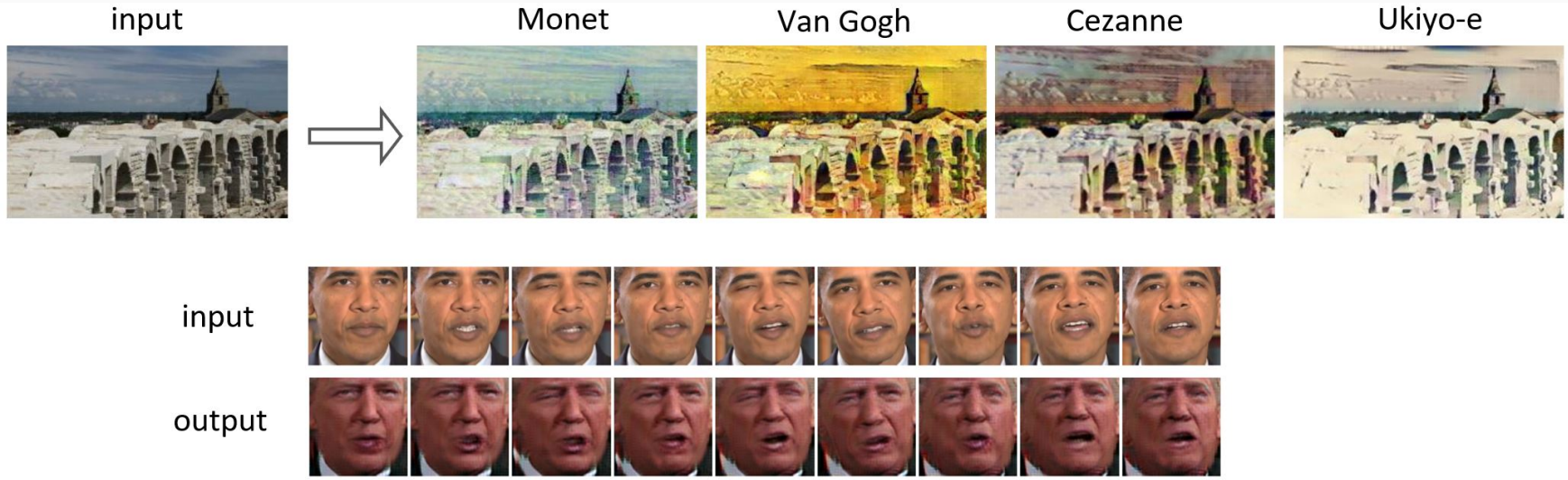
Learning Cycle-Consistent Cooperative Networks via Alternating MCMC Teaching for Unsupervised Cross-Domain Translation

Jianwen Xie[1*], Zilong Zheng[2*], Xiaolin Fang[3], Song-Chun Zhu[2,4,5], Ying Nian Wu[2]

[1] Cognitive Computing Lab, Baidu Research, Bellevue, USA
[2] University of California, Los Angeles, USA
[3] Massachusetts Institute of Technology, Cambridge, USA
[4] Tsinghua University, Beijing, China
[5] Peking University, Beijing, China

Introduction

- Cross-domain translation, such as image-to-image translation, has shown its importance in computer vision and computer graphics.
- Unsupervised cross-domain translation is more applicable than supervised cross-domain translation, because different domains of independent data collections are easily accessible.



Introduction

- Researchers have proposed unsupervised cross-domain translation networks based on GANs [1], such as CycleGAN [2], and obtained compelling results.
- Recently, learning ConvNet-parameterized EBM [3] (i.e., an energy-based model with the energy function parameterized by a convolutional neural network) for data probability distributions has received significant attention.

---

[1] Ian Goodfellow, et al. "Generative adversarial nets." NIPS 2014.

[2] Jun-Yan Zhu, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." ICCV. 2017.

[3] Jianwen Xie, et al. "A Theory of Generative ConvNet." ICML, 2016.

Introduction

- Cooperative network (CoopNets) [4] proposes to learn the EBM simultaneously with a generator in a cooperative learning scheme.
- In CoopNets, the generator plays a role of a fast sampler to initialize the MCMC sampling of the EBM, while the EBM teaches the generator via a finite-step MCMC.
- The CoopNets has several conceptual advantages:
  1. Avoid mode collapse.
  2. MCMC refinement.
  3. Fast-thinking and slow-thinking.

[4] Jianwen Xie, et al. "Cooperative Training of Descriptor and Generator Networks." TPAMI, 2018.

Introduction

The contributions of our paper are four-folds:

1. We propose a novel framework, `CycleCoopNets`, based on cooperative learning to study unsupervised cross-domain translation.
2. We successfully apply our framework to a wide range of applications of unsupervised image-to-image translation, including object transfiguration, season transfer, and art style transfer.
3. We show that our model can achieve comparative results with GAN-based and flow-based methods.
4. We further generalize our framework to the task of unsupervised image sequence translation.

Problem definition

Notation

Suppose we have two different data domains, say $\mathcal{X}$ and $\mathcal{Y}$, and data collections from these two different domains $\{x_i, i = 1, ..., n_x\}$ and $\{y_i, i = 1, ..., n_y\}$, where $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$. Let $p_{\text{data}}(x)$ and $p_{\text{data}}(y)$ denote the unknown data distributions for these two domains.

Problem

Without instance-level correspondence between two collections, we want to learn translators between two domains for cross-domain translation.

Latent variable model as a translator

Model for one-way translation, e.g., $\mathcal{Y} \to \mathcal{X}$.

We specify a mapping $G_{\mathcal{Y}\to\mathcal{X}}$ that seeks to re-express the image y in domain $\mathcal{Y}$ by the image x in domain $\mathcal{X}$.

Translator

The representational model is of the following form:

$$\begin{aligned} y &\sim p_{\text{data}}(y), \\ x &= G_{\mathcal{Y}\to\mathcal{X}}(y;\alpha_{\mathcal{X}}) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2 I_D), \end{aligned} \tag{1}$$

where $G_{\mathcal{Y}\to\mathcal{X}}(y;\alpha_{\mathcal{X}})$ is an encoder-decoder network whose parameters are $\alpha_{\mathcal{X}}$, and ϵ is a Gaussian residual. We assume σ is given and I_D is the D-dimensional identity matrix.

y is the latent variable of x, because for each $x \in \mathcal{X}$, its version $y \in \mathcal{Y}$ is unobserved. (x and y have the same number of dimension.)

Latent variable model as a translator

prior distribution : $p_{\text{data}}(y)$

conditional distribution : $q(x|y) \sim \mathcal{N}(G_{\mathcal{Y}\to\mathcal{X}}(y;\alpha_{\mathcal{X}}), \sigma^2 I_D)$

joint density : $q(x, y; \alpha_{\mathcal{X}}) = p_{\text{data}}(y) q(x|y; \alpha_{\mathcal{X}})$

marginal density : $q(x; \alpha_{\mathcal{X}}) = \int q(x, y; \alpha_{\mathcal{X}}) dy.$

The maximum likelihood estimation (MLE) requires a prior $p_{\text{data}}(y)$ with tractable density (e.g., Gaussian white noise distribution) to calculate

$$\frac{\partial}{\partial \alpha_{\mathcal{X}}} [\frac{1}{n} \sum_{i=1}^{n} \log q(x_i; \alpha_{\mathcal{X}})].$$

Due to the unknown prior $p_{\text{data}}(y)$, we can not estimate $\alpha_{\mathcal{X}}$ via MLE with an MCMC-based inference or a variational inference.

Energy-based model as a teacher

To avoid the challenging problem of inferring y from x, we can train $G_{\mathcal{Y}\to\mathcal{X}}$ via MCMC teaching [5] by recruiting an energy-based model (EBM) introduced by [Xie, et al 2016] as a teacher.

teacher

The EBM specifies the distribution of x explicitly by

$$p(x;\theta_{\mathcal{X}}) = \frac{1}{Z(\theta_{\mathcal{X}})} \exp\left[f(x;\theta_x)\right] p_0(x), \tag{2}$$

where $p_0(x) \propto \exp(-||x||^2/2s^2)$ is a Gaussian reference distribution. The energy function $\mathcal{E}(x;\theta_{\mathcal{X}}) = -f(x;\theta_{\mathcal{X}}) + ||x||^2/2s^2$, where f is parametrized by a bottom-up deep neural network with parameters $\theta_{\mathcal{X}}$. $Z(\theta_{\mathcal{X}}) = \int \exp\left[f(x;\theta_{\mathcal{X}})\right] p_0(x)dx$ is the intractable normalizing constant.

[5] Jianwen Xie, et al. "Cooperative Learning of Energy-Based Model and Latent Variable Model via MCMC Teaching." AAAI, 2018.

Energy-based model as a teacher

Maximum Likelihood Estimation

In general, the EBM $p(x;\theta_{\mathcal{X}})$ can be learned by maximum likelihood, which equivalently minimizes the KL-divergence $\mathrm{KL}(p_{\mathrm{data}}(x)\|p(x;\theta_{\mathcal{X}}))$ over $\theta_{\mathcal{X}}$. The gradient is given by

$$\begin{aligned} &-\frac{\partial}{\partial\theta_{\mathcal{X}}}\mathrm{KL}(p_{\mathrm{data}}(x)\|p(x;\theta_{\mathcal{X}})) \\ =&\mathrm{E}_{p_{\mathrm{data}}(x)}\left[\frac{\partial}{\partial\theta_{\mathcal{X}}}f(x;\theta_{\mathcal{X}})\right]-\mathrm{E}_{p(x;\theta_{\mathcal{X}})}\left[\frac{\partial}{\partial\theta_{\mathcal{X}}}f(x;\theta_{\mathcal{X}})\right], \end{aligned} \tag{3}$$

where $\mathrm{E}_{p(x;\theta_{\mathcal{X}})}$ denotes the expectation with respect to $p(x;\theta_{\mathcal{X}})$, which is analytically intractable.

Energy-based model as a teacher

Analysis by Synthesis

In practice, the gradient in Eq.(3) can be approximated by

$$\frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial\theta}f(x_i;\theta)-\frac{1}{n}\sum_{i=1}^{\tilde{n}}\frac{\partial}{\partial\theta}f(\tilde{x}_i;\theta), \tag{4}$$

where $\tilde{x}_i$ are MCMC examples sampled from the current distribution $p(x;\theta_{\mathcal{X}})$. With an EBM model defined on domain $\mathcal{X}$, we can sample x by MCMC, such as Langevin dynamics, which iterates

$$x_{\tau+1}=x_\tau-\frac{\delta^2}{2}\frac{\partial}{\partial x}\mathcal{E}(x_\tau;\theta_{\mathcal{X}})+\delta U_\tau, \tag{5}$$

where τ indexes the time step, δ is the step size, and $U_\tau\sim\mathrm{N}(0,I_D)$.

Translator and teacher

MCMC teaching

(1) Let $\mathcal{M}_{\theta_{\mathcal{X}}}$ be the Markov transition kernel of l steps of Langevin dynamics that samples from $p(x; \theta_{\mathcal{X}})$, and $\mathcal{M}_{\theta_{\mathcal{X}}} q_{\alpha_{\mathcal{X}}}$ be the marginal distribution obtained by running the Markov transition $\mathcal{M}_{\theta_{\mathcal{X}}}$ from distribution $q(x, \alpha_{\mathcal{X}})$. That is we use the translator $q(x, \alpha_{\mathcal{X}})$ to initialize the Langevin dynamics of p.

(2) The EBM $p(x; \theta_{\mathcal{X}})$ can distill its MCMC algorithm to $q(x; \alpha_{\mathcal{X}})$ through MCMC teaching, which seeks to find α at time t to minimize $\mathrm{KL}(\mathcal{M}_{\theta_{\mathcal{X}}} q_{\alpha_{\mathcal{X}}^{(t)}} | q_{\alpha_{\mathcal{X}}})$. That is, $q(x; \alpha_{\mathcal{X}})$ gets close to $p(x; \theta_{\mathcal{X}})$.

Cycle-consistent cooperative network

Problem

Given two domains $\mathcal{X}$ and $\mathcal{Y}$ in the absence of paired training examples, we learn translators between two domains for cross-domain translation .

The proposed solution

We propose the Cycle-consistent cooperative network (`CycleCoopNets`), to simultaneously learn and align two translator-critic pairs, i.e.,

$$\mathcal{Y} \to \mathcal{X} : \{p(x; \theta_{\mathcal{X}}), G_{\mathcal{Y}\to\mathcal{X}}(y; \alpha_{\mathcal{X}})\},$$
$$\mathcal{X} \to \mathcal{Y} : \{p(y; \theta_{\mathcal{Y}}), G_{\mathcal{X}\to\mathcal{Y}}(x; \alpha_{\mathcal{Y}})\},$$

where each pair of models is trained via MCMC teaching to form a one-way translation. We align them by enforcing mutual invertibility, i.e.,

$$x_i = G_{\mathcal{Y}\to\mathcal{X}}(G_{\mathcal{X}\to\mathcal{Y}}(x_i; \alpha_{\mathcal{Y}}); \alpha_{\mathcal{X}}),$$
$$y_i = G_{\mathcal{X}\to\mathcal{Y}}(G_{\mathcal{Y}\to\mathcal{X}}(y_i; \alpha_{\mathcal{X}}); \alpha_{\mathcal{Y}}).$$

Unsupervised cooperative translation (Overview)

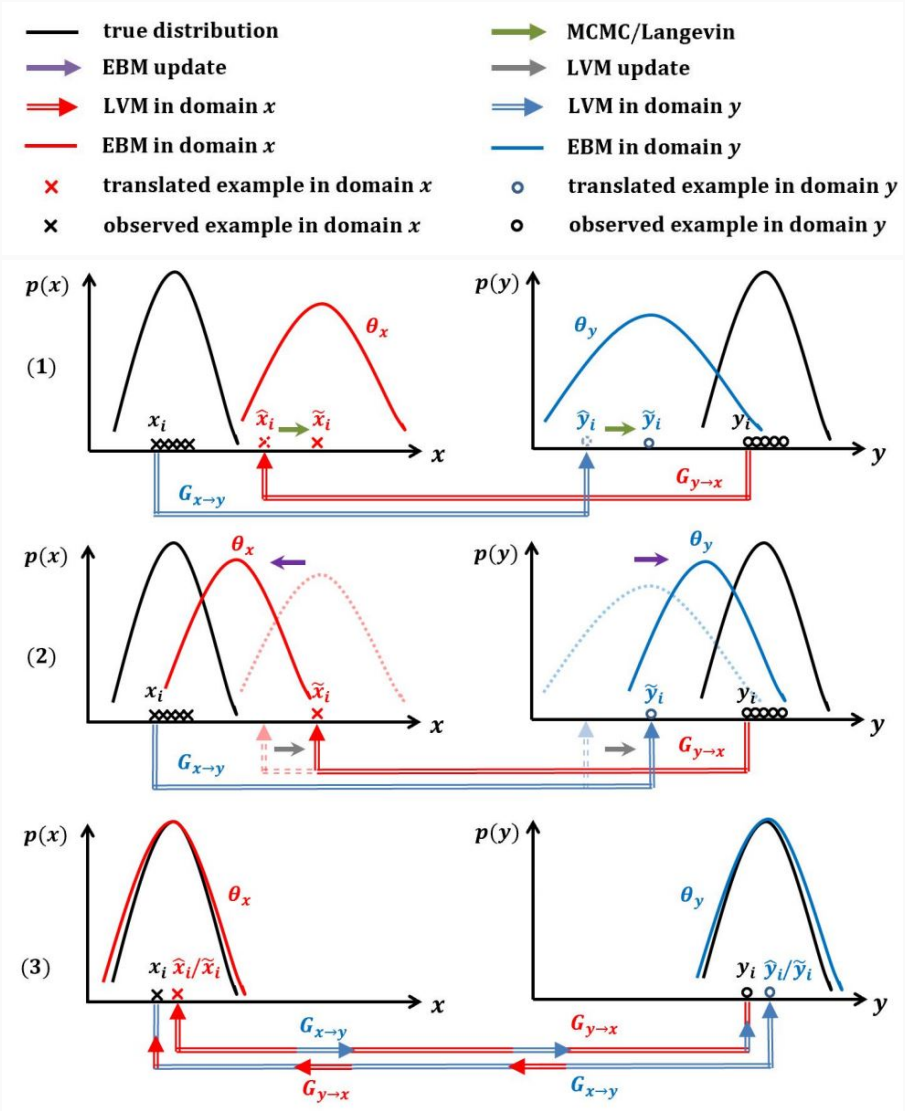


Figure 1: Illustration of `CycleCoopNetss`

We iterate the following three steps:
(1) Cross-domain mapping
(2) Density shifting
(3) Mapping shifting with cycle consistency

Unsupervised cooperative translation (Step 1)

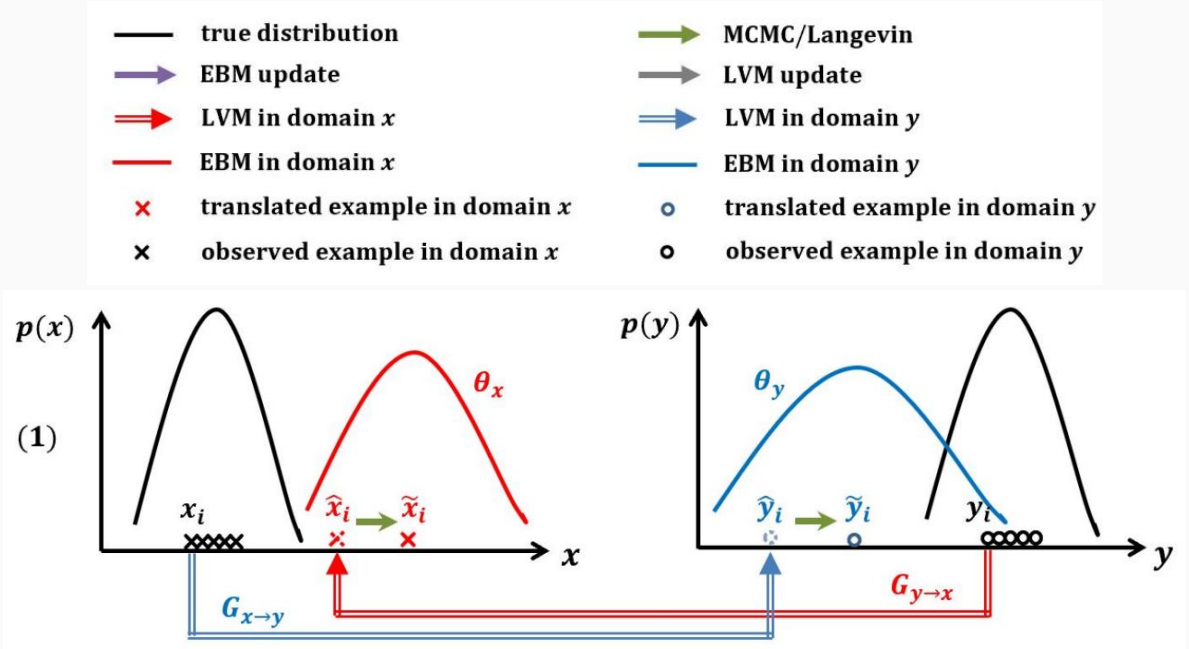


Figure 2: Step (1): cross-domain mapping

$$\{x_i \sim p_{data}(x)\}_{i=1}^{\tilde{n}} \ \{\hat{y}_i = G_{\mathcal{X}\to\mathcal{Y}}(x_i;\alpha_{\mathcal{Y}})\}_{i=1}^{\tilde{n}}.$$

$$\{y_i \sim p_{data}(y)\}_{i=1}^{\tilde{n}} \ \{\hat{x}_i = G_{\mathcal{Y}\to\mathcal{X}}(y_i;\alpha_{\mathcal{X}})\}_{i=1}^{\tilde{n}}$$

Unsupervised cooperative translation (Step 2)

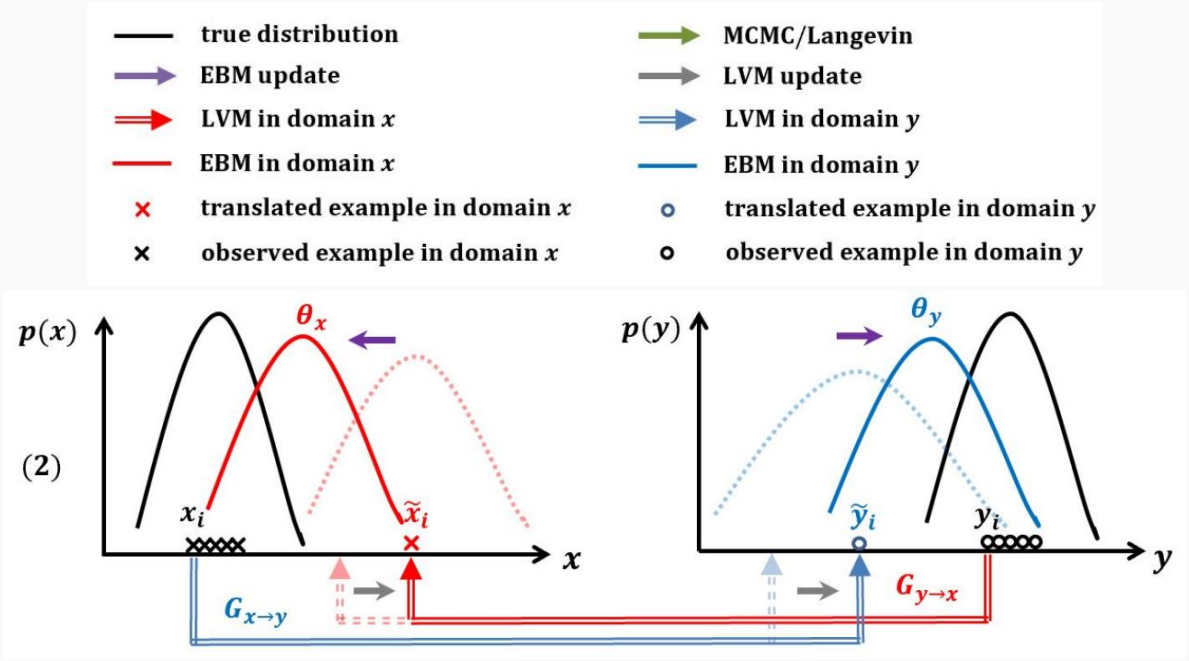


Figure 3: Step (2): density shifting

Starting from $\{\hat{y}_i\}_{i=1}^{\tilde{n}}$, run l steps of Langevin revision to obtain $\{\tilde{y}_i\}_{i=1}^{\tilde{n}}$.
Starting from $\{\hat{x}_i\}_{i=1}^{\tilde{n}}$, run l steps of Langevin revision to obtain $\{\tilde{x}_i\}_{i=1}^{\tilde{n}}$.

Given $\{x\}_{i=1}^{\tilde{n}}$ and $\{\tilde{x}\}_{i=1}^{\tilde{n}}$, update $\theta_{\mathcal{X}}^{(t+1)} = \theta_{\mathcal{X}}^{(t)} + \gamma_{\theta_{\mathcal{X}}} \Delta(\theta_{\mathcal{X}}^{(t)})$.
Given $\{y\}_{i=1}^{\tilde{n}}$ and $\{\tilde{y}\}_{i=1}^{\tilde{n}}$, update $\theta_{\mathcal{Y}}^{(t+1)} = \theta_{\mathcal{Y}}^{(t)} + \gamma_{\theta_{\mathcal{Y}}} \Delta(\theta_{\mathcal{Y}}^{(t)})$.

Unsupervised cooperative translation (Step 3)

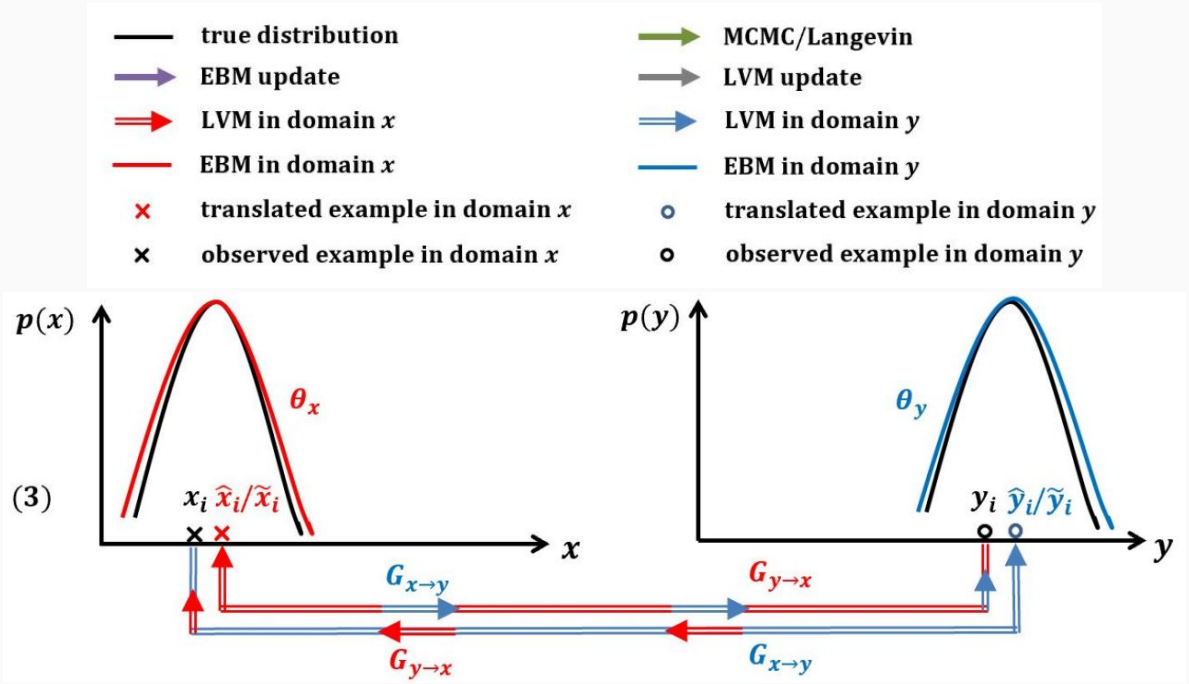


Figure 4: Step (3): Mapping shifting with cycle consistency

$$L_{teach}(\alpha_{\mathcal{X}}) = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} \|\tilde{x}_i - G_{\mathcal{Y}\to\mathcal{X}}(y_i, \alpha_{\mathcal{X}})\|^2.$$
$$L_{teach}(\alpha_{\mathcal{Y}}) = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} \|\tilde{y}_i - G_{\mathcal{X}\to\mathcal{Y}}(x_i, \alpha_{\mathcal{Y}})\|^2.$$

$$L_{\text{cycle}}(\alpha_{\mathcal{X}}, \alpha_{\mathcal{Y}}) =$$
$$\frac{1}{\tilde{n}} \sum_{i=1}^{n} \|x_i - G_{\mathcal{Y}\to\mathcal{X}}(G_{\mathcal{X}\to\mathcal{Y}}(x_i; \alpha_{\mathcal{Y}}); \alpha_{\mathcal{X}})\|_1 + \frac{1}{\tilde{n}} \sum_{i=1}^{n} \|y_i - G_{\mathcal{X}\to\mathcal{Y}}(G_{\mathcal{Y}\to\mathcal{X}}(y_i; \alpha_{\mathcal{X}}); \alpha_{\mathcal{Y}})\|_1.$$

Experiment 1: Object transfiguration

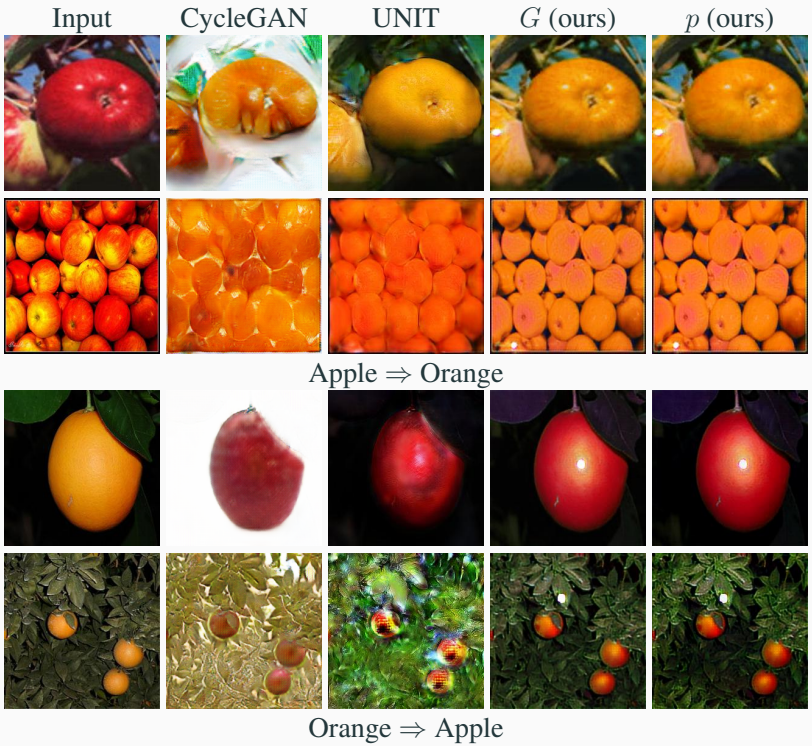


Figure 5: Object transfiguration. The top panel displays the translation from apples to oranges, and the bottom panel displays the translation from oranges to apples.

Table 1: Quantitative evaluation on apple⇔orange dataset with respect to Fréchet Inception Distance (FID) and Domain-invariant Perceptual Distance (DIPD). The top two rows show baseline results of CycleGAN and UNIT. The middle two rows show the results of G and p, where *s_step* is the number of MCMC teaching steps. The last three rows show performances of models with different numbers of MCMC teaching steps.

methods	**apple ⇒ orange**		**orange ⇒ apple**	
	FID ↓	DIPD ↓	FID↓	DIPD ↓
CycleGAN (Zhu et al. 2017)	160.78	1.75	143.87	1.73
UNIT (Liu et al. 2017)	170.66	1.58	122.04	1.62
$G(s\_step = 15)$	158.66	1.28	119.27	1.34
$p(s\_step = 15)$	**154.58**	**1.23**	**118.82**	**1.25**
$p(s\_step = 1)$	192.60	1.43	143.00	1.42
$p(s\_step = 5)$	166.41	1.43	170.38	1.40
$p(s\_step = 10)$	189.60	1.32	141.60	1.32

Experiment 2: Season transfer

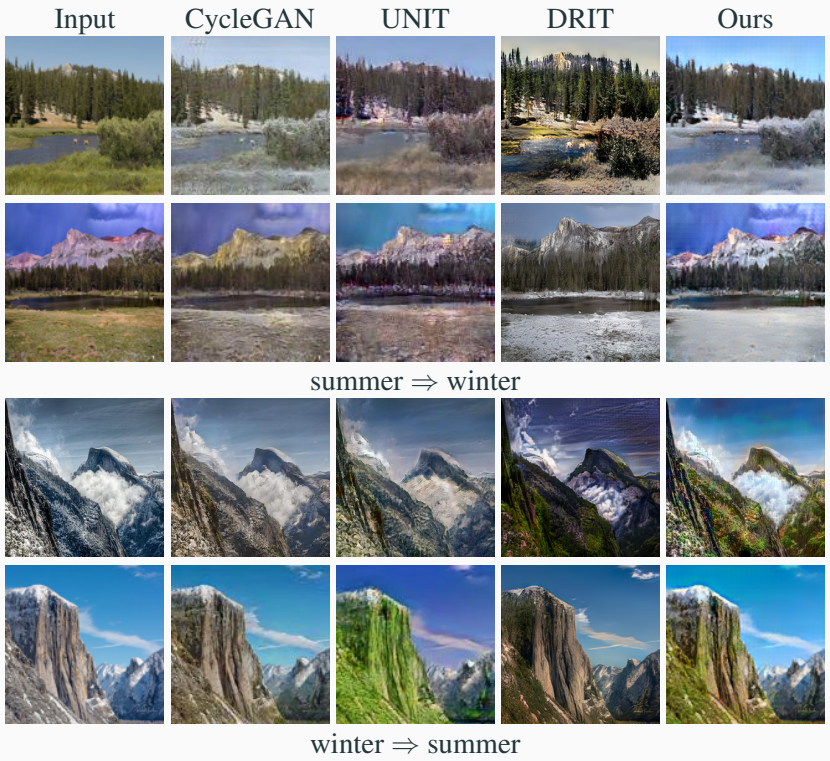
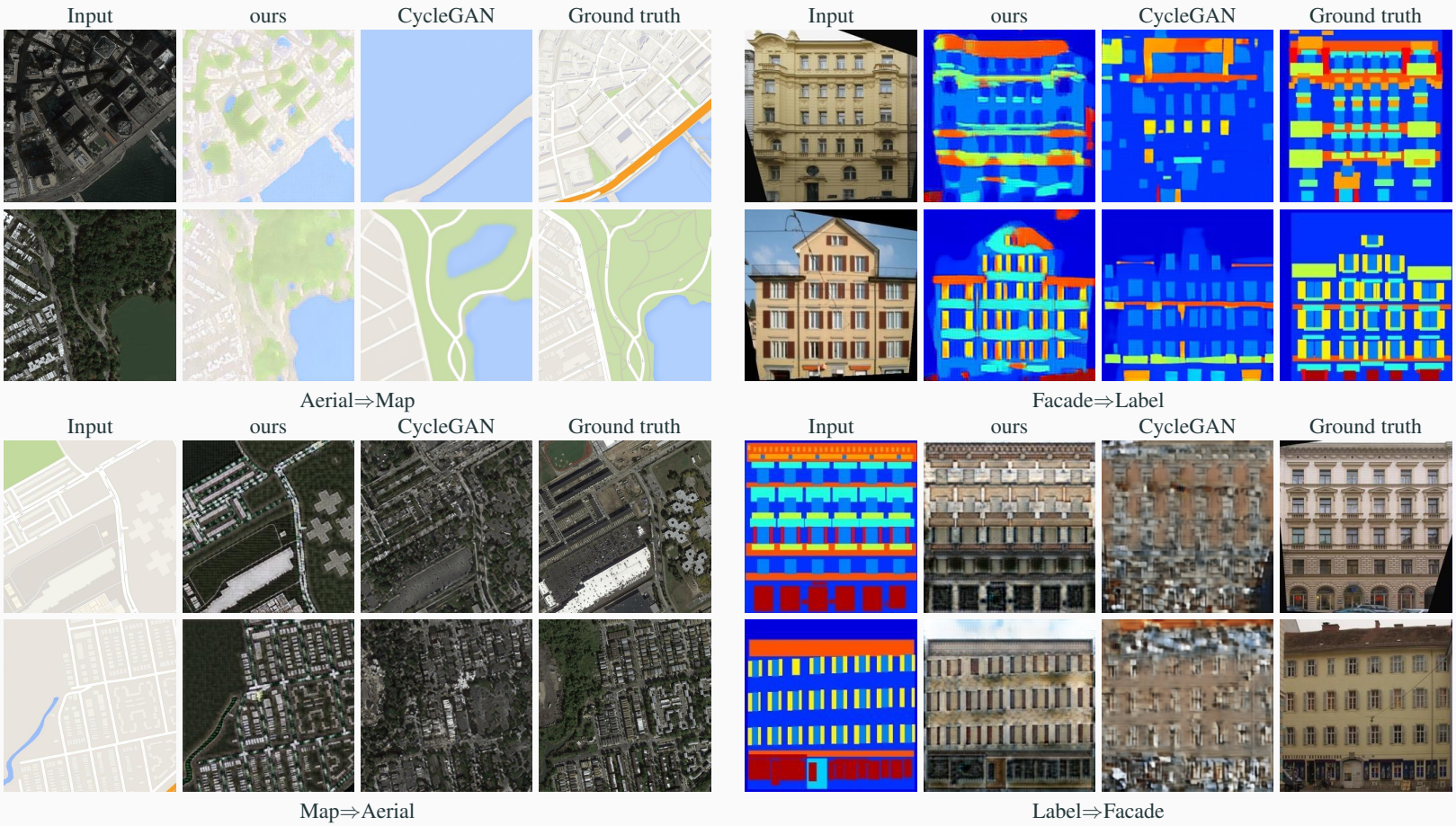


Figure 6: Season transfer. Example results of unpaired image-to-image translation on summer and winter Yosemite photos.

We train the `CycleCoopNetss` model on 854 winter photos and 1,273 summer photos of Yosemite for season transfer. Figure 6 shows some qualitative results and compares against three baseline methods CycleGAN, UNIT and DRIT (Lee et al. 2020).

Experiment 3: Translation between photo image and semantic label image



(a) Aerial ⇔ Map **(b)** Facade ⇔ Label

Figure 7: Translation between photo image and semantic label image.

Experiment 3: Translation between photo image and semantic label image

We show a comparison of performances of our method and some baselines, which includes CycleGAN and AlignFlow [6].

Table 2: Quantitative evaluation of unsupervised image-to-image translation by PSNR.

Dataset	Model	↑ L⇒R	↑ R⇒L
Aerial⇔Map	CycleGAN	21.59	12.67
	AlignFlow(mle)	19.47	13.60
	AlignFlow(adv)	20.16	**15.17**
	Ours	**22.29**	14.50
Facade⇔Label	CycleGAN	6.68	7.61
	AlignFlow(mle)	6.47	8.26
	AlignFlow(adv)	7.74	11.74
	Ours	**9.34**	**11.93**

These two datasets provide one-to-one paired images. In this experiment, we train our model on the datasets in an unsupervised manner, where the correspondence information between two image domains is omitted. We only use this correspondence information at testing stage for evaluation.

[6] Grover, Aditya, et al. "AlignFlow: Cycle Consistent Learning from Multiple Domains via Normalizing Flows." AAAI. 2020.

Experiment 4: Style transfer

Figure 8: Collection style transfer from photo realistic images to artistic styles.

Experiment 4: Style transfer

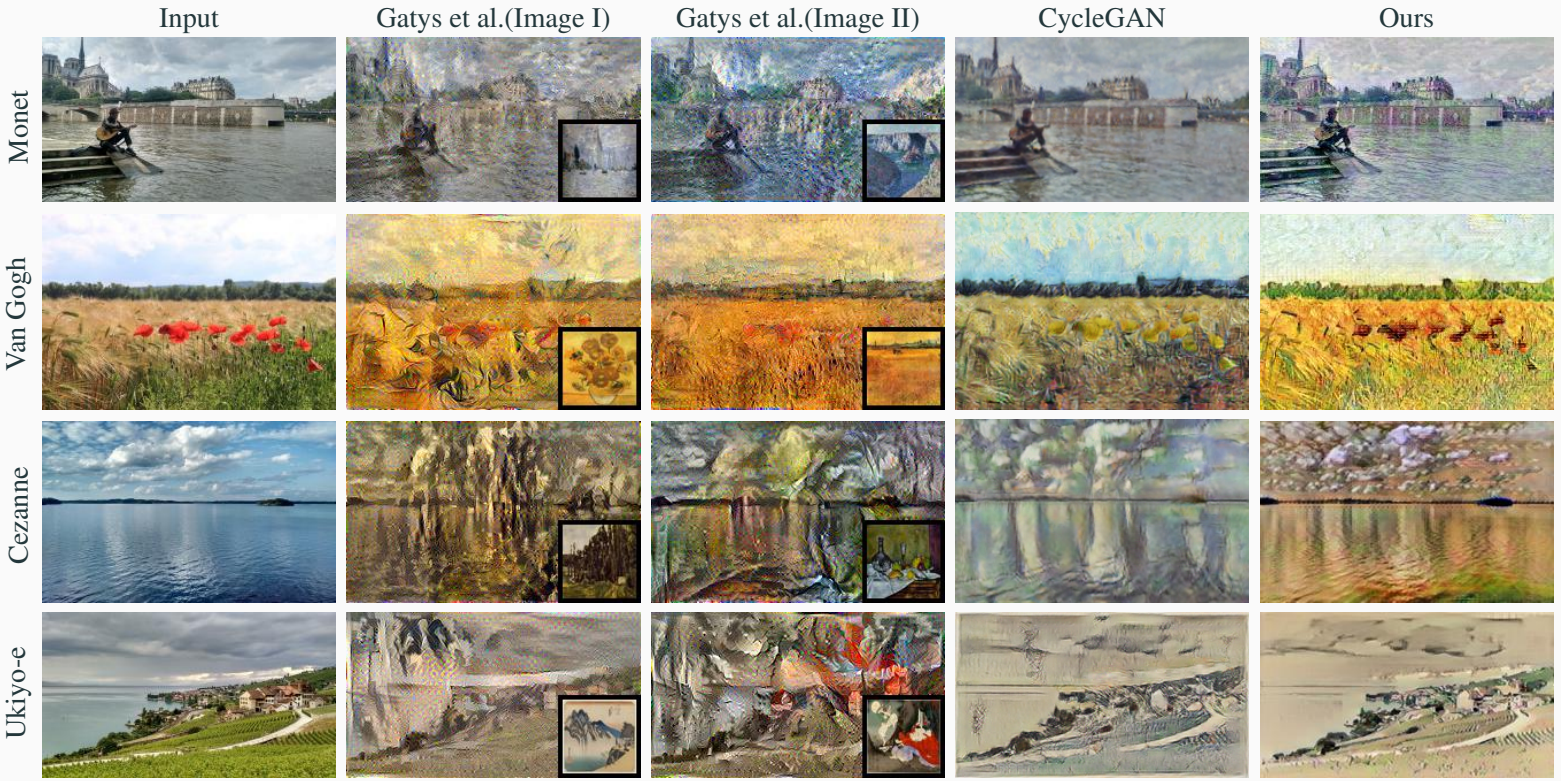


Figure 9: We compare our framework with style transfer method using neural network (Gatys, Ecker, and Bethge 2016) on photo stylization. Each row represents one example, where the first column shows the input image, the second and the third columns show results from (Gatys, Ecker, and Bethge 2016) using two different representative artworks as style images, the fourth column displays the result of CycleGAN, and the last one is the result by our method.

Experiment 5: Image sequence translation

We can further generalize the `CycleCoopNets` framework to learning a translation between two domains of sequences where paired examples are unavailable.

For example, given an image sequence of Donald Trump's speech, we can translate it to an image sequence of Barack Obama, where the content of Donald Trump is transferred to Barack Obama but the speech is in Donald Trump's style.

Such an appearance translation and motion style preservation framework may have a wide range of applications in video manipulation.

Experiment 5: Image sequence translation

Suppose we observe two unpaired but ordered image sequences $X = (x_1, x_2, ..., x_t, ...)$ and $Y = (y_1, y_2, ..., y_t, ...)$.

Each long sequence can be turned into a collection of short sequences with an equal length, i.e., $\{x_{t:t+k}\}_{t=1}^{T_{\mathcal{X}}}$ and $\{y_{t:t+k}\}_{t=1}^{T_{\mathcal{Y}}}$, where $x_{t:t+k} = (x_t, ..., x_{t+k})$ and $y_{t:t+k} = (y_t, ..., y_{t+k})$.

We make two modifications to adapt the `CycleCoopNets` to this new task:

Experiment 5: Image sequence translation

(1) We learn a temporal prediction model in each domain to predict future image frame given the past image frames in a sequence.

Let $R_{\mathcal{X}}$ and $R_{\mathcal{Y}}$ denote temporal prediction models for domain $\mathcal{X}$ and $\mathcal{Y}$ respectively. We learn $R_{\mathcal{X}}$ and $R_{\mathcal{Y}}$ by minimizing

$$
\begin{aligned}
L_{\mathrm{tp}}(R_{\mathcal{X}}) &= \frac{1}{T_{\mathcal{X}}} \sum_{t=1}^{T_{\mathcal{X}}} \|x_{t+k} - R_{\mathcal{X}}(x_{t:t+k-1})\|_1, \\
L_{\mathrm{tp}}(R_{\mathcal{Y}}) &= \frac{1}{T_{\mathcal{Y}}} \sum_{t=1}^{T_{\mathcal{Y}}} \|y_{t+k} - R_{\mathcal{Y}}(y_{t:t+k-1})\|_1,
\end{aligned}
\tag{6}
$$

where $x_{t:t+k} = (x_t, ..., x_{t+k})$ and $y_{t:t+k} = (y_t, ..., y_{t+k})$.

Experiment 5: Image sequence translation

(2) With the temporal prediction models, we modify the loss for G to take into account spatial-temporal information as below

$$
\begin{aligned}
& L_{\mathrm{st}}(G_{\mathcal{X}\to\mathcal{Y}}, R_{\mathcal{Y}}, G_{\mathcal{Y}\to\mathcal{X}}) \\
=& \frac{1}{T_{\mathcal{X}}} \sum_{t=1}^{T_{\mathcal{X}}} \|x_{t+k} - G_{\mathcal{Y}\to\mathcal{X}}(R_{\mathcal{Y}}(G_{\mathcal{X}\to\mathcal{Y}}(x_{t:t+k-1})))\|_1, \\
& L_{\mathrm{st}}(G_{\mathcal{Y}\to\mathcal{X}}, R_{\mathcal{X}}, G_{\mathcal{X}\to\mathcal{Y}}) \\
=& \frac{1}{T_{\mathcal{Y}}} \sum_{t=1}^{T_{\mathcal{Y}}} \|y_{t+k} - G_{\mathcal{X}\to\mathcal{Y}}(R_{\mathcal{X}}(G_{\mathcal{Y}\to\mathcal{X}}(y_{t:t+k-1})))\|_1,
\end{aligned}
$$

The final objective of G and R is given by

$$
\begin{aligned}
& \min_{G,R} L(G,R) = L_{\mathrm{teach}}(G_{\mathcal{Y}\to\mathcal{X}}) + L_{\mathrm{teach}}(G_{\mathcal{X}\to\mathcal{Y}}) \\
& + \lambda_1 L_{\mathrm{tp}}(R_{\mathcal{X}}) + \lambda_1 L_{\mathrm{tp}}(R_{\mathcal{Y}}) + \lambda_2 L_{\mathrm{st}}(G_{\mathcal{X}\to\mathcal{Y}}, R_{\mathcal{Y}}, G_{\mathcal{Y}\to\mathcal{X}}) \\
& + \lambda_2 L_{\mathrm{st}}(G_{\mathcal{Y}\to\mathcal{X}}, R_{\mathcal{X}}, G_{\mathcal{X}\to\mathcal{Y}}),
\end{aligned}
$$

Experiment 5: Image sequence translation

Figure 10: Image sequence translation. (a) We translate Barack Obama's facial motion to Donald Trump. (b) We translate from the blooming of a violet flower to a yellow flower. (c) We translate the blooming of a purple flower to a red flower.

Conclusion

- This paper studies unsupervised cross-domain translation problem based on a cooperative learning scheme.
- Our framework consist of two cooperative networks, each of which jointly trains an latent variable model as a translator and an energy-based model as a critic to account for one domain distribution.
- Two cooperative networks that model data distributions of two different domains are simultaneously learned and aligned by the proposed alternating MCMC teaching algorithm.
- Experiments show that the proposed framework can be useful for different unsupervised cross-domain translation tasks.