Multivariate-Information Adversarial Ensemble for Scalable Joint Distribution Matching

Ziliang Chen*
Sun Yat-sen University
c.ziliang@yahoo.com

Zhanfu Yang*
Purdue University
Yang1676@purdue.edu

Xiaoxi Wang*
Sun Yat-sen University
wangxx35@mail2.sysu.edu.cn

Xiaodan Liang
Sun Yat-sen University
xdliang328@gmail.com

Xiaopeng Yan
Sun Yat-sen University
yanxp3@mail2.sysu.edu.cn

Guanbin Li
Sun Yat-sen University
liguanbin@mail.sysu.edu.cn

Liang Lin *
Sun Yat-sen University
linliang@ieee.org

Abstract

A broad range of cross-m-domain generation researches boil down to matching a joint distribution by deep generative models (DGMs). Hitherto algorithms excel in pairwise domains while as m increases, remain struggling to scale themselves to fit a joint distribution. In this paper, we propose a domain-scalable DGM, *i.e.*, MMI-ALI for m-domain joint distribution matching. As an m-domain ensemble model of ALIs [1], MMI-ALI is adversarially trained with maximizing *Multivariate Mutual Information* (MMI) *w.r.t.* joint variables of each pair of domains and their shared feature. The negative MMIs are upper bounded by a series of feasible losses that provably lead to matching m-domain joint distributions. MMI-ALI linearly scales as m increases and thus, strikes a right balance between efficacy and scalability. We evaluate MMI-ALI in diverse challenging m-domain scenarios and verify its superiority.

1 Introduction

Remarkable advances of Deep Generative Models (DGMs), *e.g.*, *Generative Adversarial Net* (GAN) [2], give rise to a variety of cross-domain generation and transfer tasks, *e.g.*, label-to-image translation [3, 4], visual / text style transfers [5, 6], *etc*. In these scenarios, examples drawn from one domain transform their appearances via DGMs to synthesize the data patterns that belong to the other domains. This magic is formally interpreted as learning a joint distribution *w.r.t.* multi-domain random variables. Specifically, suppose that m ($\forall m \in \mathbb{N}_+$) domains underly marginal distributions $\{p_1, \cdots, p_m\}$. Given an example $\boldsymbol{x}_i \sim p_i$ ($\forall i \in [m] = \{1, \cdots, m\}$), DGMs generate $\boldsymbol{x}_j$ ($\forall j \in [m], j \neq i$) to satisfy the equation:

$$\begin{aligned} p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m) :&= p(\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i)p(\boldsymbol{x}_i) \\ &= p_\Theta(\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i)p(\boldsymbol{x}_i) \end{aligned} \tag{1}$$

where $p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$ denotes the joint distribution on m-domain random variables. $p(\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i)$ is the conditional distribution *w.r.t.* $\boldsymbol{x}_i$, and $p_\Theta(\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i)$ is parametrized from DGMs to match the m-domain joint distribution (Θ indicates the parameters of those DGMs). Eq.1 is connected with a broad set of GAN-based DGMs. Particularly when $m = 2$,

*indicates equal contribution. Correspondence: Liang Lin.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

(1) refers to finding a pair of generation nets to model $p(\boldsymbol{x}_2|\boldsymbol{x}_1)$ and $p(\boldsymbol{x}_1|\boldsymbol{x}_2)$, exactly the learning goal shared by c-GAN [3], CycleGAN [6, 7, 8] and other DGM methods [1, 9].

Despite rapid progresses in learning paired-domain joint distribution, existing DGMs seldom prepare for the challenges as m>2, notably, the balance between model efficacy and scalability. On one hand, to cover $m(m-1)$ cross-domain transfer cases, most DGMs, *e.g.*, CycleGAN and JointGAN [10], have to deploy the same amount of (or even more) generation nets to learn m-domain joint distributions. It lacks efficiency in parameters and in turn, hinders them to capture richer information to improve their performances. On the other hand, recent heuristic methods, *i.e.*, StarGAN [11], attempt to suit all the transfer tasks by a single pipeline where each domain is treated as a class. Their pipelines are indeed scalable but the algorithms do not promise them to learn joint distributions. In fact, this line of methods can be technically fragile: If the supports of $\{p_i\}_{i=1}^m$ tend to intersect, treating domains as classes will fail and arouse serious model collapse.

In this paper, we focus on matching a m-domain joint distribution in a scalable and effective way. Instead of hacking a complex DGM pipeline, we revisit a famous *Adversarially Learned Inference* (ALI) [1] model from a prospective of ensemble [12].We assign m ALIs (allowed to share some of parameters) to each domain for learning m domain marginals by sharing their feature variables. By this mutual feature variable, each sample from domain i can be encoded to a feature by the inference net in the i^{th} ALI, then mapped into the j^{th} domain ($j \neq i$) by the generation net in the j^{th} ALI. This m inference-generation ensemble enable $m(m-1)$ transfer cases and more importantly, may lead to m-domain joint distribution by appropriately regulating cross-domain dependency.

Specifically, we reframe this m-ALI ensemble trained with maximizing *multivariate mutual information* (MMI) [13, 14]. The MMIs act on arbitrary joint variables originating from each pair of domains and the domain-shared feature, which implies that m-domain information flow may exchange via their mutual feature. This observation nails down to a series of upper bounds that indicates conditional generation [3] and cycle consistency [6]. They are provably connected with matching a m-domain joint distribution and make the m-ALI ensemble our final model, *i.e.*, MMI-ALI.

MMI-ALI mainly contributes as:

1). MMI-ALI is linearly-scalable with m and more importantly, holds a series of loss upper bounds for provable joint distribution matching.

2). MMI-ALI revisit classical ALI from a view of ensemble model and learn with a adversarial ensemble loss (Sect.2.5), which are powerful for cross-domain generative modeling

3). A variety of m-domain experiments ($m \geq 2$) are placed in diverse scenarios, *e.g.*, 6-domain setup, visual / text style transfer, *etc*. The evaluation in supervised and unsupervised learning demonstrate the superiority of MMI-ALI.

Related work. Joint distribution matching has been considerably discussed in pairwise domain setups. Relevant researches based on GANs are classed into two lines. Models in the first line present as bidirectional DGMs associated with sample generation and feature inference, [1, 15, 16, 17], real-real domain translations, *e.g.*, CycleGANs [6, 7, 8], the variants [18, 19] and other adversarial dual learning models [20, 21]. When cross-real-domain data are given in pairs, the second branch is connected with c-GAN [3] and other conditional adversarial DGMs [22, 23, 24, 4] . [9] shows their relationships by conditional entropy (CE). Our paper extends it into m-domain scenarios.

In m-domain setup, joint distribution becomes more cumbersome to learn and a few of recent DGMs refer to this problem. To the best of our knowledge, JointGAN [10] is the only existing research that promises (1) when $m > 2$. JointGAN chases for fully learning joint distribution, but ignores the scalability when m increases and requires C_m^3 generative modules to attain $m(m-1)$ cross-domain transformations. StarGAN [11] and its variants [25, 26] use a domain-shared backbone where each domain is viewed as a class. They cast m-domain transfer to a category generation problem and do not aim to learn a joint distribution.

2 Multivariate Mutual Information Adversarially Learned Inference

In this section, we elaborate MMI-ALI in the following routine: **1).** We introduce ALI (Sect.2.1) and how it leads to an ensemble to achieve $m(m-1)$ cross-domain transfer tasks (Sect.2.2); **2).** We show the limitation of the m-ALI ensemble in cross-domain transfer (Sect.2.3) and how MMI induces a feasible regulation for the m-ALI ensemble to learn a joint distribution (Sect.2.4). **3).** We provide the adversarial ensembel learning algorithm of MMI-ALI (Sect.2.5). All proofs are deferred in our Appendix.A.

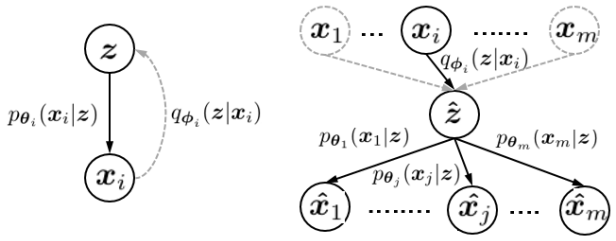


Figure 1: The overviews of ALI and m-ALI ensemble. MMI-ALI is learned from m-ALI ensemble with MMI constraints (Sect.2.4).

2.1 Preliminary: Adversarially Learned Inference

ALI is a bidirectional DGM derived from GAN, as it additionally incorporates an inference net trained with a generation net by playing against a discriminator. More specifically, in our context, suppose that a ALI model refers to generating a fake domain-i example $\hat{\boldsymbol{x}}_i$ ($\forall i \in [m]$). Without loss of generality, we employ a distribution $q(\boldsymbol{z})$ as a prior on feature space $\mathbb{R}^d$, e.g. $q(\boldsymbol{z}) = \mathcal{N}(\mathbf{0}^d, \mathbf{I}^{d\times d})$. Under the nonparametric assumption, we present the generation and inference nets by conditional distributions $p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})$ and $q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)$, where $\boldsymbol{\theta}_i$, $\boldsymbol{\phi}_i$ denote their parameters and their inputs $\boldsymbol{z}$, $\boldsymbol{x}_i$ are treated as the conditions. In this manner, ALI casts an adversarial game between $p_{\boldsymbol{\theta}_i}$, $q_{\boldsymbol{\phi}_i}$ and a $\boldsymbol{\omega}_i$-parameterized critic net (discriminator) $f_{\boldsymbol{\omega}_i}$ in

$$\begin{aligned}
&\min_{\boldsymbol{\theta_i},\boldsymbol{\phi}_i}\max_{\boldsymbol{\omega_i}} \mathcal{L}^{(i)}_{\text{ALI}}(\boldsymbol{\theta}_i,\boldsymbol{\phi}_i,\boldsymbol{\omega}_i) = \\
&\mathbb{E}_{\boldsymbol{x}_i\sim p(\boldsymbol{x}_i),\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)}\big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] \\
&\qquad + \mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),z\sim q(\boldsymbol{z})}[\log\big(1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})\big)]
\end{aligned} \tag{2}$$

where $(\boldsymbol{x}_i,\hat{\boldsymbol{z}})$ denotes a real domain-i example $\boldsymbol{x}_i$ with its corresponding feature $\hat{\boldsymbol{z}}$ inferred by $q_{\boldsymbol{\phi}_i}$ and $(\hat{\boldsymbol{x}}_i,\boldsymbol{z})$ denotes a fake domain-i sample $\hat{\boldsymbol{x}}_i$ generated from $\boldsymbol{z}\sim q(\boldsymbol{z})$ via $p_{\boldsymbol{\theta}_i}$. $f_{\boldsymbol{\omega}_i}(\cdot,\cdot)$ is a binary classifier that distinguishes each sample-feature joint pair drawn from either $q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})$ or $p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})$. The minimax objective (17) encourages the iterative update between $\boldsymbol{\omega}_i$ and $\boldsymbol{\theta}_i$, $\boldsymbol{\phi}_i$. Similar to GAN, their resulting saddle point promises marginal matching on $p(\boldsymbol{x}_i)$, $q(\boldsymbol{z})$.

Lemma 1 ([1]). *The optimal generation, inference and critic nets w.r.t.,*$\{\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*\}$ ($\forall i \in [m]$) *refer to a saddle point in Eq.17* $\iff$ $p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i|\boldsymbol{z})q(\boldsymbol{z}) = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i)p_i(\boldsymbol{x}_i)$.

2.2 m-ALI Ensemble

With regards to m domains, there can be m ALIs that share the feature variable $\boldsymbol{z}$ to make marginal matchings on their own. It inspires an ensemble that associates m domains to enable $m(m-1)$ cross-domain data transformations. As illustrated in Fig.1.Right, suppose that $\forall \boldsymbol{x}_i \sim p_i$ is demanded to transform to the other j^{th} domain ($\forall i,j \in [m]$, $j \neq i$). By the aid of inference net $q_{\boldsymbol{\phi}_i}$ in the i^{th} ALI, it is able to encode $\boldsymbol{x}_i$ into a domain-agnostic feature $\hat{\boldsymbol{z}}$, and then use the generation net $p_{\boldsymbol{\theta}_j}$ in the j^{th} ALI to decode $\hat{\boldsymbol{z}}$ into $\hat{\boldsymbol{x}}_j$. This cross-domain generative process can be formulated as:

$$\begin{aligned}
&p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i) \\
&= \int p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i}|\hat{\boldsymbol{z}},\boldsymbol{x}_i)p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)d\hat{\boldsymbol{z}} \\
&= \int \underbrace{\Big(\prod_{j\in[m]\&j\neq i} p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{z}})\Big)}_{\substack{\text{Given } \hat{\boldsymbol{z}},\ \{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i} \text{ and} \\ \boldsymbol{x}_i \text{ are independent}}} p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)d\boldsymbol{z} \\
&= \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{z}})q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)d\hat{\boldsymbol{z}},\ s.t.\forall i \in [m]
\end{aligned} \tag{3}$$

where we summarize the parameters of m-domain generation, inference, critic nets by $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_i\}_{i=1}^m$, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i\}_{i=1}^m$, $\boldsymbol{\Omega} = \{\boldsymbol{\omega}_i\}_{i=1}^m$. As a cross-$m$-domain generative model, the m-ALI ensemble in (3) presents two advantages.

- **Scalability**: (3) is linearly-scalable with m. For sub-nets $\{q_{\boldsymbol{\phi}_i}\}_{i=1}^m$ and $\{p_{\boldsymbol{\theta}_i}\}_{i=1}^m$, it is possible to share their high-level layers across domains, as m-domain ALIs share their feature variable $\boldsymbol{z}$.
- **Generative model capability**: According to Lemma.2, (3) with $\boldsymbol{\phi}_i^*$ and $\boldsymbol{\theta}_j^*$ promises the transformed item $\hat{\boldsymbol{x}}_j$ following the true domain marginal p_j:

Proposition 1. *Given a pair of domains $\forall i, j \in [m]$, $i \neq j$, their well-trained ALIs (in Lemma.1) construct a cross-domain transfer process $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)$ that satisfies*

$$p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j) = \int p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)p_i(\boldsymbol{x}_i)d\boldsymbol{x}_i = p_j(\hat{\boldsymbol{x}}_j)$$

where $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)$ is the parameterized marginal of (3).

2.3 MMI-ALI: Motivation

How to learn m-ALI ensemble. As we previously discuss, m-ALI ensemble is a promising non-parametric model to achieve $m(m-1)$ cross-domain transfer, as the scalability and generative model capability have verified its potential. But the vital problem is, how to encourage the m-ALI ensemble to learn a m-domain joint distribution. Obvisouly, since each ALI model in m-ALI ensemble is independently trained, no cross-domain dependencies enforce $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}$ to approximate the joint distribution $p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$. As long as generated data can match domain marginals (Proposition.1), (3) may tolerate all erratic cross-domain transfer. To tackle this problem, we first need to understand how to match a joint distribution in the m-domain scenario.

Criterion for m-domain joint distribution matching. In terms of supervised and unsupervised learning, joint distribution matching presents as satisfying different criterion. **1).** In *supervised learning*, we have access to draw samples from the true joint density $p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$ and each of them presents as a m-tuple. Hence $p(\{\boldsymbol{x}_i\}_{i=1}^m)$ can be learned by minimizing the log-likelihood estimator:

$$\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} \; -\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_i\}_{i=1}^m)\big] \tag{4}$$

2). In *unsupervised learning*, data across domains are unparalleledly aligned so that no access is provided to draw m-tuple from $p(\boldsymbol{x}_i, \cdots, \boldsymbol{x}_m)$. In the pairwise domain setup [6], the unsupervised learning is typically considered as a cross-domain data reproduction problem that decreasing their conditional entropy (CE) theoretically helps to solve (see more in **(author?)** 9):

$$\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} H\big(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j\big) = -\mathbb{E}_{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}}\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j)\big] \tag{5}$$

where $H\big(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j\big)$ measures the input reproduction uncertainty *w.r.t.* $\boldsymbol{x}_i$ in the condition of $\hat{\boldsymbol{x}}_j$, *i.e.*, what the input has produced. In our scenario, we develop (5) to incorporate m-domain variables

$$\begin{aligned} &\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} H\big(\boldsymbol{x}_i|\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i}\big) \\ &= -\mathbb{E}_{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}}\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i})\big] \end{aligned} \tag{6}$$

where $\forall i \in [m]$, $\boldsymbol{x}_i$ denotes an empirical draw from p_i; $\{\hat{\boldsymbol{x}}_j\}_{j=1\&j\neq i}^m$ denote fake items generated from $\boldsymbol{x}_i$ via (3).

It is worth noting that, (37) (38) with $m=2$ refer to condition [3] and cycle-consistency loss [6] that have been widely-used in GAN-based DGM. But in general cases ($m \geq 2$), they are typically intractable and disconnected with the learning algorithm of ALI.

Rather than directly optimizing (37) (38), we prefer exploring the information-theoretic meaning behind m-domain joint distribution. In the next subsection, we introduce *Multivariate Mutual Information* (MMI) and explain it in the m-ALI ensemble context. We derive feasible MMIs *w.r.t.* each pair of domains and feature. They refer to a series of upper bounds that can also be interpreted as condition and cycle losses. They result in (37) (38) to promise m-ALI ensemble learn for joint distribution matching.

2.4 MMI-Induced Regularization

Before diving into further technical analysis, let's quickly go through MMI, the pivotal ingredient of our regularization.

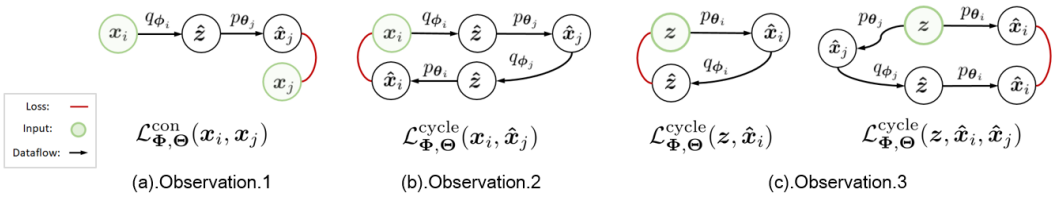


Figure 2: The diagram of constructing MMI-induced regularizations by generation and inference nets in m ALIs. Best viewed in color.

Multivariate Mutual Information (MMI). Given a pair of random variables $\boldsymbol{x}$, $\boldsymbol{y}$, *Mutual Information* (MI) $I(\boldsymbol{x};\boldsymbol{y})$ quantifies the amount of information one of them contains about the other, *i.e.*,

$$I(\boldsymbol{x};\boldsymbol{y}) = I(\boldsymbol{y};\boldsymbol{x}) := H(\boldsymbol{y}) - H(\boldsymbol{y}|\boldsymbol{x}) \tag{7}$$

. Maximizing $I(\boldsymbol{x};\boldsymbol{y})$ relates to an invertible function that knowing one of $\boldsymbol{x}$, $\boldsymbol{y}$ almost reveals the other. MMI extends MI by including n random variables $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_n$ ($\forall n \in \mathbb{N}_+$). It can be recursively defined as

$$\begin{aligned} &I(\boldsymbol{y}_1;\cdots;\boldsymbol{y}_n) \\ &\quad := I(\boldsymbol{y}_1;\cdots;\boldsymbol{y}_{n-1}) - I(\boldsymbol{y}_1;\cdots;\boldsymbol{y}_{n-1}|\boldsymbol{y}_n) \end{aligned} \tag{8}$$

where $I(\boldsymbol{y}_1;\cdots;\boldsymbol{y}_{n-1}|\boldsymbol{y}_n)$ denotes *Conditional Mutual Information* (CMI), the expectation of $I(\boldsymbol{y}_1;\cdots;\boldsymbol{y}_{n-1})$ when its value is conditioned on $\boldsymbol{y}_n$.

MMI for joint distribution matching. MMI resembles the information-theoretic sense of MI. Maximizing m-domain MMI with respect to densities parameterized by $\mathbf{\Phi}$, $\mathbf{\Theta}$, *i.e.*, $I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_1;\cdots;\boldsymbol{x}_m)$, intuitively encourages discovering an identical information flow from one domain to the others. It corresponds to the cross-domain transfer $p_{\mathbf{\Phi},\mathbf{\Theta}}$ under m-domain joint distribution matching. However, on the basis of the recursive routine in (8), m-variable MMI is comprised of $\mathcal{O}(2^m)$ entropy terms that can be positive or negative. It makes $I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_1;\cdots;\boldsymbol{x}_m)$ intractable and formidable to extend with m. Besides, it probably arouses unstable optimization, as $I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_1;\cdots;\boldsymbol{x}_m)$ may be unbounded.

Instead of simultaneously considering m-domain variables, we tend to explore the linear combination of MMIs on each pair of domain variables $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ with the m-domain-shared feature variable $\boldsymbol{z}$. In this principle, MMI $I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ has been covered $m(m-1)$ transfer cases and their maximizations are understood as

$$\min_{\mathbf{\Phi},\mathbf{\Theta}} \quad -\sum_{i,j\in[m],i\neq j} I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z}) \tag{9}$$

which implies the m-domain information flows exchange via their features. $I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ conceives two technical merits. First, three-variable MMI is always non-positive and thus, the minimization $-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ is lower bounded by 0, which substantially stabilizes the optimization process. Second, $-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ can be pushed into a line of upper bounds that serve as condition and cycle-consistency losses. Their minimization results in (37) (38) that encourages $p_{\mathbf{\Phi},\mathbf{\Theta}}$ to learn the m-domain joint distribution. We are going to elaborate them.

Upper bounds. Derived from ALIs, $-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ consists of generation and inference nets. Hence inputs underlie true distributions and may be drawn from either m domain marginals $\{p_i\}_{i=1}^m$ or feature density $q(\boldsymbol{z})$. Suppose that $\boldsymbol{x}_i$, $\boldsymbol{x}_j$, $\boldsymbol{z}$ denote the observed variables *w.r.t.* true distributions and $\hat{\boldsymbol{x}}_i$, $\hat{\boldsymbol{x}}_j$, $\hat{\boldsymbol{z}}$ denote the variables *w.r.t.* $\mathbf{\Phi}$, $\mathbf{\Theta}$-parameterized distributions. The upper bounds derived from $-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ can be interpreted in three aspects.

In the supervised case, training instances are m-tuples and for each domain-i empirical draw, it is able to search its corresponding domain-j empirical draw as the transformation groundtruth. In this scenario, $-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\boldsymbol{z})$ is bounded by the condition loss $\mathcal{L}^{\text{con}}_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i,\boldsymbol{x}_j)$ as below

Observation 1. *Given empirical draws from p_i ($\forall i \in [m]$), in supervised learning,*

$$\begin{aligned} &-I_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i;\boldsymbol{x}_j;\hat{\boldsymbol{z}}) \leq H_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) \\ &\leq \mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\left[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_j}(\hat{\boldsymbol{z}}|\boldsymbol{x}_j) d\hat{\boldsymbol{z}}\right] \triangleq \mathcal{L}^{\text{con}}_{\mathbf{\Phi},\mathbf{\Theta}}(\boldsymbol{x}_i,\boldsymbol{x}_j) \end{aligned} \tag{10}$$

where $p_{i,j} = p(\boldsymbol{x}_i,\boldsymbol{x}_j)$.

In Fig.2.a., we show how to build $\mathcal{L}^{\text{con}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. The loss can be implemented by l_1/l_2 norms.

In the unsupervised case, each empirical draw is separately given, therefore we have no access to $\boldsymbol{x}_j$. Distinct from (25), the MMI turns into $I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \hat{\boldsymbol{x}}_j; \hat{\boldsymbol{z}})$ where $\hat{\boldsymbol{x}}_j$ implies that domain-j samples are counterfeits and the bound constitutes a cross-domain cycle-consistency loss by means of $\hat{\boldsymbol{z}}$:

Observation 2. *Given empirical draws from p_i ($\forall i \in [m]$), in unsupervised learning,*

$$\begin{aligned} &- I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \hat{\boldsymbol{x}}_j; \hat{\boldsymbol{z}}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j) \\ \leq &\mathop{\mathbb{E}}_{\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j \sim p_{\boldsymbol{\theta}_j, \boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_j}(\hat{\boldsymbol{z}}|\boldsymbol{x}_j) d\hat{\boldsymbol{z}}\big] \triangleq \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j) \end{aligned} \tag{11}$$

where $p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i} = p(\boldsymbol{x}_i) \int_{\hat{\boldsymbol{z}}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i) d\hat{\boldsymbol{z}}$.

$\mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j)$ is constructed as illustrated in Fig.2.b.

The observations above presumed inputs drawn from the domain marginals $\{p_i\}_{i=1}^m$. If inputs are drawn from the feature distribution $q(\boldsymbol{z})$, $\hat{\boldsymbol{x}}_i$, $\hat{\boldsymbol{x}}_j$ would be generated from $\boldsymbol{z}$, and $-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_i; \hat{\boldsymbol{x}}_j; \boldsymbol{z})$ is upper bounded by the conditional entropies $H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)$ and $H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{x}}_i)$. They are equivalent to the cycle losses $\mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i)$ and $\mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j)$, which are revealed in Fig.2.c.

Observation 3. *Given empirical draws from $q(\boldsymbol{z})$,*

$$-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_i; \hat{\boldsymbol{x}}_j; \boldsymbol{z}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{x}}_i) \tag{12}$$

$$\begin{aligned} H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) &= \mathop{\mathbb{E}}_{\hat{\boldsymbol{x}}_i \sim p_{\boldsymbol{\theta}_i}, \boldsymbol{z} \sim q(\boldsymbol{z})} - \log q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) \triangleq \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i) \\ H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{x}}_i) &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{z} \sim q(\boldsymbol{z}) \\ \hat{\boldsymbol{x}}_i \sim p_{\theta_i}, \hat{\boldsymbol{x}}_j \sim p_{\theta_j}}} - \big[\log \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) d\boldsymbol{z}\big] \\ &\triangleq \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j) \end{aligned}$$

Associate Observations (1-3) and we impose cross-domain structure dependencies on $\boldsymbol{\Phi}$, $\boldsymbol{\Theta}$ by

$$\begin{aligned} \mathcal{R}_{\text{SL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) = \sum_{i,j \in [m], i \neq j} &\mathcal{L}^{\text{con}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i) \\ &+ \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j) \\ \mathcal{R}_{\text{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) = \sum_{i,j \in [m], i \neq j} &\mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j) + \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i) \\ &+ \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \hat{\boldsymbol{x}}_i, \hat{\boldsymbol{x}}_j) \end{aligned} \tag{13}$$

where $\mathcal{R}_{\text{SL}}$ / $\mathcal{R}_{\text{UL}}$ respectively regulate the supervised / unsupervised learning and upper bound (9). It implies that the minimization of $\mathcal{R}_{\text{SL}}$, $\mathcal{R}_{\text{UL}}$ equal to maximizing the MMIs. By Proposition.1, desire that adversarial learning (17) encourages $\{p_i\}_{i=1}^m$ and parameterized domain marginals agree with a high likelihood to domain variables $\big($*i.e.*, $\boldsymbol{x}_i = \hat{\boldsymbol{x}}_i$ in (39)$\big)$, then the minimization of $\mathcal{R}_{\text{SL}}$, $\mathcal{R}_{\text{UL}}$ leads to the joint distribution matching criterion (37),(38).

Theorem 1. *Suppose that true and parameterized domain marginal distributions maintain a high likelihood to domain variables, $\mathcal{R}_{\text{SL}} \to 0$ leads to the optima in (37); $\mathcal{R}_{\text{UL}} \to 0$ leads to the optima in (38).*

2.5 Adversarial Ensemble Learning

Learning m-ALI ensemble by (39) is able to capture the m-domain joint density. But it can be problematic as samples directly generated from $q(\boldsymbol{z})$ can be of low quality, *e.g.*, due to the poorly-efficient sampling in a high-dimensional feature space. To overcome this issue, we invent a *domain mixture adversarial ensemble (DMAE) loss* to refine (17) :

$$\begin{aligned} \mathcal{L}^{(i)}_{\text{DMAE}}(\boldsymbol{\Phi}, \boldsymbol{\Theta}, \boldsymbol{\Omega}) = &\mathbb{E}_{\boldsymbol{x}_i, \hat{\boldsymbol{z}} \sim q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \hat{\boldsymbol{z}})} \big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i, \hat{\boldsymbol{z}})\big] \\ &+ \sum_{j=1}^m \pi_j \Big(\mathbb{E}_{\hat{\boldsymbol{x}}_i \sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}), \boldsymbol{z} \sim q_{\boldsymbol{\phi}_j}} \big[\log \big(1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i, \boldsymbol{z})\big)\big]\Big) \end{aligned} \tag{14}$$

where $\sum_{j=1}^{m} \pi_j = 1$ indicates the proportion of the domain mixture for adversary. Compared with (17) whose fake samples are solely generated from $q(\boldsymbol{z})$, $\mathcal{L}^{(i)}_{\mathrm{DMAE}}(\boldsymbol{\Phi}, \boldsymbol{\Theta}, \boldsymbol{\Omega})$ consider fake samples generated from the domain-encoded features, which are derived from the real samples that belong to the other domains, *i.e.*, $\boldsymbol{z} \sim \int q_{\boldsymbol{\phi}_j}(\boldsymbol{z}, \boldsymbol{x}_j) d\boldsymbol{x}_j$ ($\forall j \in [m]$). These fake samples converted from different domains are unified into the DMAE loss (14) to cheat the domain-i critic net $f_{\boldsymbol{\omega}_i}$. It can be provably verified that, the adversarial ensemble learning retains the theoretical property of (17):

Proposition 2. *The optimum of the generation, inference and critic networks in*

$$\min_{\boldsymbol{\Theta}, \boldsymbol{\Phi}} \max_{\boldsymbol{\Omega}} (1-\gamma) \sum_{i=1}^{m} \mathcal{L}^{(i)}_{\mathrm{ALI}} + \gamma \sum_{i=1}^{m} \mathcal{L}^{(i)}_{\mathrm{DMAE}} \tag{15}$$

refer to their saddle points in Lemma.2 if and only if $\forall i \in [m]$*, there exist* $p_{\boldsymbol{\theta}^*_i}(\boldsymbol{x}|\boldsymbol{z}) q(\boldsymbol{z}) = q_{\boldsymbol{\phi}^*_i}(\boldsymbol{z}|\boldsymbol{x}) p(\boldsymbol{x})$.

where γ denotes the trade-off between (17) and DAME loss. Proposition.4 demonstrates that, even if we change the learning objective (17), Lemma.2 and the other analysis based on (17) can be completely followed by the new objective (62).

Combining (39) and (62), we formalize MMI-ALI as

$$\min_{\boldsymbol{\Theta}, \boldsymbol{\Phi}} \max_{\boldsymbol{\Omega}} (1-\gamma) \sum_{i=1}^{m} \mathcal{L}^{(i)}_{\mathrm{ALI}} + \gamma \sum_{i=1}^{m} \mathcal{L}^{(i)}_{\mathrm{DMAE}} + \beta \, \mathcal{R}_{\mathrm{SL}} / \mathcal{R}_{\mathrm{UL}} \tag{16}$$

where $\mathcal{R}_{\mathrm{SL}}/\mathcal{R}_{\mathrm{UL}}$ are switched by supervised/unsupervised learning and $\beta > 0$ denotes the loss-balance factor. Normally, we set $\beta = 1$ in our implementation.

3 Experiments

In this section, we propose diverse cross-m-domain experiments to evaluate our MMI-ALI in generative modeling and show the primal empirical results. More experiments (*e.g.*, ablation) and visualization are founded in Appendix.B.

3.1 Balance between efficacy and scalability

Compared with existing methods, MMI-ALI strikes a right balance between model capacity and scalability. To highlight this merit, we design the first experiment on synthetic data domains with m ranged in $2 \sim 6$. We choose $q(\boldsymbol{z})$ as an isotropic Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$, then each density in $\{p_i\}_{i=1}^{m}$ is a 2D Gaussian Mixture Model (GMM) with 5 components $\mathcal{N}(\mathbf{0}, 0.2\mathbf{I})$. (As illustrated in Fig.4) Due to the simplicity of synthetic data, we only consider unsupervised learning across them. We evaluate MMI-ALI and its parameter-shared version termed "MMI-ALI (PS)", with CycleGAN and StarGAN. All of them are trained on 2048 with vanilla GAN loss and tested on 1024 examples drawn from each of $\{p_i\}_{i=1}^{m}$. For a fair comparison, all baselines use two-layered fully-connected nets with ReLU to generate data and make critics. l_2-norm is chosen as the cycle-consistency loss for all baseline during training.

Evaluation. Two measures have been introduced. The first is *geometric score* (GS) [27] that evaluates generation quality by comparing the topological properties of the supports behind the generated and true domain marginals. The other is *mean squared error* (MSE) broadly used to measure the conditional density modeling via sample reconstruction quality across domains. Each baseline is performed in average of $m(m-1)$ transfer cases on two measures to thoroughly reflect the learned joint distribution. The results and parameters are shown in Fig.4.(a-b) and (c), respectively. Note that, StarGAN uses a domain-shared pipeline so that its parameter scale is almost consistent as m increases. However, StarGAN's GS, MSE heavily suffer even in toy domains, due to its intrinsic vulnerability as we have discussed. Particularly, when there exists an overlap across domains, the examples drawn from the overlap (or close to the overlap) can belong to all of these domains. This phenomena is general (see our empirical results in real data) and StarGANs can do nothing to help. On the other hand, MMI-ALI and CycleGAN are close in GS and MSE, yet CycleGAN requires exponentially-increasing parameters. They demonstrate that MIM-ALIs remain convincing performances as they scale to the scenarios with more domains. We show more visualization results in SM.

3.2 Geometry-varying m domains.

Geometry-varying information is difficult to capture in generative modeling [28]. Based on this challenge, our second experiment considers cross-m-domain generation where the m-domain samples

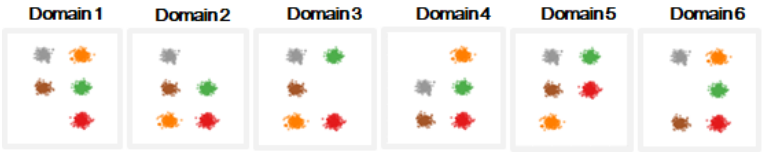


Figure 3: Synthetic domains used in our first experiments. As m increases, they are proceedingly incorporated for multi-domain joint distribution leanring from left to right.

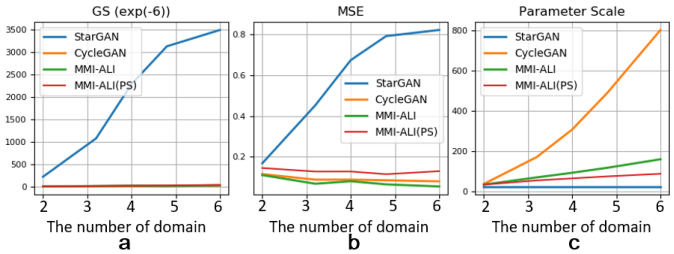


Figure 4: Transfer evaluations with 2∼6 synthetic domains: (a). Geometric Score (GS, lower is better); (b). Mean Square Error (MSE, lower is better); (c). Parameter Scale (lower is better).

present significant variation in geometry. We evaluate whether this information can be captured by MMI-ALI and the other baselines.

Table 1: SSIM of StarGAN (ST), CycleGAN (CG) and MMI-ALI(MA) in supervised cross-domain generation case.

	1%	5%	10%
ST	0.00	0.00	0.00
CG	0.32	0.31	0.35
MA	**0.57**	**0.68**	**0.72**

Table 2: IS of StarGAN (ST), CycleGAN (CG) and MMI-ALI(MA) in unsupervised cross-domain generation case.

	$-\frac{\pi}{2}\rightarrow 0$	$\frac{\pi}{2}\rightarrow 0$	$0\rightarrow\frac{\pi}{2}$	$-\frac{\pi}{2}\rightarrow\frac{\pi}{2}$	$-\frac{\pi}{2}\rightarrow 0$	$\frac{\pi}{2}\rightarrow-\frac{\pi}{2}$
ST	1.00	1.00	1.00	1.00	1.00	1.00
CG	8.34	6.13	2.25	2.38	1.71	1.04
Ours	8.99	9.01	2.95	3.86	3.31	3.08

Specifically, we choose MNIST as the base domain, then rotate the images by $-\frac{\pi}{2}^\circ$, $\frac{\pi}{2}^\circ$ to create two other domains. Then MMI-ALI, CycleGAN and StarGAN are demanded to learn pattern transfer across the three domains in supervised and unsupervised learning setups. In supervised setup, data present as triplets so that each example from one domain has its corresponding groundturth in other domains. This information is not provided in unsupervised cases. In supervised case, we compare (supervised) MMI-ALI with CycleGAN and StarGAN augmented with condition loss used by c-GAN. In unsupervised case, we compare (unsupervised) MMI-ALI with ordinary CycleGAN and StarGAN. For a fair comparison, we standardize backbone behind the baselines in DCGAN [1], and they are trained with vanilla GAN and l_1-norm cycle losses.

Evaluation. In supervised learning setup, we measure transformed results by Structured SIMilarity (SSIM) [29]. The visualization and quantitative results are shown in Fig.4 and Table.6, respectively. MMI-ALI is the *only baseline* that can produce all transfer patterns. StarGAN collapses during training and create nothing for transfer. CycleGAN performs better than MMI-ALI in $0\rightarrow-\frac{\pi}{2}, \frac{\pi}{2}$, however, fails in capturing larger rotation (*e.g.*, $-\frac{\pi}{2}\rightarrow\frac{\pi}{2}$). It demonstrates a weakness of CycleGAN, which merely learns a pairwise joint distribution per time. In other word, it can not leverage m-domain knowledge to enhance the cross-domain generation performance. MMI-ALI avert this issue due to modeling m-domain joint distribution by ensemble. For more concrete evaluation, we provide different proportion of supervised data, *i.e.*, 1%, 5%, 10%, to check how much the model can benefit from supervision. We find that in 3-domain Rotated MNIST, cross-domain alignment can not significantly help StarGAN and CycleGAN to improve their joint distribution learning performance. But MMI-ALI can benefit from small amount of supervision. Cross-domain digit transformation conceives structure variation, thus, the patterns are difficult to capture without supervisions. This statement is verified in unsupervised results shown in Fig.4. Even so, our MMI-ALI is still powerful in

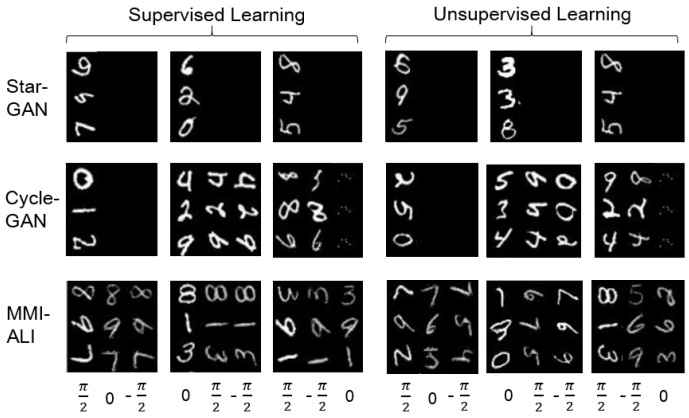


Figure 5: Cross-3-domain generation performed by StarGAN, CycleGAN and MMI-ALI (ours) in supervised and unsupervised learning setups. ***For each sub-picture, the left column indicates inputs*** and the rest indicate the cross-domain transformed results.

generative modeling. To be specific, we evaluate the unsupervised generation by *Inception Score* [30]. MMI-ALI consistently outperform the other baselines across 6 cross-domain generation scenarios.

3.3 Cross-m-domain visual style transfer.

In this experiment, we consider 3-domain object transfiguration and 3-heterogeneous-domain style transfer.

In object transfiguration, evaluated DGMs are required to transform a specific part of an object to some target pattern whereas the other parts remain the same. One example is to translate a sort of animals (*e.g.*, 1000 classes in ImageNET) to become another kind with visual similarity. In our experiment, we consider the 3-object transfiguration in $\mathrm{Zebra} \leftrightarrow \mathrm{Horse} \leftrightarrow \mathrm{Okapi}$, where Zebra and Horse share their shapes while differ from the strip; then Okapi is *"zebra- striped" on its legs with a "horse-like" torso*. The experiment is conducted by reconfiguring the state-of-the-art residual-block-based [31] CycleGAN into MMI-ALI. For a fair comparison with CycleGAN, we depart the generator of CycleGAN as a pair of inference and generation net for our MMI-ALI, and follow the identical training tricks. Instead of using a non-informative prior, we apply $\boldsymbol{z} = \mu(\boldsymbol{z}) + \epsilon$ to provide features. As for StarGAN, we employ the official code reported in their original paper where their models are also built on ResNet.

In 3-heterogeneous-domain transfer, we consider Cityscape [32] as the base benchmark, then employ the real data and their segmentation labels to construct two domains (R and Seg). We further applied the pretrained sketch detector [33] to generate the third domain (Ske). To this we are able to evaluate all baselines in unsupervised and supervised learning manners (Condition loss is used in the supervised case). We resemble the similar configuration and training strategy in object transfiguration.

Evaluation. Amazon Mechanical Turk (AMT) is employed to evaluate the object transfiguration experiment. We follow the perceptual evaluation from [34], where workers are provided with a pair of generated image (ours and the other baseline), and given unlimited time to select the one more likely as a target domain image. In Cityscape, we take *Frechet Inception Distance* (FID)[35] and MSE as the metrics (MSE deferred in SM).

Table 3: Pairwise comparison of MMI-ALI with other baselines. Chance is at 50%. Each cell indicates the percentage where our result is preferred over the other method. MMI-ALI overwhelmingly outperforms StarGAN and stay ahead of CycleGAN.

	Okapi2Zebra	Okapi2Horse	Zebra2Okapi	Horse2Okapi
StarGAN	100.0%	100.0%	97.6%	100.0%
CycleGAN	57.2%	52.1%	56.5%	67.2%

The visualization of object transfiguration are illustrated in Fig.6. First of all, StarGAN takes a mild effect. Due to the its category-generative pipeline, cross-domain style knowledge is hardly disentangled and thus, drives the produced images lack of fidelity in details. In a comparison,

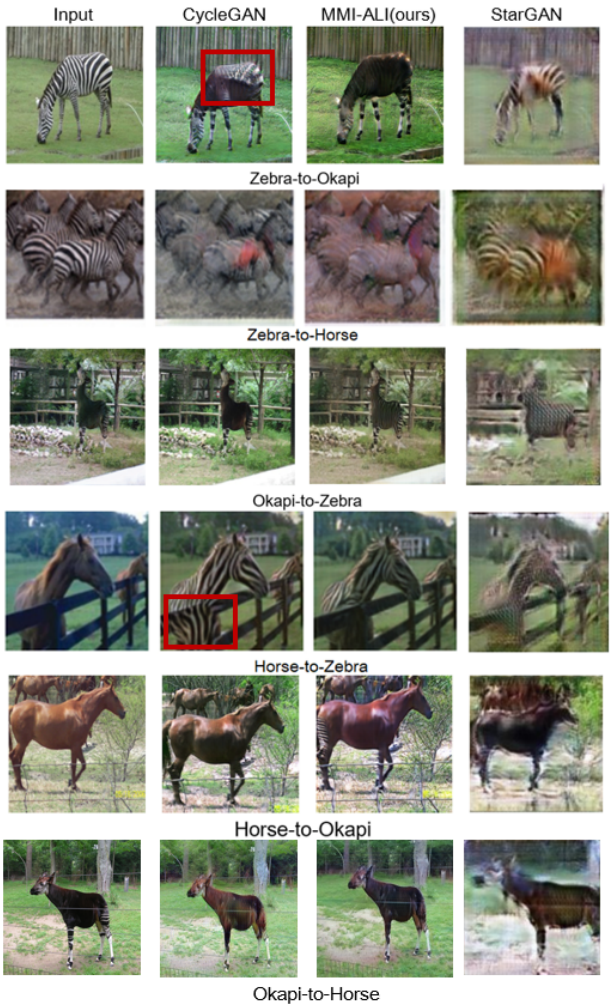


Figure 6: Style transfer on Zebra&Horse&Okapi.

CycleGAN performs so aggressive that some details in the original images have been undesirably modified (Such negative effect is highlighted in red boxes). MMI-ALI successfully avoids the problem CycleGAN and StarGAN suffer from. Table.7 shows the consistent quantitative results.

Table 4: FID in cross-3-domain transfer in Cityscape

		R→Seg	Seg→R	R→Ske	Ske→R	Seg→Ske	Ske→Seg
Unsuper	ST	405.16	372.59	385.08	388.97	357.19	417.39
	CG	224.04	**213.43**	164.65	**222.24**	**60.20**	**144.07**
	Ours	**202.93**	254.41	**150.98**	246.04	101.30	192.13
Super	ST	382.90	440.53	419.11	383.72	400.70	299.82
	CG	**217.28**	260.41	**171.04**	**223.43**	65.18	228.61
	Ours	250.48	**246.01**	196.06	229.45	**55.76**	**143.20**

In Cityscape, MMI-ALI achieved the leg-and-leg performances with CycleGAN in FID in supervised and unsupervised learning (Table 8). But CycleGAN gets less benefits from supervision. They significantly outperformed StarGAN. As observed in Fig 7 8, when MMI-ALI is compared with the target generation groundtruth, it has achieved superior transfers so that avoided modeling C_m^2 generators.

3.4 Cross-m-emotion text style transfer.

In final experiment, we conduct a emotion style transfer in a text semantic embedding space. Specifically, we employ MojiTalk dataset [36] that contains 64 emojis, and we collect a part of them to construct 4 domains related to 'Happy' (40000 entries), 'Angry' (29000 entries), 'Pensive' (14000 entries) and 'Abash' (6261 entries), respectively. In this scenario, the goal of MMI-ALI is to transform

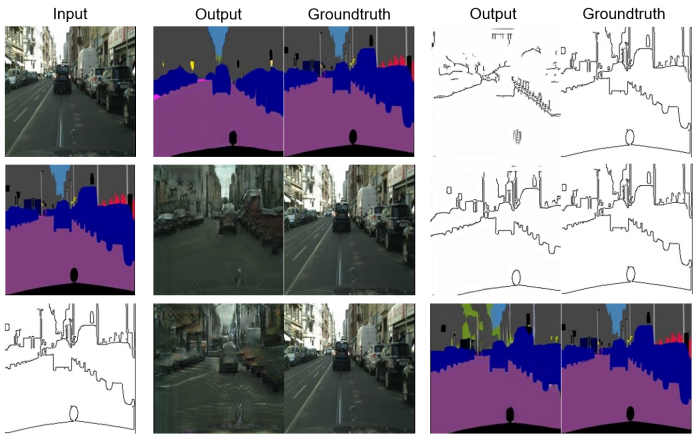


Figure 7: Cross-3-domain supervised transfer in Cityscape.

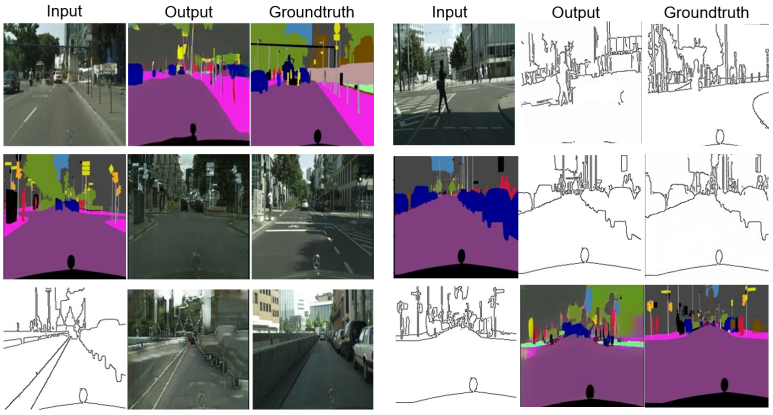


Figure 8: Cross-3-domain unsupervised transfer in Cityscape.

the emotional text embeddings (we choose skip-thought [37] as our language model to extract the representation of each text in the domains) from one domain to the others.

Evaluation. Due to the embedding space is substantially discrete, the aforementioned metrics are not appropriate to evaluate the transfer efficiency. In this way, we employ a famous MRR (Mean Reciprocal Rank, [38]), to measure the emotion transfer quality. For instance, when MMI-ALI transfer "happy" into "angry", we sort all sentences' embeddings based on their cosine distance to the embeddings generated from MMI-ALI. Then we calculate the rank of the nearest "angry" embedding and use its average of all transfer score. We use a simple fully-connected network with ReLU as the base backbone of MMI-ALI and train it with Batch normalization (BN). We compare MMI-ALI with the no-adaptation groundtruth results and the state-of-the-art unaligned text style transfer model [5] that trained by the official code .The results are shown in Table.5. We provide more visualization by retrieving the nearest neighbor of each target domain, for the embeddings before (*no adaptation*) and after MMI-ALI transform (Fig.9). As can be observed, the transferred embeddings (outputs of MMI-ALI) leads to the neighbor embeddings with the texts containing more significant emotion.

Table 5: MRR for each domain transfer evaluation. Higher is better. As can be seen, MRRs in "Happy" and "Abush" are even higher than the original domain, indicating the effectiness of MMI-ALI.

	Happy	Angry	Pensive	Abash
groundtruth	0.71	0.41	0.53	0.21
[33]	0.52	0.17	0.31	0.07
MMI-ALI	1.0	0.40	0.27	0.24

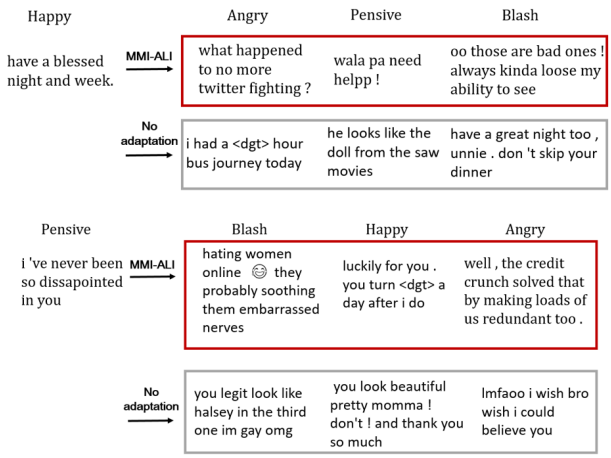


Figure 9: The illustration of emotion style transfer in skipthough embedding space. We compare our MMI-ALI with no adaptation.

4 Conclusion

In this paper, we have delved into the problem of multiple domain joint distribution matching that summarized a variety of cross-domain generation tasks. Instead of hacking a complex DGM pipeline, we propose MMI-ALI, which reshapes classical ALI from the perspective of model integration and is linearly-scalable with the domain number. It learns with an adversarial ensemble loss and can be applied in both supervised and unsupervised learning schemes. Extensive evaluation results on diverse m-domain scenarios have demonstrated the superiority of the proposed framework to the existing DGMs feasible for cross-m-domain generation, e.g., CycleGAN and Star-GAN.

Appendix.A

Lemma 2 ([1]). *The domain-i generation, inference and critic nets w.r.t.,$\{\boldsymbol{\theta}_i^*, \boldsymbol{\phi}_i^*, \boldsymbol{\omega}_i^*\}$ $(\forall i \in [m])$ refer to a saddle point in*

$$\begin{aligned}\min_{\boldsymbol{\theta_i},\boldsymbol{\phi}_i}\max_{\boldsymbol{\omega_i}} \mathcal{L}_{\text{ALI}}^{(i)}(\boldsymbol{\theta}_i,\boldsymbol{\phi}_i,\boldsymbol{\omega}_i) = \mathbb{E}_{\boldsymbol{x}_i\sim p(\boldsymbol{x}_i),\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i)}\big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] \\ + \mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),z\sim q(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})]\end{aligned} \tag{17}$$

$\iff p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i|\boldsymbol{z})q(\boldsymbol{z}) = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i)p_i(\boldsymbol{x}_i)$, i.e., $\int p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i|\boldsymbol{z})q(\boldsymbol{z})d\boldsymbol{z} = p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i) = p_i(\boldsymbol{x}_i)$ *and* $\int q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i)p_i(\boldsymbol{x}_i)d\boldsymbol{x}_i = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}) = q(\boldsymbol{z})$.

Proposition 3. *Given a pair of domains $\forall i, j \in [m]$, $i \neq j$, their well-trained ALIs (in Lemma.2) construct a cross-domain transfer process $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)$ that satisfies*

$$p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j) = \int_{\boldsymbol{x}_i} p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)p_i(\boldsymbol{x}_i)d\boldsymbol{x}_i = p_i(\hat{\boldsymbol{x}}_j) \tag{18}$$

Proof. First, we obtain $p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i)$ by marginalizing out all random variables in

$$p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i) = \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z} \tag{19}$$

except for those with respect to domain i and j:

$$\begin{aligned} p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i) &= \int p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) \prod_{k\in[m],k\neq j,k\neq i} p_{\boldsymbol{\theta}_k^*}(\hat{\boldsymbol{x}}_k|\boldsymbol{z})q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i) \prod_{k\in[m],k\neq j,k\neq i} d\hat{\boldsymbol{x}}_k d\boldsymbol{z} \\ &= \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) \prod_{k\in[m],k\neq j,k\neq i} \Big(\int_{\hat{\boldsymbol{x}}_k} \frac{p_{\boldsymbol{\theta}_k^*}(\hat{\boldsymbol{x}}_k|\boldsymbol{z})q(\boldsymbol{z})}{q(\boldsymbol{z})} d\hat{\boldsymbol{x}}_k\Big) q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z} \\ &= \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) \prod_{k\in[m],k\neq j,k\neq i} \Big(\frac{\int_{\hat{\boldsymbol{x}}_k} p_{\boldsymbol{\theta}_k^*}(\hat{\boldsymbol{x}}_k,\boldsymbol{z})d\hat{\boldsymbol{x}}_k}{q(\boldsymbol{z})}\Big) q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z} \end{aligned} \tag{20}$$

Since $\forall k \in [m]$,

$$\int_{\hat{\boldsymbol{x}}_k} p_{\boldsymbol{\theta}_k^*}(\hat{\boldsymbol{x}}_k, \boldsymbol{z}) d\hat{\boldsymbol{x}}_k = \int_{\boldsymbol{x}_k} p_{\boldsymbol{\theta}_k^*}(\boldsymbol{x}_k, \boldsymbol{z}) d\boldsymbol{x}_k = q(\boldsymbol{z}) \tag{21}$$

, it holds

$$p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i) = \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i) d\boldsymbol{z} \tag{22}$$

and

$$p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i) p_i(\boldsymbol{x}_i) = p_i(\boldsymbol{x}_i) \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i) d\boldsymbol{z} \tag{23}$$

. Marginalize out $\boldsymbol{x}_i$,

$$\begin{aligned} p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j) &= p_i(\boldsymbol{x}_i) \int_{\boldsymbol{x}_i} p_{\boldsymbol{\Phi}^*,\boldsymbol{\Theta}^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{x}_i) d\boldsymbol{x}_i \\ &= \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) \Big(\int_{\boldsymbol{x}_i} q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x}_i) p_i(\boldsymbol{x}_i) d\boldsymbol{x}_i \Big) d\boldsymbol{z} \\ &= \int p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}) d\boldsymbol{z} = p_{\boldsymbol{\theta}_j^*}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q(\boldsymbol{z}) d\boldsymbol{z} = p_j(\hat{\boldsymbol{x}}_j) \end{aligned} \tag{24}$$

where the equalities are induced by Lemma.2. □

Observation 4. *Given empirical draws from p_i ($\forall i \in [m]$), in supervised learning,*

$$\begin{aligned} &- I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) \\ \leq &\mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j \sim p_{i,j}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_j}(\hat{\boldsymbol{z}}|\boldsymbol{x}_j) d\hat{\boldsymbol{z}} \big] \triangleq \mathcal{L}^{\text{con}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) \end{aligned} \tag{25}$$

where $p_{i,j} = p(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

Observation 5. *Given empirical draws from p_i ($\forall i \in [m]$), in unsupervised learning,*

$$\begin{aligned} &- I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \hat{\boldsymbol{x}}_j; \hat{\boldsymbol{z}}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j) \\ \leq &\mathop{\mathbb{E}}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j \sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_j}(\hat{\boldsymbol{z}}|\boldsymbol{x}_j) d\hat{\boldsymbol{z}} \big] \triangleq \mathcal{L}^{\text{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \hat{\boldsymbol{x}}_j) \end{aligned} \tag{26}$$

where $p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i} = p_i(\boldsymbol{x}_i) \int_{\hat{\boldsymbol{z}}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{z}}) q_{\boldsymbol{\phi}_i}(\hat{\boldsymbol{z}}|\boldsymbol{x}_i) d\hat{\boldsymbol{z}}$.

The two observations are proved by a similar method and we summarize it as follows:

Proof. According to the definition of MI and MMI, $-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}})$ can be factorized as

$$\begin{aligned} -I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}}) &= -I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j) + I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j|\hat{\boldsymbol{z}}) \\ &= H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) - H(\boldsymbol{x}_i) + I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j|\hat{\boldsymbol{z}}) \end{aligned} \tag{27}$$

. Since examples in $I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}})$ comes from domain i, q_i is the input distribution and un-relevant with parameters $\boldsymbol{\Phi}$, $\boldsymbol{\Theta}$. Therefore $H(\boldsymbol{x}_i)$ is constant and we can ignore its effect to learning:

$$-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) + I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j|\hat{\boldsymbol{z}}) \tag{28}$$

. Since given $\hat{\boldsymbol{z}}$, random variables $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are independent, so $I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}}) = 0$ and

$$\begin{aligned} -I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i; \boldsymbol{x}_j; \hat{\boldsymbol{z}}) &\leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) \\ &= \mathbb{E}_{\boldsymbol{x}_i,\boldsymbol{x}_j \sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z} \big] \end{aligned} \tag{29}$$

. Based on the definition of $\boldsymbol{\phi}_i$ from the preliminary in our paper, $p_{\boldsymbol{\phi}_i}$ indicates a network that takes $\boldsymbol{x}_i$ as an input to infer feature $\boldsymbol{z}$, so the condition density $p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z})$ is intractable. Similarly, $q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j)$

is also intractable. In this case, we would like to upper bound (12) by following the technique in variational information maximization [39]:

$$\begin{aligned}
H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) &= \mathbb{E}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big] \\
&= \mathbb{E}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \frac{\int p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}}{\int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}}\big] - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big] \\
&= -\mathrm{KL}(\int p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z} || \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}) \\
&\qquad + \mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big] \\
&\leq \mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big]
\end{aligned} \tag{30}$$

In supervised learning, $p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$ is known so that $\forall i, j \in [m]$ and $j \neq i$, $p(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is also known. In this way, for each empirical draw $\boldsymbol{x}_i \sim p_i$, we actually obtain an empirical pair $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Note that when $p(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is given, there is an approximation of $p_{\theta_j,\phi_i} \to p_{i,j}$. So it holds

$$\begin{aligned}
H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\boldsymbol{x}_j) &\leq \mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big] \\
&\approx \mathbb{E}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} - \big[\log \int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big]
\end{aligned} \tag{31}$$

In unsupervised learning, $p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$ is not observed so that for each empirical draw $\boldsymbol{x}_i \sim p_i$, we are only provided fake item $\hat{\boldsymbol{x}}_j$. Under the analysis above, we obtain

$$H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\hat{\boldsymbol{x}}_j) \leq \mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j) d\boldsymbol{z}\big] \tag{32}$$

It is important to note that, when $\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{x}}_j\sim p_{\boldsymbol{\theta}_j,\boldsymbol{\phi}_i}} - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}\big] \to 0$, it holds $\mathrm{KL}(\int p_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z} || \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}) q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j) d\boldsymbol{z}) \to 0$. So the bounds in (31), (32) are tight. □

Observation 6. *Given empirical draws from $q(\boldsymbol{z})$,*

$$-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_i; \hat{\boldsymbol{x}}_j; \boldsymbol{z}) \leq H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{x}}_i) \tag{33}$$

where

$$\begin{aligned}
H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) &= \mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i},\boldsymbol{z}\sim q(\boldsymbol{z})} - \big[\log q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)\big] \\
H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_j|\hat{\boldsymbol{x}}_i) &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}\sim q(\boldsymbol{z}) \\ \hat{\boldsymbol{x}}_i\sim p_{\theta_i},\hat{\boldsymbol{x}}_j\sim p_{\theta_j}}} - \big[\log \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z}) q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) d\boldsymbol{z}\big]
\end{aligned} \tag{34}$$

Proof. Obviously,

$$\begin{aligned}
-I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_i; \hat{\boldsymbol{x}}_j; \boldsymbol{z}) &= -H(\boldsymbol{z}) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + I_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}; \hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j) \\
&= -H(\boldsymbol{z}) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\mathrm{KL}\Big(\frac{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}; \hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j)}{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j) p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j)}\Big) \\
&= -H(\boldsymbol{z}) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z})}\mathrm{KL}\Big(\frac{p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}) p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j)}{p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j) p_{\boldsymbol{\phi}_j,\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j)}\Big) \\
&= -H(\boldsymbol{z}) + H_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i) + \mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}),p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j),p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}\big[\log \frac{p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}{p_{\boldsymbol{\phi}_j,\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j)}\big]
\end{aligned} \tag{35}$$

. Observe that given intractable $p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j)$, it holds $q(\boldsymbol{z})p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j) = p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j, \boldsymbol{z}) = p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z})q(\boldsymbol{z})$. So there is

$$
\begin{aligned}
&\mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}), p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j), p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}\big[\log \frac{p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}{p_{\boldsymbol{\phi}_j,\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\hat{\boldsymbol{x}}_j)}\big] \\
=&\mathbb{E}_{\boldsymbol{z}\sim q(\boldsymbol{z}), p_{\boldsymbol{\theta}_j}(\boldsymbol{z}|\hat{\boldsymbol{x}}_j), p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}\big[\log \frac{p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})}{\int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)}\big] \\
=&\underbrace{-\mathbb{E}_{p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j,\boldsymbol{z})}\mathrm{CrossEntropy}\Big(p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z})\Big)}_{\leq 0} + \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}\sim q(\boldsymbol{z}) \\ \hat{\boldsymbol{x}}_i\sim p_{\theta_i}, \hat{\boldsymbol{x}}_j\sim p_{\theta_j}}} - \big[\log \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\hat{\boldsymbol{x}}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\hat{\boldsymbol{x}}_i)d\boldsymbol{z}\big]
\end{aligned}
\tag{36}
$$

. Combine (35), (36) and it leads to the observation. □

Theorem 2. *Suppose that true and parameterized domain marginal distributions maintain a high likelihood, $\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) \to 0$ leads to the optima of*

$$
\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} -\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_i\}_{i=1}^m)\big] \tag{37}
$$

where $p = p(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$; $\mathcal{R}_{\mathrm{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) \to 0$ leads to

$$
H\big(\boldsymbol{x}_i|\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i}\big) = -\mathbb{E}_{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}}\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\hat{\boldsymbol{x}}_j\}_{j\in[m]\&j\neq i})\big] \tag{38}
$$

where $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}} = p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m)$

Proof. As true and parameterized domain marginal distributions maintain a high likelihood, it would be appropriate to unify the domain-specific random variables by the same remarks, *i.e.*, $\forall i \in [m]$, $\hat{\boldsymbol{x}}_i \to \boldsymbol{x}_i$. Hence we frame $\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta}, \bar{\boldsymbol{\Phi}})$ and $\mathcal{R}_{\mathrm{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi})$ as

$$
\begin{aligned}
\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) &= \sum_{i,j\in[m],i\neq j} \mathcal{L}^{\mathrm{con}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i) + \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i, \boldsymbol{x}_j) \\
\mathcal{R}_{\mathrm{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) &= \sum_{i,j\in[m],i\neq j} \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i) + \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i, \boldsymbol{x}_j)
\end{aligned}
\tag{39}
$$

to ease our analysis. Besides, since each term in $\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi})$ and $\mathcal{R}_{\mathrm{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi})$ is non-negative, so $\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) \to 0$ if and only if

$$
\begin{aligned}
&\forall i, j \in [m], j \neq i, \\
&\qquad \mathcal{L}^{\mathrm{con}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) \to 0,\ \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i) \to 0,\ \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i, \boldsymbol{x}_j) \to 0
\end{aligned}
\tag{40}
$$

, namely,

$$
\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big] &\approx 0 \\
\mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i}, \boldsymbol{z}\sim q(\boldsymbol{z})} -\big[\log q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)\big] &\approx 0 \\
\mathop{\mathbb{E}}_{\boldsymbol{x}_i\sim p_{\theta_i}, \boldsymbol{x}_j\sim p_{\theta_j}} -\big[\log \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] &\approx 0
\end{aligned}
\tag{41}
$$

. Similarly, $\mathcal{R}_{\mathrm{UL}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}) \to 0$ if and only if

$$
\begin{aligned}
&\forall i, j \in [m], j \neq i, \\
&\qquad \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i, \boldsymbol{x}_j) \to 0,\ \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i) \to 0,\ \mathcal{L}^{\mathrm{cycle}}_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{z}, \boldsymbol{x}_i, \boldsymbol{x}_j) \to 0
\end{aligned}
\tag{42}
$$

, namely,

$$
\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big] &\approx 0 \\
\mathbb{E}_{\boldsymbol{x}_i\sim p_{\boldsymbol{\theta}_i}, \boldsymbol{z}\sim q(\boldsymbol{z})} -\big[\log q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)\big] &\approx 0 \\
\mathop{\mathbb{E}}_{\boldsymbol{x}_i\sim p_{\theta_i}, \boldsymbol{x}_j\sim p_{\theta_j}} -\big[\log \int_{\boldsymbol{z}} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] &\approx 0
\end{aligned}
\tag{43}
$$

. Here we start our proof based on the observations in (41),(43).

First, we rewrite $-\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_i\}_{i=1}^m)\big]$, *i.e.*,

$$\begin{aligned}-\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_i\}_{i=1}^m)\big] &= -\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_j\}_{j\in[m],j\neq i}|\boldsymbol{x}_i)p(\boldsymbol{x}_i)\big]\\ &= -\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_j\}_{j\in[m],j\neq i}|\boldsymbol{x}_i)\big] + H(\boldsymbol{x}_i)\\ &= -\mathbb{E}_p\big[\log \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] + H(\boldsymbol{x}_i)\end{aligned} \tag{44}$$

where $H(\boldsymbol{x}_i)$ is constant. Hence $\min\limits_{\boldsymbol{\Phi},\boldsymbol{\Theta}} -\mathbb{E}_p\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_i\}_{i=1}^m)\big]$ if and only if

$$\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} \ -\mathbb{E}_p\big[\log \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] \tag{45}$$

. Observe that,

$$\begin{aligned}&\mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big]\\ =&\int_{\mathcal{X}_i\times\mathcal{X}_j} p(\boldsymbol{x}_i,\boldsymbol{x}_j) - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big]d\boldsymbol{x}_i d\boldsymbol{x}_j\\ =&\int_{\mathcal{X}} p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m) - \big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big]\prod_{i=1}^m d\boldsymbol{x}_i\end{aligned} \tag{46}$$

where $\mathcal{X}_i$ indicates the support of p_i and $\mathcal{X} = \mathcal{X}_1\times\cdots\times\mathcal{X}_m$ indicates the product space of the supports of $\{p_i\}_{i=1}^m$. In this way,

$$\begin{aligned}&\sum_{j\in[m],j\neq i}\ \mathop{\mathbb{E}}_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\big[\log \int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]\\ =&\sum_{j\in[m],j\neq i}\int_{\mathcal{X}} p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m)\Big(-\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big]\Big)\prod_{k=1}^m d\boldsymbol{x}_k\\ =&\int_{\mathcal{X}} p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m)\Bigg(\sum_{j\in[m],j\neq i} -\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big]\Bigg)\prod_{k=1}^m d\boldsymbol{x}_k\\ =&\int_{\mathcal{X}} p(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_m)\Bigg(-\big[\log \prod_{j\in[m],j\neq i}\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big]\Bigg)\prod_{k=1}^m d\boldsymbol{x}_k\\ =&-\mathbb{E}_p\big[\log \prod_{j\in[m],j\neq i}\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big]\end{aligned} \tag{47}$$

. Based on (45) and (47), we would like to prove that $\forall i,j\in[m]$, $i\neq j$,

$$-\mathbb{E}_p\big[\log \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] \leq -\mathbb{E}_p\big[\log \prod_{j\in[m],j\neq i}\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big] \tag{48}$$

. If the aforementioned inequality is satisfied, $\mathop{\mathbb{E}}\limits_{\boldsymbol{x}_i,\boldsymbol{x}_j\sim p_{i,j}} -\big[\log \int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_j}(\boldsymbol{z}|\boldsymbol{x}_j)d\boldsymbol{z}\big] \to 0$ would lead to the right side of (48) approach to 0 $\big($the result of (47)$\big)$ and thus, bounds the non-negative left side approach to 0. Based on (41) we observe that given $\forall(\boldsymbol{x}_i,\boldsymbol{x}_j)\sim p(\boldsymbol{x}_i,\boldsymbol{x}_j)$, as functions *w.r.t.* $\boldsymbol{z}$, $p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\cdot)$ and $p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\cdot)$ simultaneously obtain high likelihood about $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ (vice versa). By Chebyshev's algebraic inequality [40] we obtain

$$\begin{aligned}&\int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z}\\ &\qquad\geq \Big(\int p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z}\Big)\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z}\Big)\end{aligned} \tag{49}$$

where $\{i,j,k\} \subseteq [m]$. Follow the similar routine and then we can induce

$$\begin{aligned}\int \prod_{i\in I\subseteq[m]/\{k,j\}} & p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z} \\ &\geq \Big(\int \prod_{i\in I\subseteq[m]/\{k,j\}} p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z}\Big)\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_k}(\boldsymbol{z}|\boldsymbol{x}_k)d\boldsymbol{z}\Big)\end{aligned} \tag{50}$$

where I is arbitrary identity subset in $[m]/\{j,k\}$. Factorize $\int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}$ by the aid of (50) and there is

$$\begin{aligned}\int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z} &= \int \prod_{j\in[m-1],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})p_{\boldsymbol{\theta}_m}(\boldsymbol{x}_m|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z} \\ &\geq \Big(\int \prod_{j\in[m-1],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\Big(\int p_{\boldsymbol{\theta}_m}(\boldsymbol{x}_m|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big) \\ &\geq \Big(\int \prod_{j\in[m-2],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\Big(\int p_{\boldsymbol{\theta}_m}(\boldsymbol{x}_m|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big) \\ &\qquad \Big(\int p_{\boldsymbol{\theta}_{m-1}}(\boldsymbol{x}_{m-1}|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big) \\ &\cdots\cdots \\ &\geq \prod_{j\in[m],j\neq i} \Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\end{aligned} \tag{51}$$

Hence

$$-\mathbb{E}_p\big[\log \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] \leq -\mathbb{E}_p\big[\log \prod_{j\in[m],j\neq i} \Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big] \tag{52}$$

. Therefore, $\mathcal{R}_{\mathrm{SL}}(\boldsymbol{\Theta},\boldsymbol{\Phi}) \to 0$ leads to $-\mathbb{E}_p\big[\log \prod_{j\in[m],j\neq i} \Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big] \to 0$, which makes

$$-\mathbb{E}_p\big[\log \int \prod_{j\in[m],j\neq i} p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big] \to 0 \tag{53}$$

, *i.e.*, $\boldsymbol{\Phi}$, $\boldsymbol{\Theta}$ achieve the optima in the minimization. By (44) we know the supervised case is proved. In unsupervised case, we aim to prove $\mathcal{R}_{\mathrm{UL}} \to 0$ leads to

$$\min_{\boldsymbol{\Phi},\boldsymbol{\Theta}} -\mathbb{E}_{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}}\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})\big] \tag{54}$$

. It is obvious that, $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})$ is intractable, and we circumvent it via variational approximation where $p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i}|\boldsymbol{x}_i)$ is used as our proposal distribution:

$$
\begin{aligned}
&-\mathbb{E}_{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}}\big[\log p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})\big]\\
=&\int p_i(\boldsymbol{x}_i)\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big(-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]\\
&\qquad\qquad-\log\frac{p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})}{\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}}\Big)\prod_{k=1}^{m}d\boldsymbol{x}_k\\
=&\underbrace{\mathbb{E}_{p_i}-\mathrm{KL}\Big(\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big\|p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})\Big)}_{\leq 0}\\
&\qquad\qquad+\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]\\
\leq&\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]
\end{aligned}
\tag{55}
$$

Obviously, when $\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]=0$, there is

$$
\mathrm{KL}\Big(\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big\|p_{\boldsymbol{\Phi},\boldsymbol{\Theta}}(\boldsymbol{x}_i|\{\boldsymbol{x}_j\}_{j\in[m]\&j\neq i})\Big)=0
$$

Hence, in the unsupervised case, the problem is reduced to prove $\mathcal{R}_{UL}\to 0$ leads to

$$
\min_{\boldsymbol{\Theta},\boldsymbol{\Phi}}\ \mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]
\tag{56}
$$

. Using the similar technique in the supervised case, we obtain

$$
\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\geq\prod_{j\in[m],j\neq i}\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)
\tag{57}
$$

. Hence

$$
\begin{aligned}
&\underbrace{-\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]}_{\geq 0}\leq-\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}\big[\log\prod_{j\in[m],j\neq i}\Big(\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\Big)\big]\\
&\qquad\qquad=-\sum_{j\in[m],j\neq i}\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}\big[\log\int p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]
\end{aligned}
\tag{58}
$$

. If $\mathcal{R}_{\mathrm{UL}}\to 0$, by (43) there are $\forall i,j\in[m],i\neq j$

$$
\mathop{\mathbb{E}}_{\boldsymbol{x}_i\sim p_{\theta_i},\boldsymbol{x}_j\sim p_{\theta_j}}-\big[\log\int_{\boldsymbol{z}}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]\to 0
\tag{59}
$$

. Combine (58),(59) we have

$$
\mathbb{E}_{p_{\boldsymbol{\Theta},\boldsymbol{\Phi}}}-\big[\log\int\prod_{j\in[m],j\neq i}p_{\boldsymbol{\theta}_j}(\boldsymbol{x}_j|\boldsymbol{z})q_{\boldsymbol{\phi}_i}(\boldsymbol{z}|\boldsymbol{x}_i)d\boldsymbol{z}\big]\to 0
\tag{60}
$$

. The unsupervised case is proved. □

Proposition 4. *The optimum of the generation, inference and critic networks in*

$$
\min_{\boldsymbol{\Theta},\boldsymbol{\Phi}}\max_{\boldsymbol{\Omega}}(1-\gamma)\sum_{i=1}^{m}\mathcal{L}_{\mathrm{ALI}}^{(i)}+\gamma\sum_{i=1}^{m}\mathcal{L}_{\mathrm{MALI}}^{(i)}
\tag{61}
$$

refer to their saddle points in Lemma.2 if and only if $\forall i\in[m]$, *there exist* $p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}|\boldsymbol{z})q(\boldsymbol{z})=q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{x})$.

Proof. Let's rewrite

$$\min_{\boldsymbol{\Theta},\boldsymbol{\Phi}} \max_{\boldsymbol{\Omega}} V(\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{\Omega}) = (1-\gamma)\sum_{i=1}^{m}\mathcal{L}_{\text{ALI}}^{(i)} + \gamma\sum_{i=1}^{m}\mathcal{L}_{\text{MALI}}^{(i)} \tag{62}$$

- $\forall i \in [m]\ p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}|\boldsymbol{z})q(\boldsymbol{z}) = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{x})$, we prove that $V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*)$ has already reached a saddle point.

  Observe that

  $$q(\boldsymbol{z}) = \int p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x},\boldsymbol{z})d\boldsymbol{x} = \int q_{\boldsymbol{\phi}_i^*}(\boldsymbol{x},\boldsymbol{z})d\boldsymbol{x} = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z})$$

  . Hence $V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*)$ can be re-formulated as follow

  $$\begin{aligned}
  V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*) &= \sum_{i=1}^{m}\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}^*{}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})}\big[\log f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] \\
  &+ (1-\gamma)\mathbb{E}_{\hat{\boldsymbol{x}}_i,\boldsymbol{z}\sim p_{\boldsymbol{\theta}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})] + \gamma\sum_{j=1}^{m}\pi_j\mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i^*}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),\boldsymbol{z}\sim q_{\boldsymbol{\phi}_j^*}(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})] \\
  &= \sum_{i=1}^{m}\Bigg(\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}^*{}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})}\big[\log f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] \\
  &+ (1-\gamma)\mathbb{E}_{\hat{\boldsymbol{x}}_i,\boldsymbol{z}\sim p_{\boldsymbol{\theta}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})] + \gamma\mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i^*}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),\boldsymbol{z}\sim q(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})]\big(\sum_{j=1}^{m}\pi_j\big)\Bigg) \\
  &= \sum_{i=1}^{m}\underbrace{\Bigg(\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}^*{}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})}\big[\log f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] + \mathbb{E}_{\hat{\boldsymbol{x}}_i,\boldsymbol{z}\sim p_{\boldsymbol{\theta}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})]\Bigg)}_{\mathcal{L}_{\text{ALI}}^{(i)}(\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*)}
  \end{aligned}$$

  . In this way, the minimax optimization $V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*)$ *w.r.t.* $\{\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*\}_{i=1}^m$ can be decomposed into m terms $\mathcal{L}_{\text{ALI}}^{(i)}(\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*)$ $(\forall i \in [m])$. Since each domain-specific optimization $\mathcal{L}_{\text{ALI}}^{(i)}(\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*)$ has already reached a saddle point (Lemma.2.1), $V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*)$ has also reached a saddle point.

- Here we prove that, provided $\{\boldsymbol{\theta}_i^*,\boldsymbol{\phi}_i^*,\boldsymbol{\omega}_i^*\}_{i=1}^m$ denoting a saddle point of $V(\boldsymbol{\Theta}^*,\boldsymbol{\Phi}^*,\boldsymbol{\Omega}^*)$, it holds $\forall i \in [m]$, $p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}|\boldsymbol{z})q(\boldsymbol{z}) = q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{x})$.

  It is obviously observed that, if the generation and inference nets are fixed in parameters ($\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$), $V(\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{\Omega})$ *w.r.t.*, $\boldsymbol{\Omega} = \{\boldsymbol{\omega}_i\}_{i=1}^m$ are separately optimized by domains:

  $$\begin{aligned}
  \max_{\boldsymbol{\Omega}} V(\boldsymbol{\Theta},\boldsymbol{\Phi},\boldsymbol{\Omega}) &= \sum_{i=1}^{m}\max_{\boldsymbol{\omega}_i}\Big(\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})}\big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] \\
  &+ (1-\gamma)\mathbb{E}_{\hat{\boldsymbol{x}}_i,\boldsymbol{z}\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})] + \gamma\sum_{j=1}^{m}\pi_j\mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),\boldsymbol{z}\sim q_{\boldsymbol{\phi}_j}(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})]\Big)
  \end{aligned} \tag{63}$$

  . Then for the i^{th} domain ($\forall i \in [m]$), there exists

  $$\begin{aligned}
  &\max_{\boldsymbol{\omega}_i}\Big(\mathbb{E}_{\boldsymbol{x}_i,\hat{\boldsymbol{z}}\sim q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})}\big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\hat{\boldsymbol{z}})\big] + (1-\gamma)\mathbb{E}_{\hat{\boldsymbol{x}}_i,\boldsymbol{z}\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})] \\
  &\qquad + \gamma\sum_{j=1}^{m}\pi_j\mathbb{E}_{\hat{\boldsymbol{x}}_i\sim p_{\boldsymbol{\theta}_i}(\hat{\boldsymbol{x}}_i|\boldsymbol{z}),\boldsymbol{z}\sim q_{\boldsymbol{\phi}_j}(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i}(\hat{\boldsymbol{x}}_i,\boldsymbol{z})]\Big) \\
  =&\max_{\boldsymbol{\omega}_i}\int_{\boldsymbol{x}_i}\int_{\boldsymbol{z}}\Big(q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i,\boldsymbol{z})\big[\log f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\boldsymbol{z})\big] + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\big[(1-\gamma)q(\boldsymbol{z}) \\
  &\qquad + \gamma\sum_{j=1}^{m}\pi_j q_{\boldsymbol{\phi}_j}(\boldsymbol{z})\big][\log 1 - f_{\boldsymbol{\omega}_i}(\boldsymbol{x}_i,\boldsymbol{z})]\big)\Big)d\boldsymbol{x}d\boldsymbol{z}
  \end{aligned} \tag{64}$$

Similar to the speculation in [2], the optimal critic network $f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i, \boldsymbol{z})$ are represented as:

$$f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i, \boldsymbol{z}) = \frac{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z})}{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})} \\ s.t.\ \ \mu_{\boldsymbol{\Phi}}(\boldsymbol{z}) = (1-\gamma)q(\boldsymbol{z}) + \gamma \sum_{j=1}^{m} \pi_j q_{\boldsymbol{\phi}_j}(\boldsymbol{z}) \tag{65}$$

where $\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})$ is a probability mixture *w.r.t.* γ and $\{\pi_j\}_{j=1}^m$, which consists of feature prior $q(\boldsymbol{z})$ and parameterized distributions $\{q_{\boldsymbol{\phi}_j}(\boldsymbol{z})\}_{j=1}^m$.

Therefore given $\forall i \in [m]$ and their optimal critic networks with parameters $\boldsymbol{\Omega}^*$, we consider the optimal $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ by minimizing $\mathcal{L}_{\mathrm{MALI}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{\Omega}^*)$.

$$\begin{aligned}
&\mathcal{L}_{\mathrm{MALI}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{\Omega}^*) = \sum_{i=1}^{m} \Big(\mathbb{E}_{\boldsymbol{x}_i, \boldsymbol{z} \sim q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z})} \big[\log f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i, \boldsymbol{z}) \big] \\
&+\ (1-\gamma)\mathbb{E}_{\boldsymbol{x}_i, \boldsymbol{z} \sim p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i, \boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i, \boldsymbol{z})] + \gamma \sum_{j=1}^{m} \pi_j \mathbb{E}_{\boldsymbol{x}_i \sim p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z}), \boldsymbol{z} \sim q_{\boldsymbol{\phi}_j}(\boldsymbol{z})}[\log 1 - f_{\boldsymbol{\omega}_i^*}(\boldsymbol{x}_i, \boldsymbol{z})] \Big) \\
&= \sum_{i=1}^{m} \Big(\mathbb{E}_{\boldsymbol{x}_i, \boldsymbol{z} \sim q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z})} \big[\log \frac{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z})}{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})} \big] \\
&\qquad + \mathbb{E}_{\boldsymbol{x}_i, \boldsymbol{z} \sim p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})} \big[\log \frac{p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})}{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})} \big] \Big) \\
&= \sum_{i=1}^{m} -2\log 2 + D_{\mathrm{KL}}\Big(q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) || \frac{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})}{2} \Big) \\
&\qquad + D_{\mathrm{KL}}\Big(p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z}) || \frac{q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) + p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z})}{2} \Big) \\
&= -2m\log 2 + 2\sum_{i=1}^{m} D_{\mathrm{JS}}\Big(q_{\boldsymbol{\phi}_i}(\boldsymbol{x}_i, \boldsymbol{z}) || p_{\boldsymbol{\theta}_i}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}}(\boldsymbol{z}) \Big)
\end{aligned} \tag{66}$$

where $D_{\mathrm{JS}}(\cdot||\cdot)$ indicates Jensen–Shannon divergence. So $\min_{\boldsymbol{\Theta}, \boldsymbol{\Phi}} \mathcal{L}_{\mathrm{MALI}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{\Omega}^*)$ equals to searching a set of parameters $\{\boldsymbol{\Theta}^*, \boldsymbol{\Phi}^*\}$ that satisfy

$$q_{\boldsymbol{\phi}_i^*}(\boldsymbol{x}_i, \boldsymbol{z}) = p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i|\boldsymbol{z})\mu_{\boldsymbol{\Phi}^*}(\boldsymbol{z}),\ s.t.\ \forall i \in [m] \tag{67}$$

Marginalize out $\boldsymbol{x}_i$ then we get

$$\begin{aligned}
q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}) &= \mu_{\boldsymbol{\Phi}^*}(\boldsymbol{z}), \quad s.t.\ \forall i \in [m] \\
\leftrightharpoons q_{\boldsymbol{\phi}_i^*}(\boldsymbol{z}) &= q(\boldsymbol{z}), \quad s.t.\ \forall i \in [m]
\end{aligned} \tag{68}$$

So

$$q_{\boldsymbol{\phi}_i^*}(\boldsymbol{x}_i, \boldsymbol{z}) = p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i|\boldsymbol{z})q(\boldsymbol{z}) = p_{\boldsymbol{\theta}_i^*}(\boldsymbol{x}_i, \boldsymbol{z}),\ s.t.\ \forall i \in [m] \tag{69}$$

Since $\boldsymbol{\Omega}^*$ has already reached a saddle point of $\max_{\boldsymbol{\Omega}} \min_{\boldsymbol{\Phi}, \boldsymbol{\Theta}} \mathcal{L}_{\mathrm{MALI}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{\Omega})$, $\boldsymbol{\Theta}^*$, $\boldsymbol{\Phi}^*$ imply the same saddle points of $\max_{\boldsymbol{\Omega}} \min_{\boldsymbol{\Phi}, \boldsymbol{\Theta}} \mathcal{L}_{\mathrm{MALI}}(\boldsymbol{\Theta}, \boldsymbol{\Phi}, \boldsymbol{\Omega})$ *w.r.t.* $\boldsymbol{\Omega}^*$.

Conclude the proof. □

Appendix.B

Balance between efficacy and scalability

Our synthetic data experiment is conducted on m-domain scenarios. Each domain is constructed by a 5-component Gaussian mixture model with a standard derivation 0.2. We employ different centroid

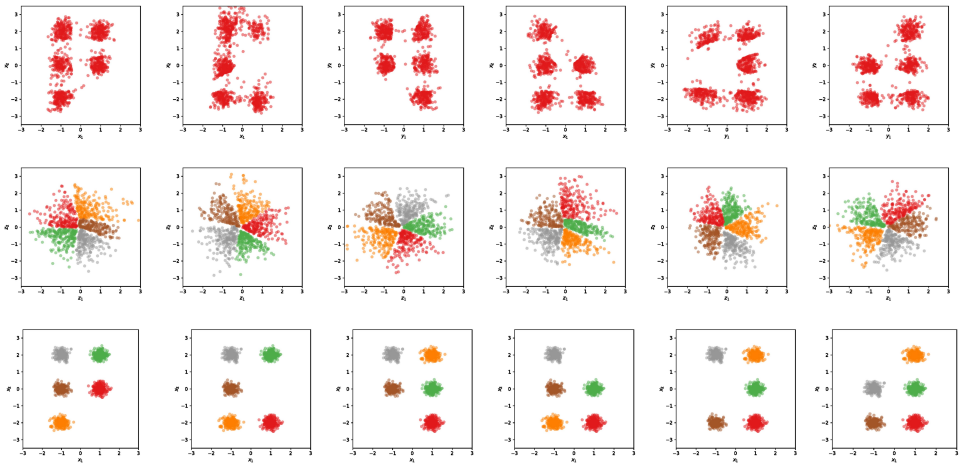
combinations to specify those diverse domains. As for the latent domain prior $q(z)$, we accept the same protocol in [9], where the isotropic Gaussian is set with mean $(0, 0)$ and standard derivation 1.0.

We testify our MMI-ALI along with StarGAN [11] and CycleGAN [9]. It is noted that we have not employed JointGAN as a comparison, due to JointGAN is not a scalable DGM and out of our research focus. Sine StarGAN is the first work attempting to solve m-domain transformation problem, we employ it as a scalable baseline in this empirical study.

For a fair comparison, we proposed to equalize our subnet selection and training options. In specific, we accept a 2-layered fully-connected architecture ($2 \times 256 \times 256 \times 2$) as the generation, inference networks, and the other 2-layered fully-connected network ($4 \times 256 \times 256 \times 2$) as the pairwise critic network in MMI-ALI and CycleGAN. As for StarGAN, the original paper shows a complicated architecture and uppermost technique to solve attribute-based transformation, while this technique can not really reveal the efficacy of StarGAN. Therefore, we follow the StarGAN pipeline to build a generation network ($(2 + M) \times 256 \times 256 \times 2$) and critic network ($2 \times 256 \times 256 \times (2 + M)$) with similar architecture setting to the corresponding subnets in MMI-ALI and CycleGAN. Moreover, all these framework have been trained by vanilla GAN loss, and accept l_2 losses to implement the cycle domain reconstruction.

In 6-domain setup, CycleGANs have to employ 45 subnets to complete all the transformation, therefore we only visualize MMI-ALI and StarGAN. MMI-ALI allows a divide-and-conquer extension. Suppose domain datasets are unevenly distributed in different local machines. In each machine, we separately run a slave MMI-ALI to obtain a local coherent semantic feature space. After that, we employ a universal machine (server) to launch a master MMI-ALI over local domain-encoded feature spaces. This two-stage process yields the model parallelism and potentially reduces our local storage of data and parameters. We apply these extended MMI-ALI to the 4,5,6-domain empirical study. The visualization sees Fig 10 ,11.

Figure 10: The generation performance of MMI-ALI in 6-domain. The last row is the groundturth and the first row denotes the generation results. The second row denotes the semantic-shared space(In our unsupervise setting, **the same colors across domain have no corresponding relations**). The joint inferred feature space presents the mutual classfication boundaries patterns across 6 domains.



Geometry-varying m-domain

We present the complete experimental evaluation of StarGAN, CycleGAN and our MMI-ALI in Rotated MNIST in unsupervised learning setup. The baseline comparison and the ablation of MMI-ALI (without MMI-induced regularization and adversarial ensemble learning, respectively) are shown in Table.6. More generation visualizations are shown in Fig 12 -15 .

Cross-m-domain visual style transfer.

In the paper, we have reported the quantitative evaluation of object transfiguration based on AMT. Here we also conduct the ablation of IS in Table.7 and ablate the model with $\lambda = \{0,0.2,0.5,0.8\}$ to the IS $\{1.22,1.29,1.43,1.32\}$. More visualizations see Fig 16 -18.

Figure 11: The visualization of StarGAN. It caueses a heavy performance drop reported in our paper.

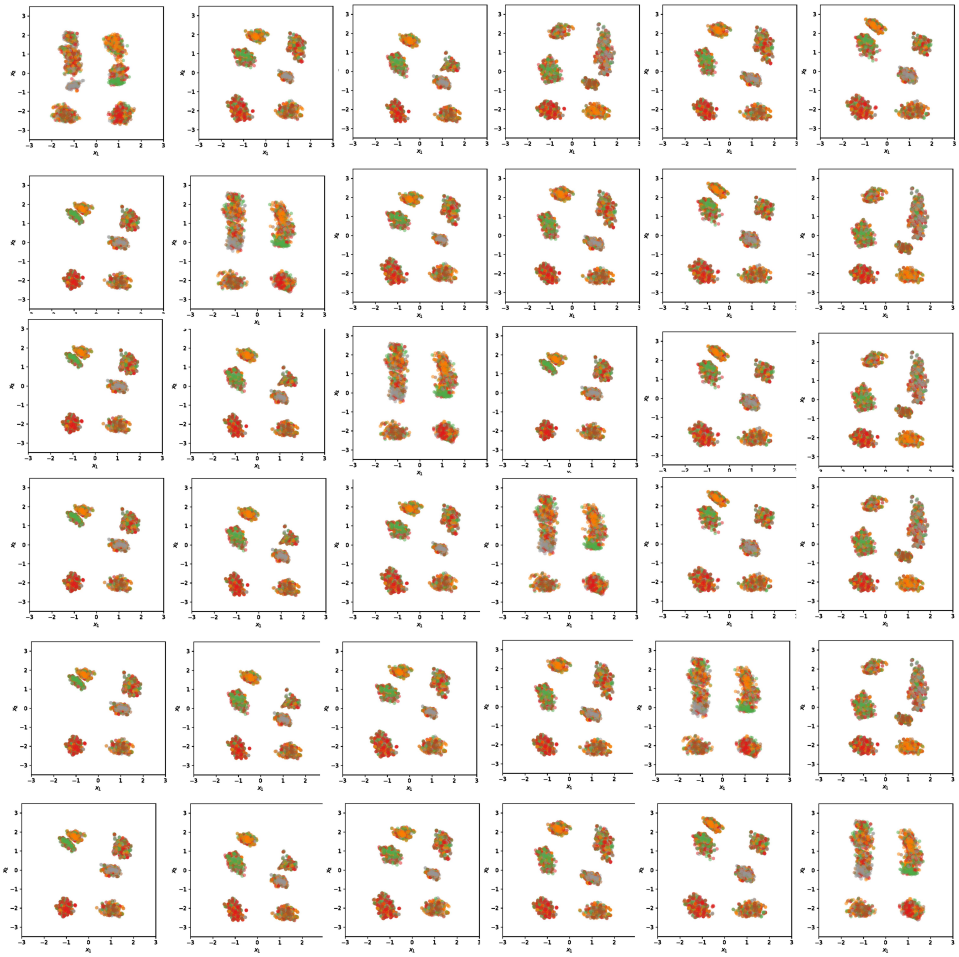


Table 6: IS of StarGAN (ST), CycleGAN (CG) and MMI-ALI(MA) in unsupervised cross-domain generation case.

	$-\frac{\pi}{2}\to 0$	$\frac{\pi}{2}\to 0$	$0\to\frac{\pi}{2}$	$-\frac{\pi}{2}\to\frac{\pi}{2}$	$-\frac{\pi}{2}\to 0$	$\frac{\pi}{2}\to-\frac{\pi}{2}$
StarGAN	1.00±0.00	1.00±0.00	1.00±0.00	1.00±0.00	1.00±0.00	1.00±0.00
CycleGAN	8.34±0.12	6.13±0.37	2.25±0.02	2.38±0.06	1.71±0.03	1.04±0.01
MMI-ALI (wo MMI)	7.48±0.05	6.06±0.10	3.19±0.04	2.90±0.05	2.73±0.09	2.47±0.12
MMI-ALI ($\lambda = 0$)	8.34±0.20	8.27±0.10	**3.26**±0.06	3.06±0.10	3.15±0.11	2.92±0.12
MMI-ALI	**8.99**±0.06	**9.01**±0.00	2.95±0.08	**3.86**±0.12	**3.31**±0.12	**3.08**±0.05

It is worth noting that, in object transfiguration, generation metrics like IS and FID, are not desirable to evaluate the transfer performance. Let's consider a simple instance: an Auto-encoder perfectly trained with the three domain data. Given any images from these domains, it is able to reconstruct the image to achieve high score on IS and FID. However, this Auto-encoder obviously fails in transferring the visual realism from one domain to another. To this end, we solely use the generation metric (IS) to make an ablation instead of a comparison. Since object transfiguration lacks groundtruth, we could not provide a more fair comparison than resorting to AMT in the paper. Supervised learning is also unavailable.

To more thoroughly reflect MMI-ALI's performances in quantitative measures, we take another visual style transfer experiment based on three domains (real images (R), segmentation labels (Seg) and sketches (Ske)). In these cross-3-domain transfers, each instance for training and testing is given as a triplet, so we are able to concurrently evaluate unsupervised and supervised learning. Since each transfer target is provided (*e.g.*, given an image, the transfer target is segmentation label or sketch, vice versa), we are able to calculate MSE between the groundtruth and the generated transfer

Table 7: Ablation (IS) of MMI-ALI in visual style transfer.

	Okapi2Zebra	Okapi2Horse	Zebra2Okapi	Horse2Okapi
Real domain	3.00±0.08	1.36±0.06	1.88±0.06	1.88±0.02
w MMI	1.43±0.14	1.14±0.21	1.65±0.22	1.48±0.12
wo MMI	1.04±0.02	1.02±0.02	1.00±0.00	1.00±0.00

result, so as to evaluate the conditional generation performance. Combining FID, the learned implicit joint distribution could be comprehensively revealed (Note that, IS relies on a ImageNet-based deep classifier for evaluation. As segmentation labels and sketches are not included in this classifier, IS is inappropriate to measure generation quality in this experiment).The quantitative results are shown in Table 8 and 9 ; the visualizations are shown in Fig 20 -31.

Table 8: FID in Cityscape

		R→Seg	Seg→R	R→Ske	Ske→R	Seg→Ske	Ske→Seg
Unsuper	ST	405.16	372.59	385.08	388.97	357.19	417.39
	CG	224.04	213.43	164.65	222.24	60.20	144.07
	Ours	202.93	254.41	150.98	246.04	101.30	192.13
Super	ST	382.90	440.53	419.11	383.72	400.70	299.82
	CG	217.28	260.41	171.04	223.43	65.18	228.61
	Ours	250.48	246.01	196.06	229.45	55.76	143.20

Table 9: MSE in Cityscape

		R→Seg	Seg→R	R→Ske	Ske→R	Seg→Ske	Ske→Seg
Unsuper	ST	4549.47	4812.46	28939.01	19318.652	22652.283	17333.963
	CG	3799.07	3101.66	4746.15	3981.52	3429.18	1957.42
	Ours	3239.67	3709.60	5021.60	3699.43	4360.70	2963.88
Super	ST	5117.48	5687.41	22290.52	4110.17	22427.65	3196.83
	CG	1996.17	3008.97	5085.19	3474.40	1153.41	3273.60
	Ours	2412.19	3185.98	4230.68	3467.36	4936.28	2765.82

Finally, we also add the experiment of season transfer. See Fig 19.

Cross-m-emotion text style transfer.

More visualization sees Fig 32-33 .

Another quantitative experiments.

We also add other two experiments based on CelebA [1] and CMP Façade [2].

To fairly compare CycleGAN and StarGAN on CelebA, we conduct experiments of our FALI under unsupervised setting and demonstrate the performance gains by 17.59

More elaborations of the experiments: 1. CelebA. To ensure unsupervised learning setup, CelebA is split into multiple domains according to some attributes (we choose three hairstyles, e.g., blonde, black and brown, to constitute a three-domain transformation), while the other attributes are not permitted to guide the omni-domain transformation. We follow the architectures in the second experiment and employ the same hyper-parameters. During training, we apply WGAN-GP loss [4] in StarGAN, CycleGAN, and FALI, so as to provide a fair comparison and stable generation performances. In terms of FID (lower is better), StarGAN, CycleGAN and FALI respectively obtain 0.324, 0.394 and 0.267 ([0.324-0.267]/0.324x100%=17.59%) on the average of the three-domain transformation performances. In terms of IS (higher is better), StarGAN, CycleGAN and FALI respectively obtain 2.597+/-0.078, 2.251+/-0.209 and 2.818+/-0.246 ([2.818-2.597]/2.597x100%=8.51%) on the average of the three-domain transformation performances.

2. CMP Façade. The dataset consists of architecture images with their segmentation labels and we shuffle their pairing relationships, so as to perform unsupervised domain transformation across image and label spaces. Since the problem is very close to style transfer, thus, we follow the same backbones, hyper-parameter setting and training setup in the third experiment in our paper. The quantitative results are based upon label2image generation on IS. The IS on the natural images in Façade is 3.702+/-0.36, which can be treated as the performance upper bound. The IS scores of

CycleGAN and FALI are 2.758+/-0.392 and 2.860+/-0.371 ([2.860-2.758]/2.758x100%=3.63%), respectively.

These visualization results are deferred in our journal version.

References

[1] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017.

[4] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018.

[5] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. Style transfer from non-parallel text by cross-alignment. 2017.

[6] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.

[7] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*, 2017.

[8] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation.

[9] Chunyuan Li, Hao Liu, Changyou Chen, Yuchen Pu, Liqun Chen, Ricardo Henao, and Lawrence Carin. Alice: Towards understanding adversarial learning for joint distribution matching. In *Advances in Neural Information Processing Systems*, pages 5501–5509, 2017.

[10] Yunchen Pu, Shuyang Dai, Zhe Gan, Weiyao Wang, Guoyin Wang, Yizhe Zhang, Ricardo Henao, and Lawrence Carin. Jointgan: Multi-domain joint distribution learning with generative adversarial nets. 2018.

[11] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. *arXiv preprint arXiv:1711.09020*, 2017.

[12] Robi Polikar. Ensemble learning. *Scholarpedia*, 4(1):1–34, 2009.

[13] Anthony J Bell. The co-information lattice. In *Proceedings of the Fifth International Workshop on Independent Component Analysis and Blind Signal Separation: ICA*, volume 2003, 2003.

[14] William J. Mcgill. Multivariate information transmission. *Transactions of the Ire Professional Group on Information Theory*, 4(4):93–111, 2003.

[15] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[16] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. 2017.

[17] Mohamed Ishmael Belghazi, Sai Rajeswar, Olivier Mastropietro, Negar Rostamzadeh, Jovana Mitrovic, and Aaron Courville. Hierarchical adversarially learned inference. 2018.

[18] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

[19] Zhe Gan, Liqun Chen, Weiyao Wang, Yuchen Pu, Yizhe Zhang, Hao Liu, Chunyuan Li, and Lawrence Carin. Triangle generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 5253–5262, 2017.

[20] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. It takes (only) two: Adversarial generator-encoder networks. 2017.

[21] Zhijie Deng, Hao Zhang, Xiaodan Liang, Luona Yang, Shizhen Xu, Jun Zhu, and Eric P Xing. Structured generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 3902–3912, 2017.

[22] Scott E Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. In *Advances in Neural Information Processing Systems*, pages 217–225, 2016.

[23] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, 2016.

[24] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[25] Bo Zhao, Bo Chang, Zequn Jie, and Leonid Sigal. Modular generative adversarial networks. 2018.

[26] Hirokazu Kameoka, Takuhiro Kaneko, Tanaka Kou, and Nobukatsu Hojo. Stargan-vc: Non-parallel many-to-many voice conversion with star generative adversarial networks. 2018.

[27] Valentin Khrulkov and Ivan Oseledets. Geometry score: A method for comparing generative adversarial networks. 2018.

[28] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. 2017.

[29] Wang Zhou, Bovik Alan Conrad, Sheikh Hamid Rahim, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. 13(4):600–612, 2004.

[30] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. 2016.

[31] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. 2015.

[32] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[33] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *International Journal of Computer Vision*, 125(1-3):3–18, 2015.

[34] Haoye Dong, Xiaodan Liang, Ke Gong, Hanjiang Lai, Jia Zhu, and Jian Yin. Soft-gated warping-gan for pose-guided person image synthesis. 2018.

[35] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. 2018.

[36] Xianda Zhou and William Yang Wang. Mojitalk: Generating emotional responses at scale. 2017.

[37] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Skip-thought vectors. *Advances in Neural Information Processing Systems*, 28, 2015.

[38] Nick Craswell. Mean reciprocal rank. 2009.

[39] David Barber and Felix V. Agakov. The im algorithm: A variational approach to information maximization. *Advances in Neural Information Processing Systems*, 2003.

[40] Martin Egozcue, L Fuentes Garcia, and Wing-Keung Wong. On some covariance inequalities for monotonic and non-monotonic functions.

Figure 12: The interpolation affects the joint-distribution-based example generation.

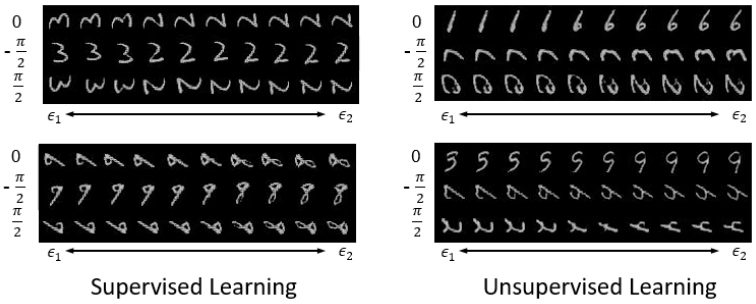


Figure 13: The transfer generation of MMI-ALI in Rotated Mnist.

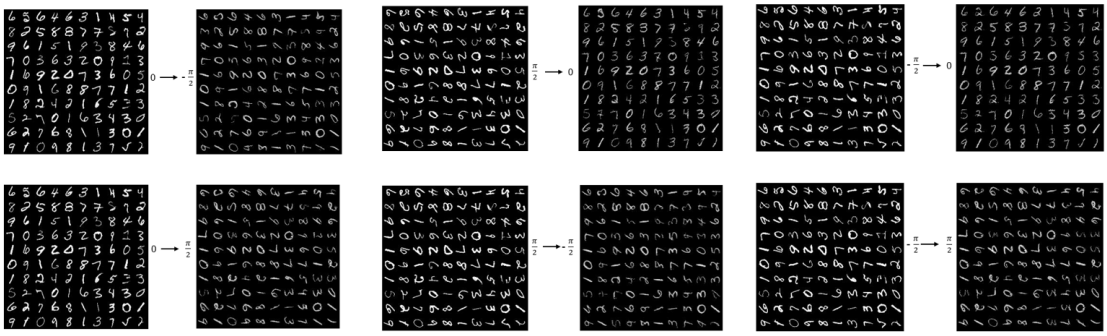


Figure 14: The transfer generation of CycleGAN in Rotated Mnist.

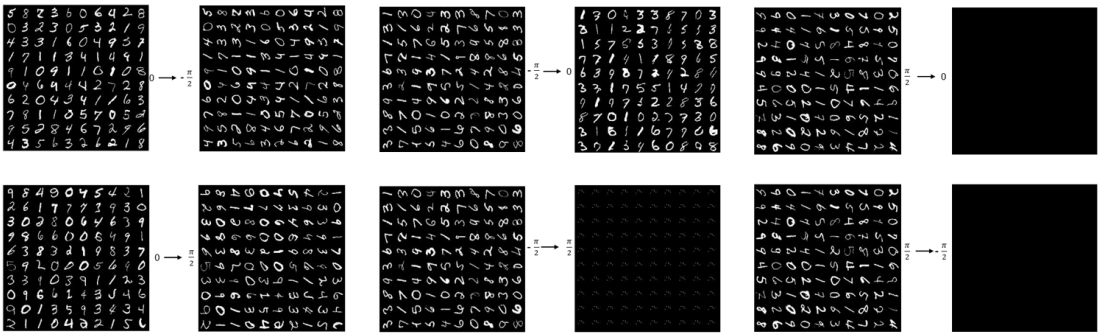


Figure 15: The transfer generation of StarGAN in Rotated Mnist.

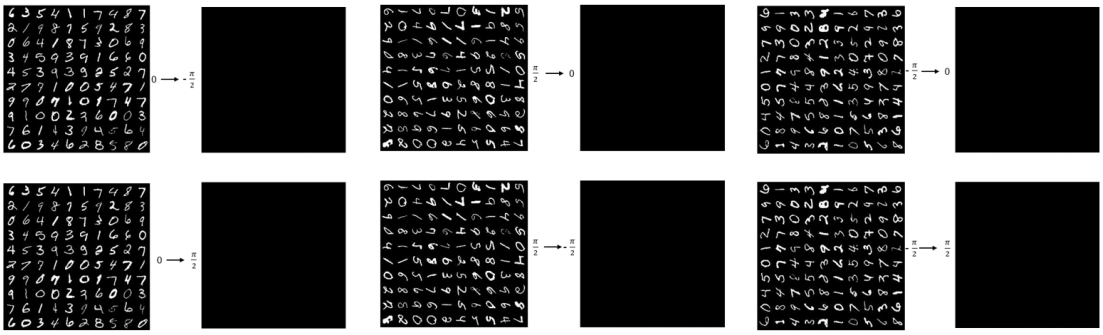


Figure 16: The object transfiguration from Horse to others.

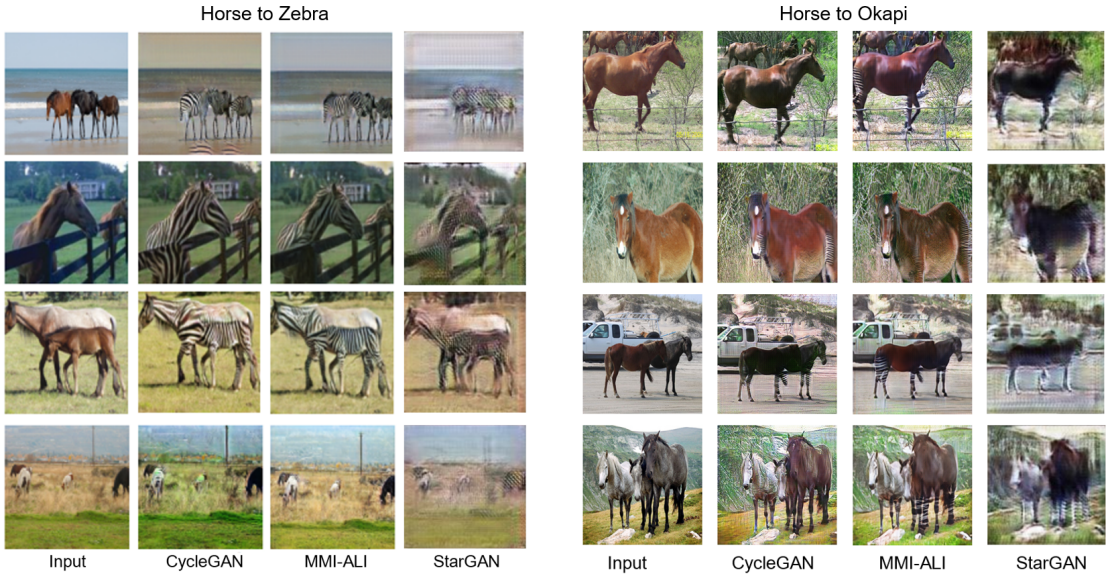


Figure 17: The object transfiguration from Okapi to others.

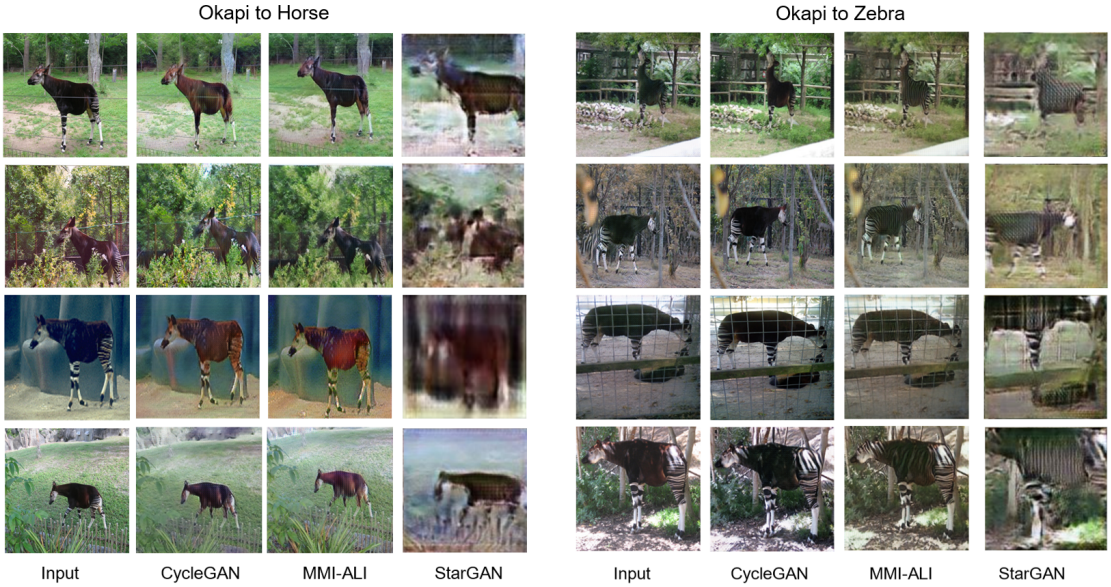


Figure 18: The object transfiguration from Zebra to others.

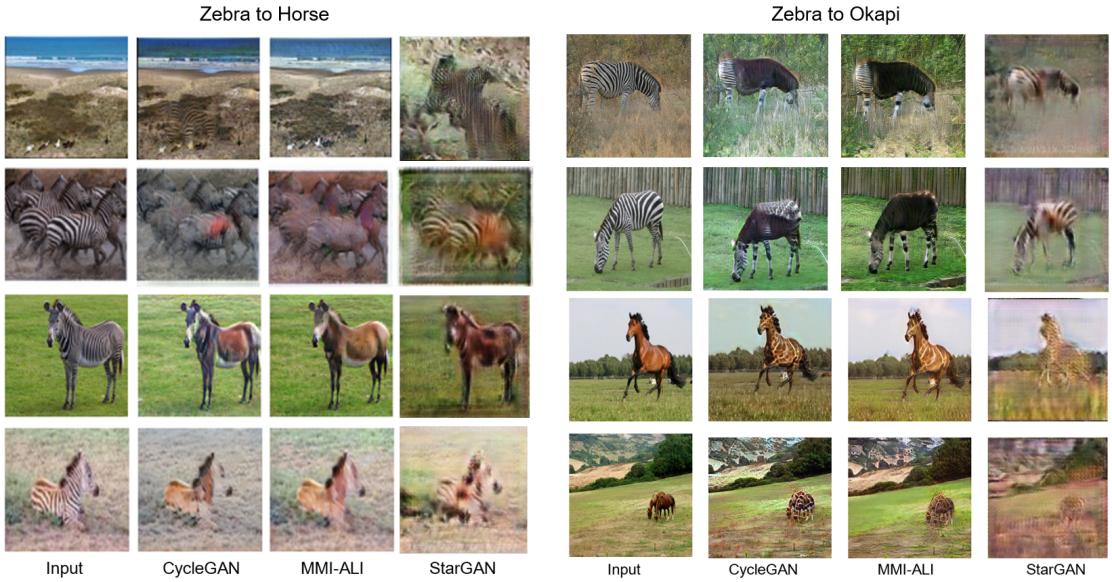


Figure 19: The season transfer (Summer and Winter).

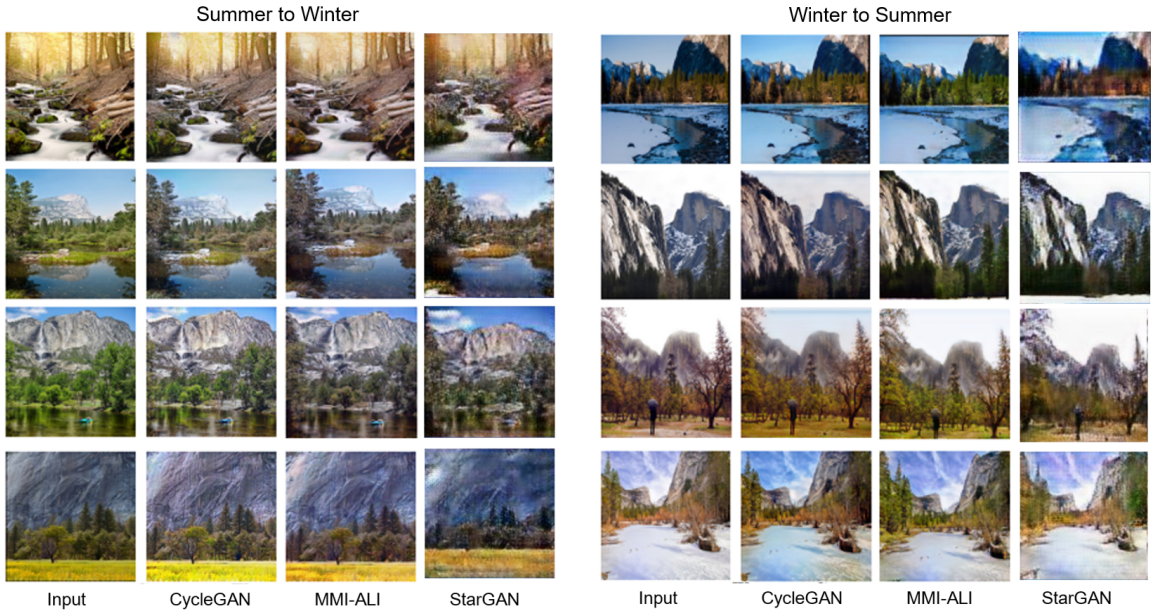


Figure 20: The supervised transfer generation in Cityscape.

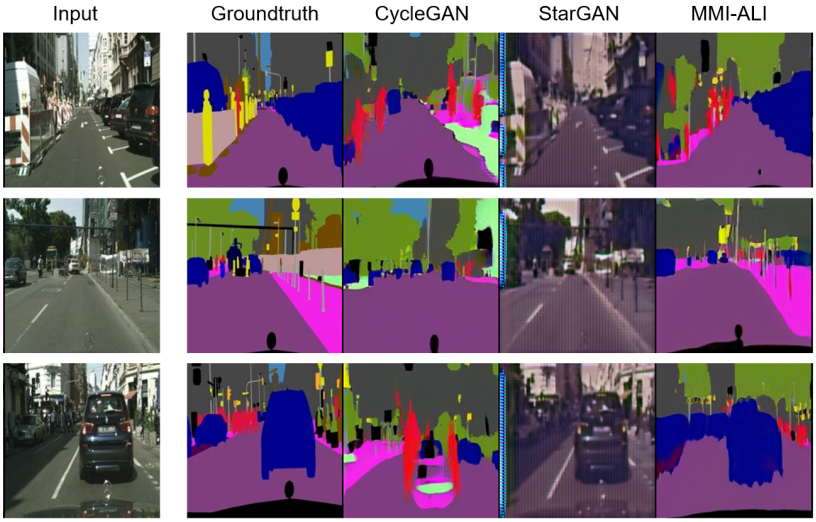


Figure 21: The supervised transfer generation in Cityscape.

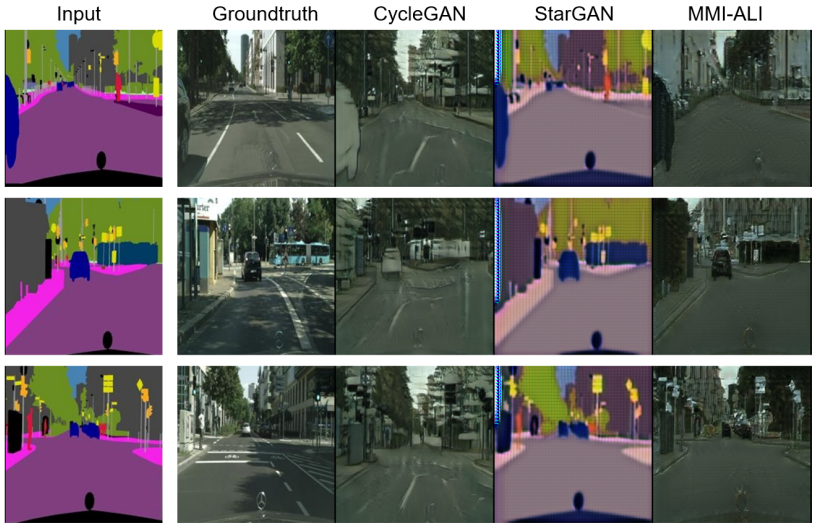


Figure 22: The supervised transfer generation in Cityscape.

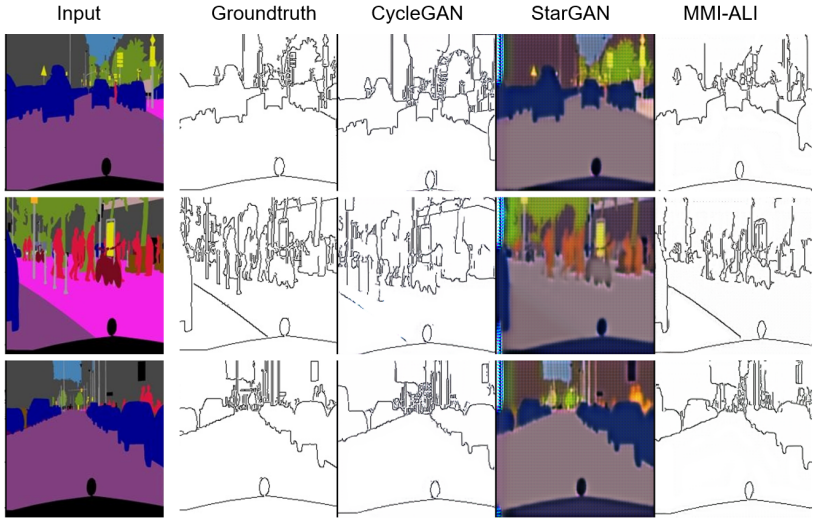


Figure 23: The supervised transfer generation in Cityscape.

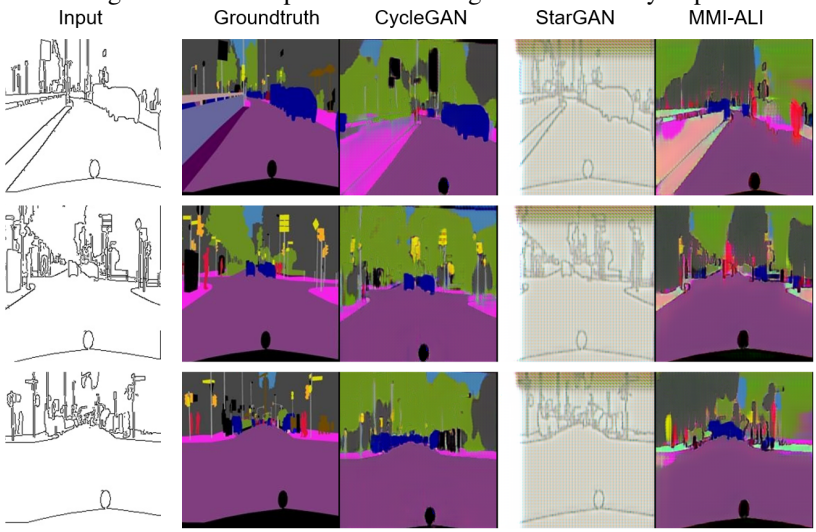


Figure 24: The supervised transfer generation in Cityscape.

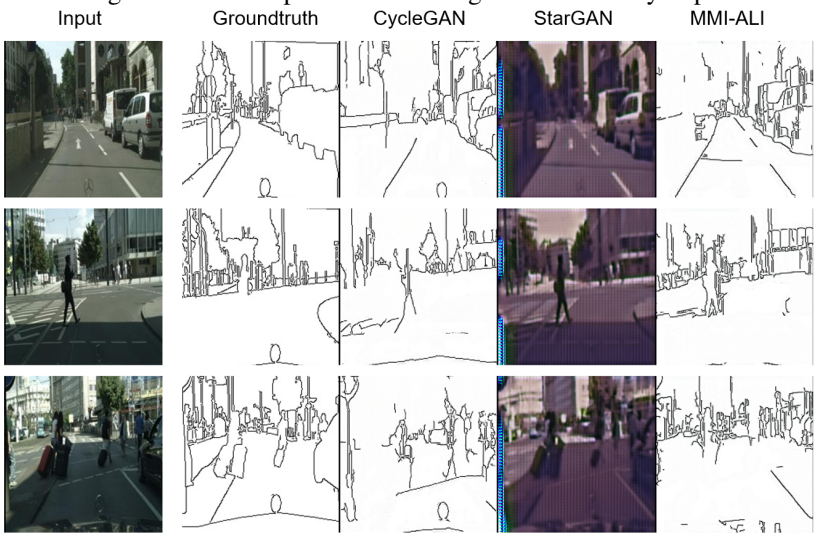


Figure 25: The supervised transfer generation in Cityscape.

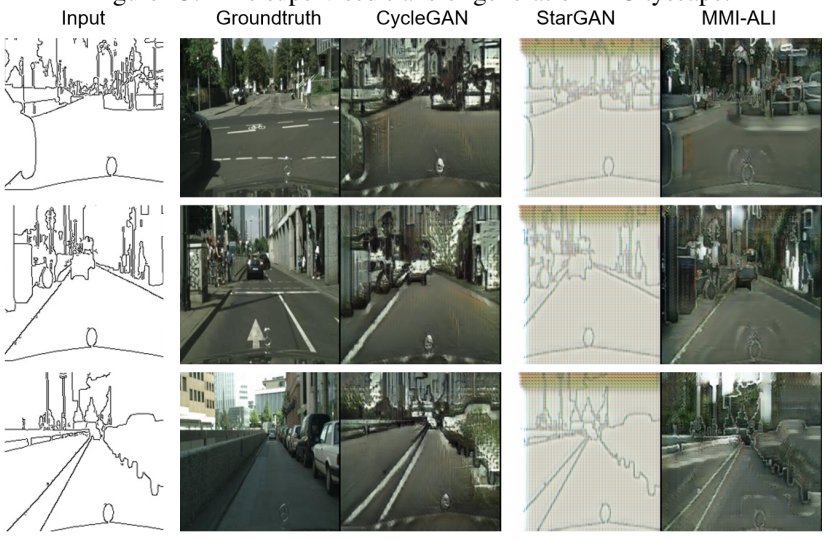


Figure 26: The supervised transfer generation in Cityscape.

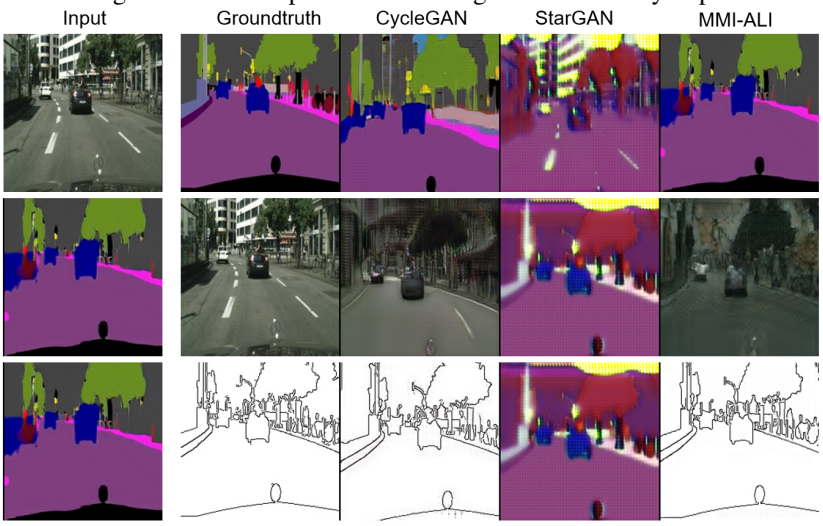


Figure 27: The supervised transfer generation in Cityscape.

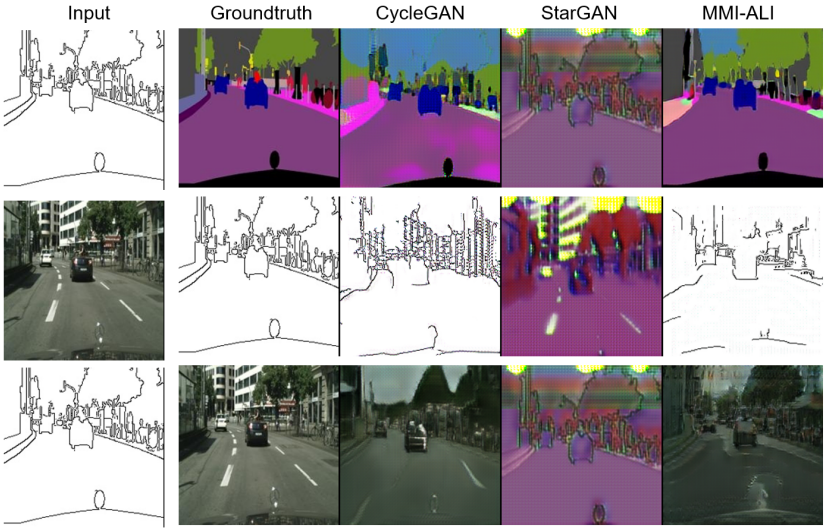


Figure 28: The supervised transfer generation in Cityscape.

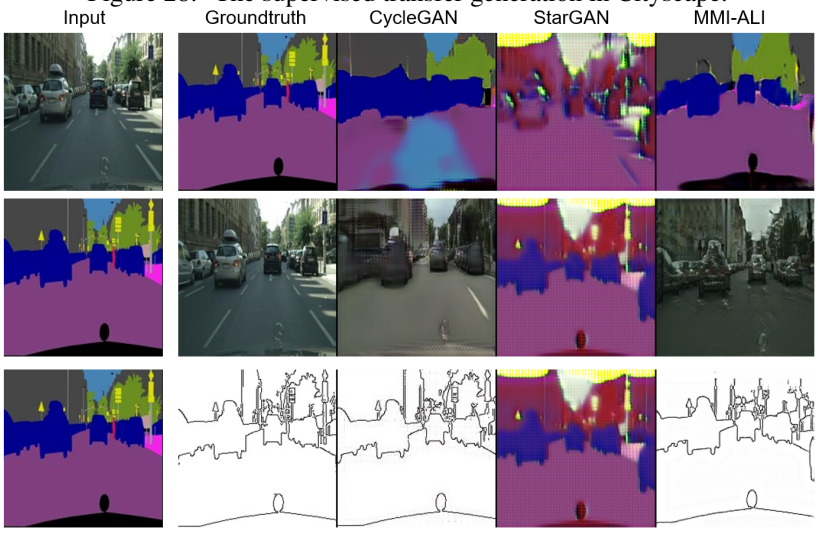


Figure 29: The supervised transfer generation in Cityscape.

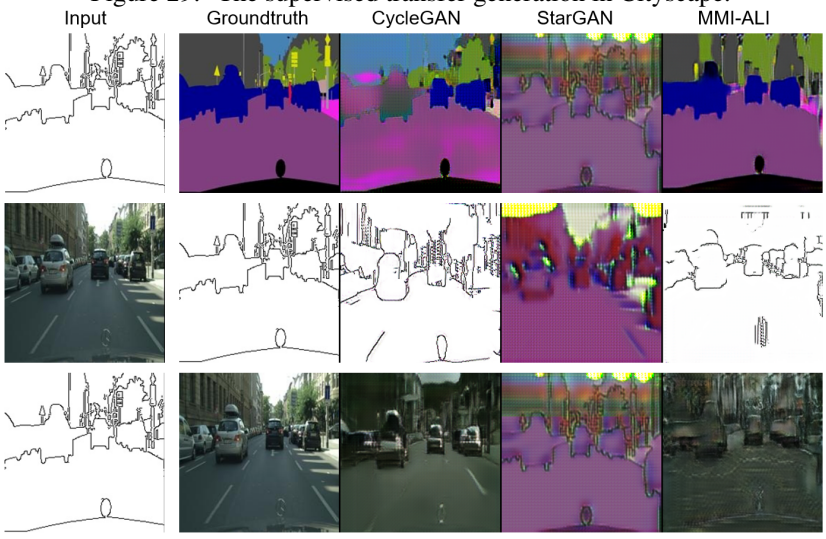


Figure 30: The supervised transfer generation in Cityscape.

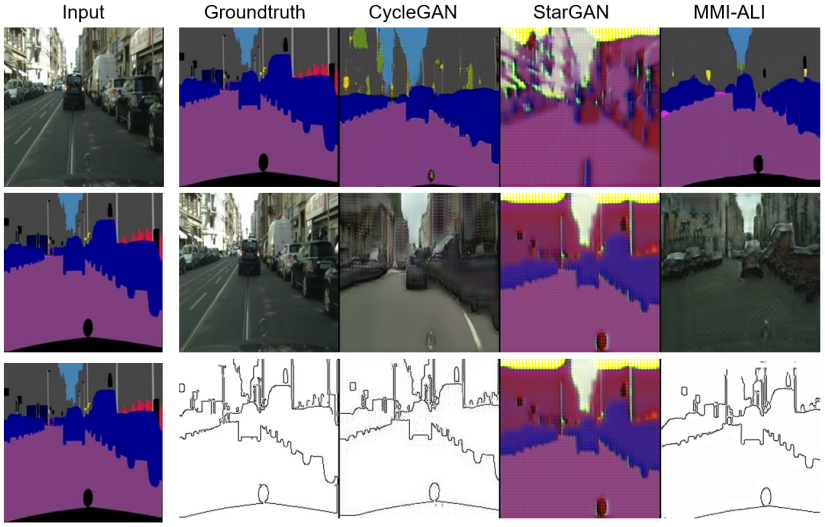


Figure 31: The supervised transfer generation in Cityscape.

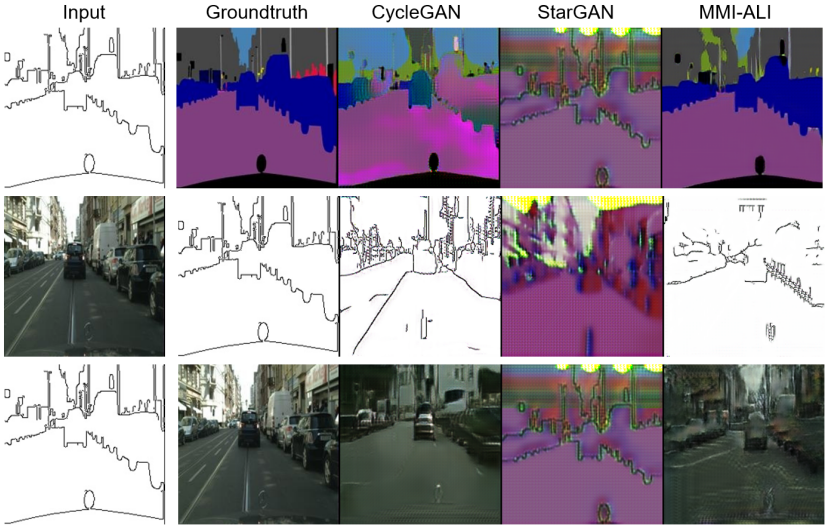


Figure 32: The visualization of emotion transfer.

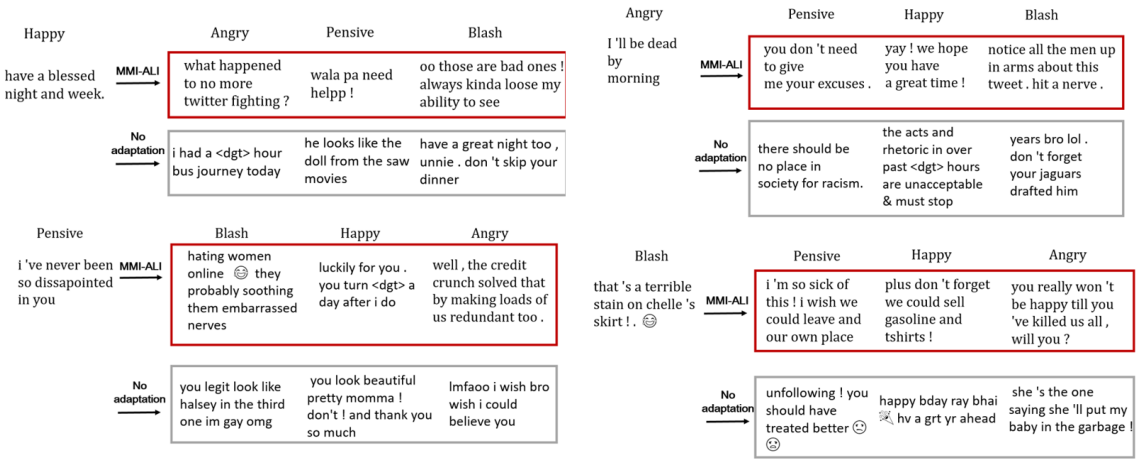


Figure 33: The visualization of emotion transfer.