# Reusing Discriminators for Encoding:
# Towards Unsupervised Image-to-Image Translation

Runfa Chen, Wenbing Huang, Binghui Huang, Fuchun Sun*, Bin Fang

Institute for Artificial Intelligence, Tsinghua University (THUAI)

Beijing National Research Center for Information Science and Technology (BNRist),

State Key Lab on Intelligent Technology and Systems,

Department of Computer Science and Technology, Tsinghua University, Beijing, P.R.China

crf18@mails.tsinghua.edu.cn, hwenbing@126.com

{hbh18@mails., fcsun@, fangbin@mail.}tsinghua.edu.cn

## Abstract

*Unsupervised image-to-image translation is a central task in computer vision. Current translation frameworks will abandon the discriminator once the training process is completed. This paper contends a novel role of the discriminator by reusing it for encoding the images of the target domain. The proposed architecture, termed as NICE-GAN, exhibits two advantageous patterns over previous approaches: First, it is more compact since no independent encoding component is required; Second, this plug-in encoder is directly trained by the adversary loss, making it more informative and trained more effectively if a multi-scale discriminator is applied. The main issue in NICE-GAN is the coupling of translation with discrimination along the encoder, which could incur training inconsistency when we play the min-max game via GAN. To tackle this issue, we develop a decoupled training strategy by which the encoder is only trained when maximizing the adversary loss while keeping frozen otherwise. Extensive experiments on four popular benchmarks demonstrate the superior performance of NICE-GAN over state-of-the-art methods in terms of FID, KID, and also human preference. Comprehensive ablation studies are also carried out to isolate the validity of each proposed component. Our codes are available at https://github.com/alpc91/NICE-GAN-pytorch.*

## 1. Introduction

Image-to-Image translation transforming images from one domain to the other has boosted a variety of applications in vision tasks, from colorization [39], image editing [6], super-resolution [20] to video generation [35]. Given the extensive effort of collecting paired images between do-

*Corresponding author: Fuchun Sun.
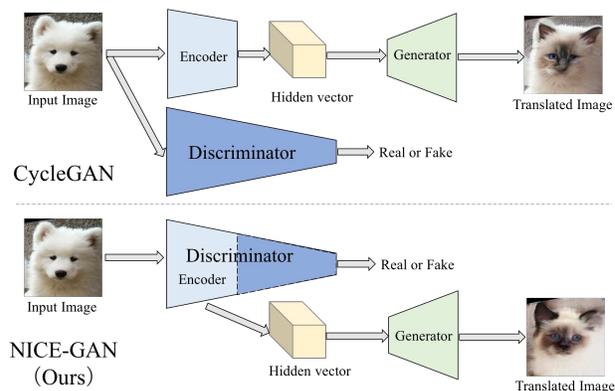


Figure 1: Illustrative difference between CycleGAN-alike methods and our NICE-GAN.

mains, a more practical line of research [40, 25, 12, 21, 16] directs the goal to unsupervised scenario where no paired information is characterized. Due to the non-identifiability problem [25] in unsupervised translation, various methods have been proposed to address this issue by using additional regulations including weight-coupling [25], cycle-consistency [40, 17, 38], forcing the generator to the identity function [34], or more commonly, combination of them.

When we revisit current successful translation frameworks (such as the one proposed by CycleGAN [40]), most of them consist of three components for each domain: an encoder to embed the input image to a low-dimension hidden space, a generator to translate hidden vectors to images of the other domain, and a discriminator for domain alignment by using GAN training [9]. While this piled-up way is standard, we are still interested in asking: *is there any possibility to rethink the role of each component in current translation frameworks?* and more importantly, *can we change the current formulation (for example, to a more compact architecture) based on our rethinking?*

The answer is yes, if we check the relation between the encoder and the discriminator. Basically, the discriminator is to distinguish between the translated image of the source domain and the real image of the target domain. To do so, the discriminator should conduct sort of semantics encoding of the input images before it can tell what images are true and what are false. This, in other words, contends the two roles of the discriminator: encoding and classifying. Indeed, the DCGAN paper [30] has revealed the encoding ability of the discriminator: strongly responses to the input image are observed in the first 6 learned convolutional features from the last convolution layer in the discriminator.

Upon the motivation mentioned above, this paper proposes to reuse the discriminator for encoding. In particular, we reuse early layers of certain number in the discriminator as the encoder of the target domain, as illustrated in Figure. 1. Such kind of reusing exhibits two-fold advantages: **I.** A more compact architecture is achieved. Since the encoder now becomes part of the discriminator, we no longer require an independent component for encoding. Also, unlike existing methods where the discriminator is abandoned after training, its encoding part is still kept for inference in our framework. **II.** The encoder is trained more effectively. Traditional training of the encoder is conducted by back-propagating the gradients from the generator, which is indirect. Here, by plugging it into the discriminator, the encoder is directly trained through the discriminative loss. Moreover, modern discriminators have resorted to the multi-scale scheme for more expressive power [8, 13, 7, 36]; our encoder will inherit the expressive ability by nature if the multi-scale discriminator is applied.

A remaining issue of our approach is how to perform adversary training. For traditional methods [40, 25, 12, 21, 16], the encoder is trained along with the generator for minimizing the GAN loss, while the discriminator is trained separately to maximize the objective. In our framework, the encoder and the discriminator become overlap, and it will bring in instability if we apply traditional training setting—the encoder as part of translation is trained for minimizing, and at the same time it belongs to the discriminator and is also trained for maximizing. To eliminate the inconsistency, we develop a decoupled training paradigm. Specifically, the training of the encoder is only associated with the discriminator, independent to the generator. Our experiments on several benchmarks show that such simple decoupling promotes the training remarkably (see details in Section 4.7). Another intuition behind is that disentangling the encoder from the training of translation will make it towards more general purpose of encoding other than translation along, thereby enabling more generality.

We summarize our contributions as follow.

- To the best of our knowledge, we are the first to reuse discriminators for encoding specifically for un-

supervised image-to-image translation. By such a reusing, a more compact and more effective architecture is derived, which is dubbed as No-Independent-Component-for-Encoding GAN (NICE-GAN).

- Given that the reusing of discriminator will incur instability in terms of typical training procedure, this paper develops a decoupled training paradigm, which is simple yet efficient.

- Extensive experimental evaluations on several popular benchmarks reveal that the proposed method outperforms various state-of-the-art counterparts. The comprehensive ablation studies are also conducted to verify the effectiveness of each proposed component.

## 2. Related Work

**Image-to-image translation.** Conditional GAN-based standard framework, proposed by Isola *et al.* [14] , promotes the study on image-to-image translation. Several works extend it to deal with super-resolution[36] or video generation[35]. Despite of the promising results they attain, all these approaches need paired data for training, which limits their practical usage.

**Unsupervised image-to-image translation.** In terms of unsupervised image-to-image translation with unpaired training data, CycleGAN [40], DiscoGAN [17], Dual-GAN [38] preserve key attributes between the input and the translated image by using a cycle-consistency loss. Various studies have been proposed towards extension of CycleGAN. The first kind of development is to enable multi-modal generations: MUNIT [12] and DRIT [21] decompose the latent space of images into a domain-invariant content space and a domain-specific style space to get diverse outputs. Another enhancement of CycleGAN is to perform translation across multiple (more than two) domains simultaneously, such as StarGAN [5]. A more funtional line of research focuses on transformation between domains with larger difference. For example, CoupledGAN [26], UNIT [25], ComboGAN [2] and XGAN [31] using domain-sharing latent space, and U-GAT-IT [16] resort to attention modules for feature selection. Recently, TransGAGA [37] and TravelGAN [1] are proposed to characterize the latent representation by using Cartesian product of geometry and preserving vector arithmetic, respectively.

**Introspective Networks.** Exploring the double roles of the discriminator has been conducted by Introspective Neural Networks (INN) [15, 19, 23] and Introspective Adversarial Networks (IAN) [4, 33]. Although INN does share the same purpose of reusing discriminator for generation, it exhibits several remarkable differences compared to our NICE-GAN. First, INN and NICE-GAN tackle different tasks. INN is for pure generation, and the discriminator is reused for generation from hidden vectors to images (as decoding); our NICE-GAN is for translation, and the discrim-

inator is reused for embedding from images to hidden vectors (as encoding). Furthermore, INN requires sequential training even when doing inference, while NICE-GAN only needs one forward pass to generate a novel image, depicting more efficiency. Regarding IAN, it is also for pure generation and reuses one discriminator to generate self-false samples, which is an introspective mechanism; our NICE-GAN reuses the discriminator of one domain to generate a false sample of the other, which is indeed a mutual introspective mechanism.

# 3. Our NICE-GAN

This section presents the detailed formulation of our method. We first introduce the general idea, and then follow it up by providing the details of each component in NICE-GAN. The decoupled training mechanism is specified as well.

## 3.1. General Formulation

**Problem Definition.** Let $\mathcal{X}$, $\mathcal{Y}$ be two image domains. While supervised image-to-image translation requires to learn the conditional mappings $f_{x \to y} = p(\mathcal{Y}|\mathcal{X})$ and $f_{y \to x} = p(\mathcal{X}|\mathcal{Y})$ given the joint distribution $p(\mathcal{X}, \mathcal{Y})$, unsupervised translation learns $f_{x \to y}$ and $f_{y \to x}$ with only the marginals $p(\mathcal{X})$ and $p(\mathcal{Y})$ provided. Unsupervised translation is ill-posed, since there are infinitely many conditional probabilities corresponded to the same marginal distributions. To address this issue, current methods resort to adding extra regulations, such as weight-coupling [26, 25, 21], cycle-consistency [40, 17, 38], and identity-mapping-enforcing [34, 40], the latter two of which are employed in this paper.

In most of existing frameworks, the translation $f_{x \to y}$ (resp. $f_{y \to x}$) is composed of an encoder $E_x$ (resp. $E_y$) and a generator $G_{x \to y}$ (resp. $G_{y \to x}$). By combining them all together, it gives $y' = f_{x \to y}(x) = G_{x \to y}(E_x(x))$ (resp. $x' = f_{y \to x}(y) = G_{y \to x}(E_y(y))$). The GAN [9] training fashion is usually adopted to enable the translated output to fit the distribution of the target domain. Namely, we use a discriminator $D_y$ (resp. $D_x$) to classify between the true image $y$ and the translated image $y'$ (resp. $x$ and $x'$).

**No Independent Component for Encoding (NICE).** As mentioned in introduction, our NICE-GAN reuses discriminators for encoding, delivering the advantages of efficiency and effectiveness for training. Formally, we divide the discriminator $D_y$ into the encoding part $E_y^D$ and classification part $C_y$. The encoding part $E_y^D$ will replace the original encoder in $f_{y \to x}$, resulting in a new translation $f_{y \to x}(y) = G_{y \to x}(E_y^D(y))$. Similarly for the discriminator $D_x$, we define $E_x^D$ and $C_x$, and reformulate the translation function as $f_{x \to y}(x) = G_{x \to y}(E_x^D(x))$. As for the classification components $C_x$ and $C_y$, we further employ the multi-scale structure to boost the expressive power. Besides, the

newly-formulated encoders $E_x^D$ and $E_y^D$ exist in the training loops of both translation and discrimination, making them difficult to train. Hence we proposed a decoupled training flowchart in NICE-GAN. The details of the architecture's build-up and training are presented in Section 3.2 and Section 3.3, respectively. Figure 2 illustrates our framework. Unless otherwise noticed, **we will remove the superscript $D$ from $E_x^D$ and $E_y^D$ for simplicity in what follows**.

## 3.2. Architecture

**Multi-Scale Discriminators $D_x$ and $D_y$.** We only discuss $D_x$ here, since the formulation of $D_y$ is similar. Full details are provided in the supplementary material (SP). Our usage of multi-scale discriminators is inspired from previous works [8, 13, 7, 36]. In these approaches, the discriminator of different scale is applied to the image of different size (the small-size images are attained from the original image by down-sampling). In this paper, we consider a more efficient way by regarding the feature maps in different layers of the single input to be the images of different scales, and then feed each of them to the classifier with the corresponding input size for discriminating, which is similar to the application of feature pyramid representations in object detection (*e.g.* SSD [27] and FPN [24]).

We now introduce our idea in a formal way. As mentioned above, the discriminator $D_x$ contains two parts: the encoder $E_x$ and the classifier $C_x$. To enable multi-scale processing, the classifier $C_x$ is further divided into three sub-classifiers: $C_x^0$ for local scale (10 x 10 receptive field), $C_x^1$ for middle scale (70 x 70 receptive field), and $C_x^2$ for global scale (286 x 286 receptive field). $C_x^0$ is directly connected to the output of $E_x$. Then, a down-sampling-convolution layer is conducted on $E_x$ to provide the feature maps of smaller scale, which are concatenated to two branches: one is linked to $C_x^1$, and the other one is further down sampled through convolution layers followed by $C_x^2$. For a single input image, $C_x^0$, $C_x^1$, and $C_x^2$ are all trained to predict whether the image is true of false. The multi-scale discriminator is also illustrated in Figure 2.

Besides the multi-scale design, we develop a residual attention mechanism to further facilitate the feature propagation in our discriminator. Using attention in discriminator is originally proposed by U-GAT-IT [16]. Suppose the encoder contains feature maps of number $K$ (*i.e.* $E_x = \{E_x^k\}_{k=1}^K$). The idea of U-GAT-IT is first learning an attention vector $w$, each element of which counts the importance of each feature map. Then, the attended features computed by $a(x) = w \times E_x(x) = \{w_k \times E_x^k(x)\}_{k=1}^K$ are leveraged for later classification. Upon but beyond U-GAT-IT, this paper further takes the residual connection into account, that is, we use $a(x) = \gamma \times w \times E_x(x) + E_x(x)$, where the trainable parameter $\gamma$ determines the trade-off between the attended features and the original ones. When $\gamma = 0$, it
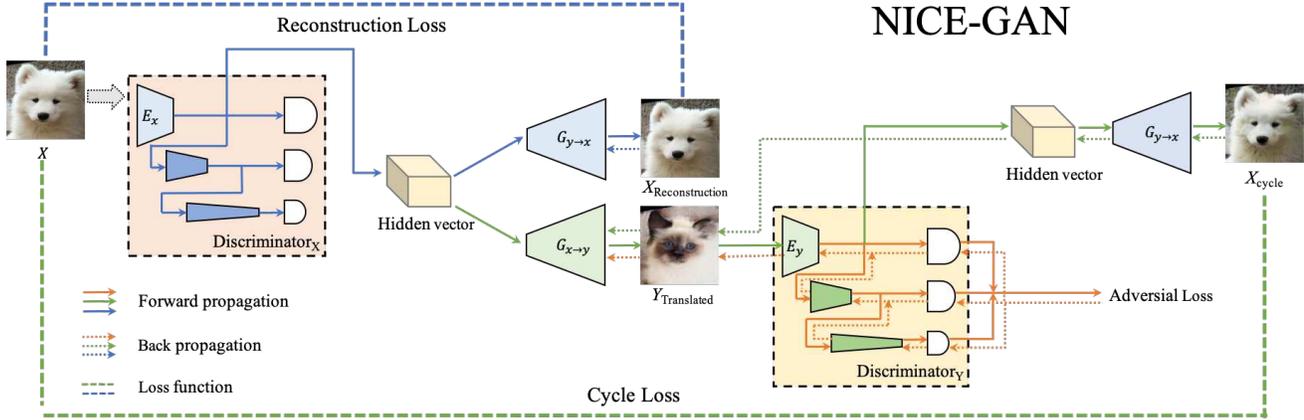
Figure 2: Illustration of the flowchart of NICE-GAN. Here we only display one translation stream from $\mathcal{X}$ to $\mathcal{Y}$ (from dog to cat). Note that we apply a decoupled training fashion: the encoder $E_y$ is fixed when minimizing the adversarial loss, the reconstruction loss and the cycle loss, and it is trained when maximizing the adversarial loss.

returns to $E_x(x)$ indicating no attention is used, and otherwise, the attention is activated. By this modification, our method becomes more flexible on adjusting the importance of different feature maps and thus attains more effectiveness in training, which will be clarified by our experiments.

**Generators $G_{x\rightarrow y}$ and $G_{y\rightarrow x}$.** Both $G_{x\rightarrow y}$ and $G_{y\rightarrow x}$ are composed of six residual blocks [10], and two sub-pixel convolutional layers for up-sampling [32]. And, we use the AdaLIN light version similar to the paper [16]. In addition, spectral normalization [29] used for the discriminator and cycle-consistency loss is conducted to prevent generators from mode collapse. Full details are presented in the SP.

### 3.3. Decoupled Training

The training process is proceeded in terms of three kinds of losses: adversarial loss, identity reconstruction loss, and cycle-consistency loss. The adversarial loss is to pursue domain transfer, while both reconstruction loss and cycle-consistency loss are for tackling the non-identifiability issue as pointed out before.

Since the encoder $E_x$ is not only a part of the discriminator $D_x$ but is also taken as the input of the generator $G_{x\rightarrow y}$, it will incur inconsistency if we apply conventional adversarial training. To overcome this defect, we decouple the training of $E_x$ from that of the generator $G_{x\rightarrow y}$. The details of formulating each loss are provided below.

**Adversarial loss.** First, we make use of the least-square adversarial loss by [28] for its more stable training and higher quality of generation. The min-max game is conducted by

$$\min_{G_{x\rightarrow y}} \max_{D_y=(C_y\circ E_y)} L_{gan}^{x\rightarrow y} := \mathbb{E}_{y\sim\mathcal{Y}}\left[(D_y(y))^2\right]$$
$$+\mathbb{E}_{x\sim\mathcal{X}}\left[(1-D_y(G_{x\rightarrow y}(E_x(x))))^2\right], \quad (1)$$

where, $E_x$ is fixed and $E_y$ is trained when maximizing

$L_{gan}^{x\rightarrow y}$, and both of them are fixed when minimizing $L_{gan}^{x\rightarrow y}$.

**Cycle-consistency loss.** The cycle-consistency loss is first introduced by CycleGAN [40] and DiscoGAN [17], which is to force the generators to be each other's inverse.

$$\min_{\substack{G_{x\rightarrow y}\\G_{y\rightarrow x}}} L_{cycle}^{x\rightarrow y} := \mathbb{E}_{x\sim\mathcal{X}}\left[|x-G_{y\rightarrow x}(E_y(G_{x\rightarrow y}(E_x(x))))|_1\right],$$
$$(2)$$

where $|\cdot|_1$ computes the $\ell_1$ norm, and both $E_x$ and $E_y$ are also frozen.

**Reconstruction loss.** Forcing the generator to be close to the identity function is another crucial regulation technique in CycleGAN [40]. Unlike CycleGAN where the identity loss is based on domain similarity assumption, our reconstruction is based on the shared-latent space assumption. Reconstruction loss is to regularize the translation to be near an identity mapping when real samples' hidden vectors of the source domain are provided as the input to the generator of the source domain. Namely,

$$\min_{G_{y\rightarrow x}} L_{recon}^{x\rightarrow y} := \mathbb{E}_{x\sim\mathcal{X}}\left[|x-G_{y\rightarrow x}(E_x(x))|_1\right], \quad (3)$$

where $E_x$ is still kept unchanged.

Similarly, we can define the losses from domain $\mathcal{Y}$ to $\mathcal{X}$: $L_{gan}^{y\rightarrow x}$, $L_{cycle}^{y\rightarrow x}$, and $L_{recon}^{y\rightarrow x}$.

**Full objective.** The discriminators' final objective is

$$\max_{E_x,C_x,E_y,C_y} \lambda_1 L_{gan}; \quad (4)$$

while the generators' final loss objective is

$$\min_{G_{x\rightarrow y},G_{y\rightarrow x}} \lambda_1 L_{gan} + \lambda_2 L_{cycle} + \lambda_3 L_{recon},$$

where, $L_{gan} = L_{gan}^{x\rightarrow y} + L_{gan}^{y\rightarrow x}, L_{cycle} = L_{cycle}^{x\rightarrow y} + L_{cycle}^{y\rightarrow x}, L_{recon} = L_{recon}^{x\rightarrow y} + L_{recon}^{y\rightarrow x}$, and $\lambda_1$, $\lambda_2$, and $\lambda_3$ are
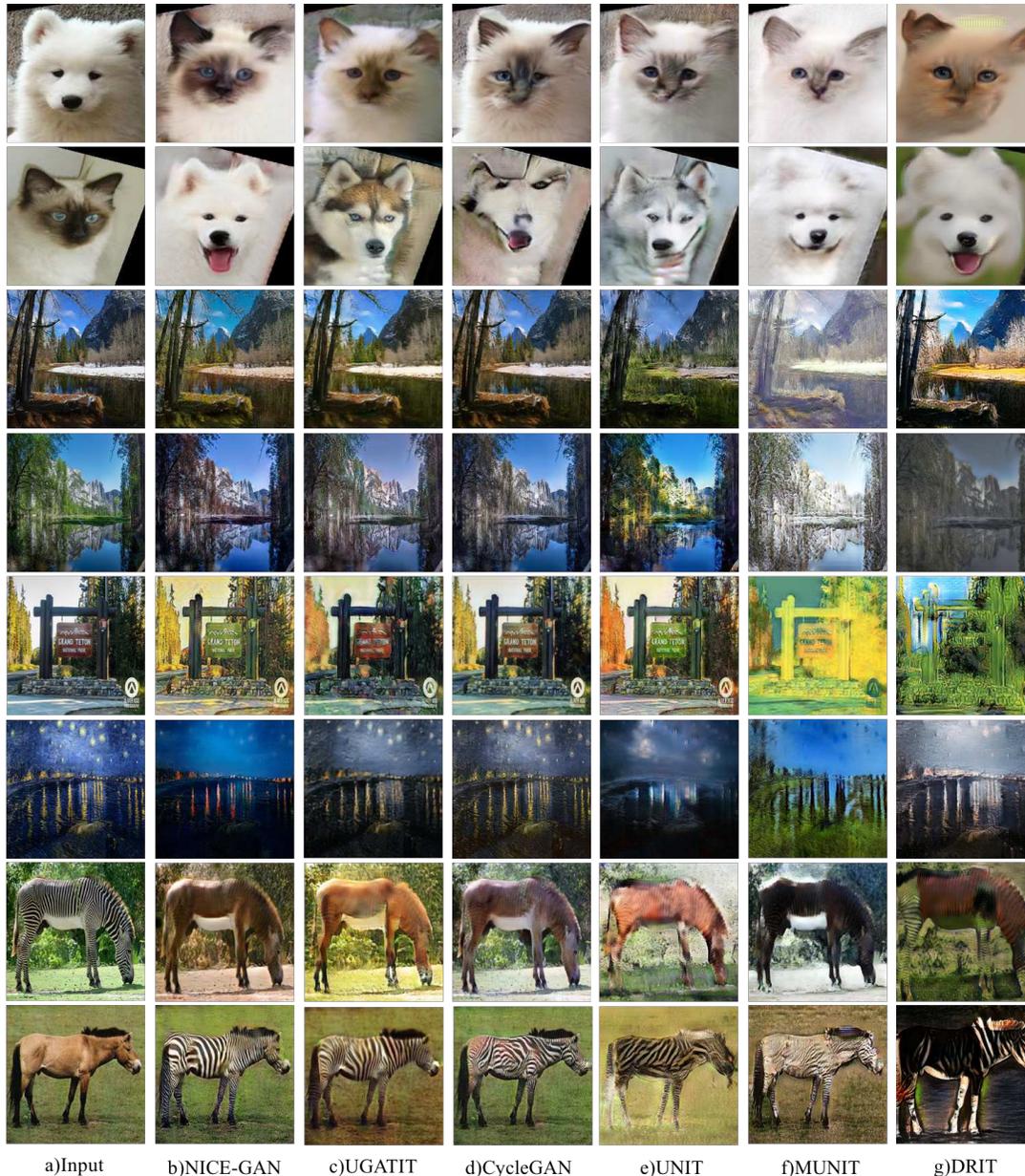
Figure 3: **Examples of generated outputs.** From top to bottom: dog↔cat, winter↔summer, photo↔vangogh, and zebra↔horse.

a)Input    b)NICE-GAN    c)UGATIT    d)CycleGAN    e)UNIT    f)MUNIT    g)DRIT

the trade-off weights (they are fixed as $\lambda_1 = 1$, $\lambda_2 = 10$, and $\lambda_3 = 10$ throughout our experiments).

Note again, the encoders $E_x$ and $E_y$ are trained under the discriminator's objective but are decoupled from the training of generators. The benefit of the proposed decoupled training paradigm will be analyzed by our experiments.

## 4. Experiments

### 4.1. Baselines

We compare the performance NICE-GAN with state-of-the-art methods including CycleGAN [40], UNIT [25],

MUNIT [12], DRIT [21], and U-GAT-IT [16] considering their competitive performance on unsupervised image-to-image translation. All compared methods are conducted by using the public codes. Specifically for U-GAT-IT, we use its light version due to the memory limit of our GPU machine. The details of all baselines are introduced in the SP.

### 4.2. Dataset

The experiments are carried out on four popular benchmarks of unpaired images: **horse↔zebra**, **summer↔winter_yosemite**, **vangogh↔photo** and **cat↔dog**. The first three datasets are used in Cy-

Table 1: The FID and the KID ×100 for different algorithms. Lower is better. All of the methods are trained to the 100K-th iterations. NICE-GAN* is the version that the generator network is composed of only four residual blocks.

| Dataset | dog → cat | | winter → summer | | photo → vangogh | | zebra → horse | |
| Method | FID | KID × 100 | FID | KID × 100 | FID | KID × 100 | FID | KID × 100 |
|---|---|---|---|---|---|---|---|---|
| NICE-GAN | **48.79** | 1.58 | **76.44** | **1.22** | **122.27** | 3.71 | 149.48 | 4.29 |
| NICE-GAN* | 51.98 | 1.50 | 79.02 | 1.35 | 122.59 | **3.53** | 150.57 | 4.43 |
| U-GAT-IT-light | 80.75 | 3.22 | 80.33 | 1.82 | 137.70 | 6.03 | **145.47** | **3.39** |
| CycleGAN | 119.32 | 4.93 | 79.58 | 1.36 | 136.97 | 4.75 | 156.19 | 5.54 |
| UNIT | 59.56 | 1.94 | 95.93 | 4.63 | 136.80 | 5.17 | 170.76 | 6.30 |
| MUNIT | 53.25 | **1.26** | 99.14 | 4.66 | 130.55 | 4.50 | 193.43 | 7.25 |
| DRIT | 94.50 | 5.20 | 78.61 | 1.69 | 136.24 | 5.43 | 200.41 | 10.12 |

| Dataset | cat → dog | | summer → winter | | vangogh → photo | | horse → zebra | |
| Method | FID | KID × 100 | FID | KID × 100 | FID | KID × 100 | FID | KID × 100 |
|---|---|---|---|---|---|---|---|---|
| NICE-GAN | **44.67** | **1.20** | **76.03** | **0.67** | **112.00** | **2.79** | **65.93** | **2.09** |
| NICE-GAN* | 55.72 | 1.89 | 77.13 | 0.73 | 117.81 | 3.61 | 84.89 | 3.29 |
| U-GAT-IT-light | 64.36 | 2.49 | 88.41 | 1.43 | 123.57 | 4.91 | 113.44 | 5.13 |
| CycleGAN | 125.30 | 6.93 | 78.76 | 0.78 | 135.01 | 4.71 | 95.98 | 3.24 |
| UNIT | 63.78 | 1.94 | 112.07 | 5.36 | 143.96 | 7.44 | 131.04 | 7.19 |
| MUNIT | 60.84 | 2.42 | 114.08 | 5.27 | 138.86 | 6.19 | 128.70 | 6.92 |
| DRIT | 79.57 | 4.57 | 81.64 | 1.27 | 142.69 | 5.62 | 111.63 | 7.40 |

Table 2: Total number of parameters and FLOPs of network modules. NICE-GAN* are the version that the generator network is composed of only four residual blocks.

| Module | Total number of params(FLOPs) | |
| Method | Generators | Discriminators |
|---|---|---|
| U-GAT-IT-light | 21.2M(105.0G) | 112.8M(15.8G) |
| NICE-GAN | 16.2M(67.6G) | 93.7M(12.0G) |
| NICE-GAN* | 11.5M(48.2G) | 93.7M(12.0G) |

cleGAN [40], whose train-test splits are respectively: 1,067/120 (horse), 1,334/140 (zebra); 1,231/309 (summer); 962/238 (winter); 400/400 (vangogh), 6,287/751 (photo). The last dataset is studied in DRIT [21][22], whose train-test splits are: 771/100 (cat), 1,264/100 (dog). All images of all datasets are cropped and resized to 256 × 256 for training and testing.

### 4.3. Evaluation Metrics

**Human Preference.** To compare the veracity of translation outputs generated by different methods, we carry out human perceptual study. Similar to Wang *et al*. [36], volunteers are shown an input image and three translation outputs from different methods, and given unlimited time they select which translation output looks better.

**The Fréchet Inception Distance (FID)** proposed by Heusel *et al*. (2017) [11] contrast the statistics of generated samples against real samples. The FID fits a Gaussian distribution to the hidden activations of InceptionNet for each compared image set and then computes the Fréchet distance (also known as the Wasserstein-2 distance) between those Gaussians. Lower FID is better, corresponding to generated images more similar to the real.
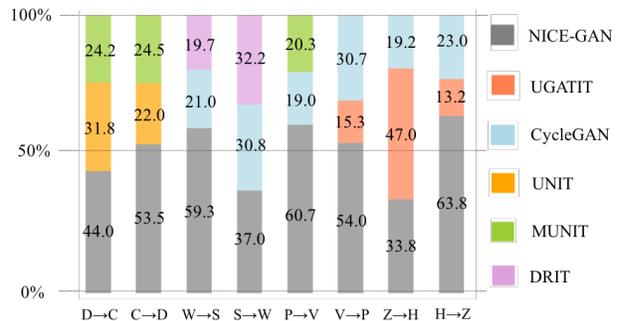


Figure 4: **Human preference results.** The number indicates the percentage of preference on that translation task. Abbreviation: (D)og, (C)at; (W)inter, (S)ummer; (P)hoto, (V)angogh; (Z)ebra, (H)orse.

**The Kernel Inception Distance (KID)** developed by [3] is a metric similar to the FID but uses the squared Maximum Mean Discrepancy(MMD) between Inception representations with a polynomial kernel, $k(x, y) = \left(\frac{1}{d}x^{\mathrm{T}}y + 1\right)^3$, where $d$ is the representation dimension. It can also be viewed as an MMD directly on input images with the kernel $K(x, y) = k(\theta(x), \theta(y))$, where $\theta$ is the function mapping images to Inception representations. Unlike FID, KID has a simple unbiased estimator, making it more reliable especially when there are much more inception features channels than image numbers. Lower KID indicates more visual similarity between real and generated images. Our implementation of KID is based on https://github.com/mbinkowski/MMD-GAN where the hidden representations are from the Inception-v3 pool3 layer.

## 4.4. Setup

We use ReLU as the actionvation function in the generator and leaky-ReLU with a slope of 0.2 in the discriminator. We train all models using the Adam [18] optimizer with the learning rate 0.0001 and $(\beta 1, \beta 2) = (0.5, 0.999)$ on NVIDIA RTX 2080Ti GPUs. For data augmentation, we flip the images horizontally with a probability of 0.5, resized them to $286 \times 286$, and randomly cropped them to $256 \times 256$. The batch size is set to 1 for all experiments. We also use a weight decay at the rate of 0.0001. All models are trained over 100K iterations. More details on the training process and network architecture are provided in the SP.

## 4.5. Comparisons with state of the arts

Table 1 shows that our approach generally achieves the lowest FID or KID scores on all cases except zebra→horse, indicating the promising translation ability of our NICE framework on varying tasks. These two metrics maintain good consistency in relative scores, which fully demonstrates our NICE-GAN reasonably performs well regardless of what measure we have used. By contrast, other methods only perform well on certain datasets; for instance, U-GAT-IT-light, UNIT and MUNIT successfully transforms the semantic of objects(*e.g.* animal faces), while CycleGAN is good at modifying low-level characteristics (*e.g.* colors and textures). U-GAT-IT-light roughly shares the same structures (multi-scale discriminators and generators) as NICE-GAN, and it differs from NICE-GAN mainly in its independent formulation of encoders. Table 2 reports total number of parameters and FLOPs of U-GAT-IT-light and NICE-GAN, and it reads that our architecture is more compact by reusing discriminators for encoding. To further observe the visual difference, Figure 3 depicts the translated images of different methods on test sets. The generated images by NICE-GAN are almost more meaningful and have less artifacts than others (see the cat↔dog task for an example).

In addition to our method, we select two baselines achieving lowest KID scores in each dataset to conduct a human perceptual study. Firstly, volunteers are shown an example pair consisting of source-domain image and a target-domain image as a reference to better understand what style is translating. Secondly, they are given an input image, and three translated outputs among which one is from NICE-GAN and the other two from the selected baselines. They have unlimited time to choose which looks most real based on perceptual realism. The synthesized images are displayed in a randomized order to ensure fair comparisons. Besides, checkpoint questions are set and distributed to each volunteer to validating human effort. A total of 123 questionnaires are collected in which we find that 120 are valid. Figure 4 shows that NICE-GAN wins the majority of votes in all cases except for zebra→horse. These results

Table 3: **Ablation Study.** Results of methods are all in 100K iterations of discriminator. NICE: No Independent Component for Encoding; RA: add residual connection in CAM attention module; $C_x^0$ for local scale (10 x 10 receptive field), $C_x^1$ for middle scale (70 x 70 receptive field), and $C_x^2$ for global scale (286 x 286 receptive field); $-$: decreasing the number of shared layers by 1; $+$: increasing by 1.

| Data Set | Components | | | | | FID | KID $\times$ 100 |
|---|---|---|---|---|---|---|---|
| | NICE | RA | $C_x^0$ | $C_x^1$ | $C_x^2$ | | |
| dog $\rightarrow$ cat | $\times$ | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 80.75 | 3.22 |
| | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 67.60 | 2.94 |
| | $\checkmark$ | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 63.80 | 3.27 |
| | $-$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | **48.55** | **1.23** |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 48.79 | 1.58 |
| | $+$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 53.52 | 1.84 |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$ | 203.56 | 15.27 |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$ | $\times$ | 216.03 | 18.57 |
| cat $\rightarrow$ dog | $\times$ | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 64.36 | 2.49 |
| | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 64.62 | 2.41 |
| | $\checkmark$ | $\times$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 51.49 | 1.68 |
| | $-$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 52.92 | 1.82 |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | **44.67** | **1.20** |
| | $+$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 54.90 | 2.17 |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$ | 238.62 | 21.41 |
| | $\checkmark$ | $\checkmark$ | $\checkmark$ | $\times$ | $\times$ | 231.24 | 22.12 |

are also consistent with the quantitative metrics in Table 1. More examples of the results from our model are included in the SP.

## 4.6. Ablation study

We conduct ablation studies on the cat↔dog datasets in Table 3 to isolate the validity of the key components of our method: the NICE strategy, the multi-scale formulation and the Residual Attention (RA) mechanism in discriminators. We perform four groups of experiments.

**NICE and RA.** The first group keeps employing the multi-scale formulation but removes either or both of NICE and RA to draw the performance difference. The results are reported in Table 3. It verifies that each of NICE and RA contributes to the performance improvement while the importance of NICE is more significant. Overall, by combing all components, NICE-GAN remarkably outperforms all other variants. Figure 5 shows the latent vectors of each domain w/ and w/o NICE on *cat ↔dog* via t-SNE, as well as MMD to compute domain difference. Interestingly, with NICE training, the latent distributions of two domains become more clustered and closer, yet separable to each other. Such phenomenon explains why our NICE-GAN performs promisingly. By shortening the transition path between domains in the latent space, NICE-GAN built upon the shared latent space assumption can probably facilitate domain translation in the image space.

**The number of shared layers.** For consistent comparison,
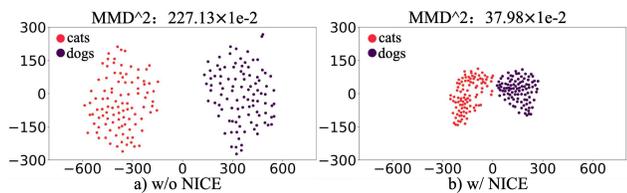
Figure 5: The t-SNE visualization of the latent vectors, as well as the MMD to measure domain difference.

we employ the commonly-used ResNet backbone in [40] as our generator and naturally share the whole encoder therein. We also evaluate the effect of changing the number of layers shared by discriminator and encoder. Table 3 shows that whether decreasing or increasing the number of layers generally hinders the performance. Thus, the best choice here is sharing the whole default encoder.

**Multi-scale.** The third group of experiments is to evaluate the impact of the multi-scale formulation ($C_x^0$, $C_x^1$, $C_x^2$) in discriminators. The results are summarized in Table 3. We find that removing $C_x^2$ will cause a serious detriment; the importance of $C_x^0$ and $C_x^1$ is task-dependent, and adding $C_x^1$ upon $C_x^0$ does not exhibit clear enhancement on this task. Actually, all three scales are generally necessary and multi-scale is more robust, with more discussions in the SP.

**Weight-coupling.** Besides, there are existing methods of model compression, such as weight-coupling [25, 21]. Sharing a few layers between two generators enables model compression but detriments the translation performance. For example on $cat \leftrightarrow dog$, the FID increases from 48.79/44.67 to 49.67/56.32 if sharing the first layer of the decoders, and from 48.79/44.67 to 55.00/55.60 if sharing the last layer of encoders. Similar results are observed if we reuse the first layer of the classifiers, the FID increases from 48.79/44.67 to 61.73/46.65 . It implies weight-coupling could weaken the translation power for each domain.

### 4.7. Decoupled Training Analysis

In our NICE framework, we decouple the training of $E_x$ from that of the generator $G_{x \rightarrow y}$. To prove the effectiveness of this strategy, we develop two additional variants: NICE-GAN-1 and NICE-GAN-2. To be specific, NICE-GAN-1 adopts the conventional training approach, where the encoders are jointly trained with the discriminators and generators. As for NICE-GAN-2, it is also performed in a decoupled way but acts oppositely with our method, that is, the encoders are trained along with the generator, independent to the classifiers in the discriminators. From a more essential sense, the discriminators indeed degenerate to the classifiers in NICE-GAN-2.

Figure 6 reports the training curves of NICE-GAN, NICE-GAN-1 and NICE-GAN-2. Clearly, the training of NICE-GAN-1 is unstable, which is consistent with our analysis. NICE-GAN-2 performs more stably and better than NICE-GAN-1, but is still inferior to our NICE-GAN. We conjecture that in NICE-GAN-2, using the classifiers for
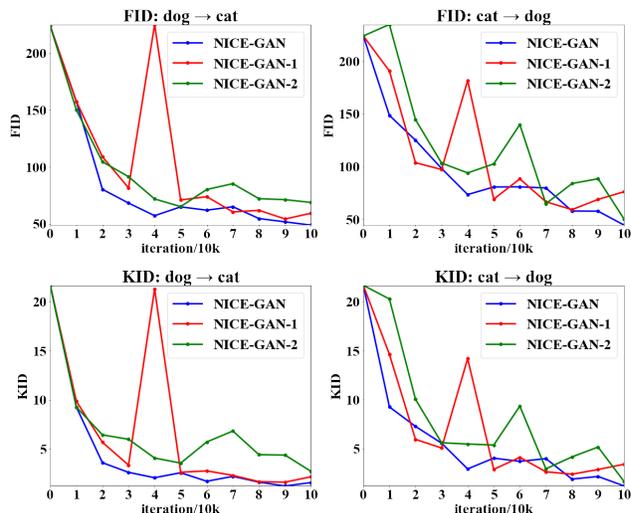


Figure 6: **Decoupled Training Analysis.** NICE-GAN: decoupled training, $E_x$ and $E_y$ will only be updated by $maxL_{gan}$; NICE-GAN-1: jointly train the discriminators and generators, $E_x$ and $E_y$ will be updated by $minmaxL_{gan}$, $minL_{cycle}$ and $minL_{recon}$ ; NICE-GAN-2: decoupled training, $E_x$ and $E_y$ will be updated by $minL_{gan}$, $minL_{cycle}$ and $minL_{recon}$.

discriminating is actually aligning the distributions of hidden vectors. Nevertheless, NICE-GAN leverages both the encoders and classifiers for discriminating, underlying that it is matching the distributions of image space, thus more precise information is captured.

A clear disentanglement of responsibilities of different components makes NICE-GAN simple and effective. Besides, It further supports the idea [4] that features learned by a discriminatively trained network tend to be more expressive than those learned by an encoder network trained via maximum likelihood, and thus better suited for inference.

## 5. Conclusion

In this paper, we present NICE-GAN, a novel framework for unsupervised image-to-image translation. It reuses discriminators for encoding and develops a decoupled paradigm for efficient training. Comparable experimental evaluations on several popular benchmarks reveal that NICE-GAN generally achieves superior performance over state-of-the-art methods. Our research is expected to evoke the rethinking on what discriminators actually can do, and it is potentially applicable to refresh the GAN-based models in other cases.

# References

[1] Matthew Amodio and Smita Krishnaswamy. Travelgan: Image-to-image translation by transformation vector learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8983–8992, 2019.

[2] Asha Anoosheh, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Combogan: Unrestrained scalability for image domain translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 783–790, 2018.

[3] M Bińkowski, DJ Sutherland, M Arbel, and A Gretton. Demystifying mmd gans. In *International Conference on Learning Representations*, 2018.

[4] Andrew Brock, Theodore Lim, James Millar Ritchie, and Nicholas J Weston. Neural photo editing with introspective adversarial networks. In *5th International Conference on Learning Representations*, 2017.

[5] 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. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018.

[6] Tali Dekel, Chuang Gan, Dilip Krishnan, Ce Liu, and William T Freeman. Sparse, smart contours to represent and edit images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3511–3520, 2018.

[7] Ugur Demir and Gozde Unal. Patch-based image inpainting with generative adversarial networks. *arXiv preprint arXiv:1803.07422*, 2018.

[8] Ishan Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. In *International Conference on Learning Representations*, 2017.

[9] 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.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] 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. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017.

[12] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018.

[13] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):107, 2017.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 2017.

[15] Long Jin, Justin Lazarow, and Zhuowen Tu. Introspective classification with convolutional nets. In *Advances in Neural Information Processing Systems*, pages 823–833, 2017.

[16] Junho Kim, Minjae Kim, Hyeonwoo Kang, and Kwang Hee Lee. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. In *International Conference on Learning Representations*, 2020.

[17] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1857–1865. JMLR. org, 2017.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Justin Lazarow, Long Jin, and Zhuowen Tu. Introspective neural networks for generative modeling. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2774–2783, 2017.

[20] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[21] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *European Conference on Computer Vision*, 2018.

[22] Hsin-Ying Lee, Hung-Yu Tseng, Qi Mao, Jia-Bin Huang, Yu-Ding Lu, Maneesh Kumar Singh, and Ming-Hsuan Yang. Drit++: Diverse image-to-image translation viadisentangled representations. *arXiv preprint arXiv:1905.01270*, 2019.

[23] Kwonjoon Lee, Weijian Xu, Fan Fan, and Zhuowen Tu. Wasserstein introspective neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3702–3711, 2018.

[24] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[25] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in neural information processing systems*, pages 700–708, 2017.

[26] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *Advances in neural information processing systems*, pages 469–477, 2016.

[27] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[28] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative

adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2017.

[29] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[30] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[31] Amélie Royer, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Mosseri, Forrester Cole, and Kevin Murphy. Xgan: Unsupervised image-to-image translation for many-to-many mappings. *arXiv preprint arXiv:1711.05139*, 2017.

[32] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.

[33] Jianlin Su. O-gan: Extremely concise approach for auto-encoding generative adversarial networks, 2019.

[34] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representations*, 2016.

[35] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Advances in Neural Information Processing Systems*, pages 1144–1156, 2018.

[36] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[37] Wayne Wu, Kaidi Cao, Cheng Li, Chen Qian, and Chen Change Loy. Transgaga: Geometry-aware unsupervised image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8012–8021, 2019.

[38] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017.

[39] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016.

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.