Multimodal Unsupervised Image-to-Image Translation

Xun Huang[1], Ming-Yu Liu[2], Serge Belongie[1], Jan Kautz[2]

Cornell University[1] NVIDIA[2]

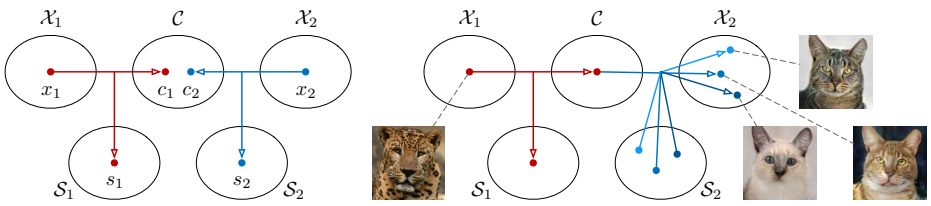
Abstract. Unsupervised image-to-image translation is an important and challenging problem in computer vision. Given an image in the source domain, the goal is to learn the conditional distribution of corresponding images in the target domain, without seeing any examples of corresponding image pairs. While this conditional distribution is inherently multimodal, existing approaches make an overly simplified assumption, modeling it as a deterministic one-to-one mapping. As a result, they fail to generate diverse outputs from a given source domain image. To address this limitation, we propose a Multimodal Unsupervised Image-to-image Translation (MUNIT) framework. We assume that the image representation can be decomposed into a content code that is domain-invariant, and a style code that captures domain-specific properties. To translate an image to another domain, we recombine its content code with a random style code sampled from the style space of the target domain. We analyze the proposed framework and establish several theoretical results. Extensive experiments with comparisons to state-of-the-art approaches further demonstrate the advantage of the proposed framework. Moreover, our framework allows users to control the style of translation outputs by providing an example style image. Code and pretrained models are available at https://github.com/nvlabs/MUNIT.

Keywords: GANs, image-to-image translation, style transfer

1 Introduction

Many problems in computer vision aim at translating images from one domain to another, including super-resolution [1], colorization [2], inpainting [3], attribute transfer [4], and style transfer [5]. This cross-domain image-to-image translation setting has therefore received significant attention [6–25]. When the dataset contains paired examples, this problem can be approached by a conditional generative model [6] or a simple regression model [13]. In this work, we focus on the much more challenging setting when such supervision is unavailable.

In many scenarios, the cross-domain mapping of interest is multimodal. For example, a winter scene could have many possible appearances during summer due to weather, timing, lighting, etc. Unfortunately, existing techniques usually assume a deterministic [8–10] or unimodal [15] mapping. As a result, they fail to capture the full distribution of possible outputs. Even if the model is made stochastic by injecting noise, the network usually learns to ignore it [6, 26].



(a) Auto-encoding (b) Translation

Fig. 1. An illustration of our method. (a) Images in each domain $\mathcal{X}_i$ are encoded to a shared content space $\mathcal{C}$ and a domain-specific style space $\mathcal{S}_i$. Each encoder has an inverse decoder omitted from this figure. (b) To translate an image in $\mathcal{X}_1$ (*e.g.*, a leopard) to $\mathcal{X}_2$ (*e.g.*, domestic cats), we recombine the content code of the input with a random style code in the target style space. Different style codes lead to different outputs.

In this paper, we propose a principled framework for the Multimodal UNsupervised Image-to-image Translation (MUNIT) problem. As shown in Fig. 1 (a), our framework makes several assumptions. We first assume that the latent space of images can be decomposed into a content space and a style space. We further assume that images in different domains share a common content space but not the style space. To translate an image to the target domain, we recombine its content code with a random style code in the target style space (Fig. 1 (b)). The content code encodes the information that should be preserved during translation, while the style code represents remaining variations that are not contained in the input image. By sampling different style codes, our model is able to produce diverse and multimodal outputs. Extensive experiments demonstrate the effectiveness of our method in modeling multimodal output distributions and its superior image quality compared with state-of-the-art approaches. Moreover, the decomposition of content and style spaces allows our framework to perform example-guided image translation, in which the style of the translation outputs are controlled by a user-provided example image in the target domain.

2 Related Works

Generative adversarial networks (GANs). The GAN framework [27] has achieved impressive results in image generation. In GAN training, a generator is trained to fool a discriminator which in turn tries to distinguish between generated samples and real samples. Various improvements to GANs have been proposed, such as multi-stage generation [28–33], better training objectives [34–39], and combination with auto-encoders [40–44]. In this work, we employ GANs to align the distribution of translated images with real images in the target domain.
Image-to-image translation. Isola *et al.* [6] propose the first unified framework for image-to-image translation based on conditional GANs, which has been extended to generating high-resolution images by Wang *et al.* [20]. Recent studies have also attempted to learn image translation without supervision. This

problem is inherently ill-posed and requires additional constraints. Some works enforce the translation to preserve certain properties of the source domain data, such as pixel values [21], pixel gradients [22], semantic features [10], class labels [22], or pairwise sample distances [16]. Another popular constraint is the cycle consistency loss [7–9]. It enforces that if we translate an image to the target domain and back, we should obtain the original image. In addition, Liu *et al.* [15] propose the UNIT framework, which assumes a shared latent space such that corresponding images in two domains are mapped to the same latent code.

A significant limitation of most existing image-to-image translation methods is the lack of diversity in the translated outputs. To tackle this problem, some works propose to simultaneously generate multiple outputs given the same input and encourage them to be different [13, 45, 46]. Still, these methods can only generate a discrete number of outputs. Zhu *et al.* [11] propose a BicycleGAN that can model continuous and multimodal distributions. However, all the aforementioned methods require pair supervision, while our method does not. A couple of concurrent works also recognize this limitation and propose extensions of CycleGAN/UNIT for multimodal mapping [47]/[48].

Our problem has some connections with multi-domain image-to-image translation [19, 49, 50]. Specifically, when we know how many modes each domain has and the mode each sample belongs to, it is possible to treat each mode as a separate domain and use multi-domain image-to-image translation techniques to learn a mapping between each pair of modes, thus achieving multimodal translation. However, in general we do not assume such information is available. Also, our stochastic model can represent continuous output distributions, while [19, 49, 50] still use a deterministic model for each pair of domains.

Style transfer. Style transfer aims at modifying the style of an image while preserving its content, which is closely related to image-to-image translation. Here, we make a distinction between example-guided style transfer, in which the target style comes from a single example, and collection style transfer, in which the target style is defined by a collection of images. Classical style transfer approaches [5, 51–56] typically tackle the former problem, whereas image-to-image translation methods have been demonstrated to perform well in the latter [8]. We will show that our model is able to address both problems, thanks to its disentangled representation of content and style.

Learning disentangled representations. Our work draws inspiration from recent works on disentangled representation learning. For example, InfoGAN [57] and β-VAE [58] have been proposed to learn disentangled representations without supervision. Some other works [59–66] focus on disentangling content from style. Although it is difficult to define content/style and different works use different definitions, we refer to "content" as the underling spatial structure and "style" as the rendering of the structure. In our setting, we have two domains that share the same content distribution but have different style distributions.

3 Multimodal Unsupervised Image-to-image Translation

Assumptions Let $x_1 \in \mathcal{X}_1$ and $x_2 \in \mathcal{X}_2$ be images from two different image domains. In the unsupervised image-to-image translation setting, we are given samples drawn from two marginal distributions $p(x_1)$ and $p(x_2)$, without access to the joint distribution $p(x_1, x_2)$. Our goal is to estimate the two conditionals $p(x_2|x_1)$ and $p(x_1|x_2)$ with learned image-to-image translation models $p(x_{1\to2}|x_1)$ and $p(x_{2\to1}|x_2)$, where $x_{1\to2}$ is a sample produced by translating x_1 to $\mathcal{X}_2$ (similar for $x_{2\to1}$). In general, $p(x_2|x_1)$ and $p(x_1|x_2)$ are complex and multimodal distributions, in which case a deterministic translation model does not work well.

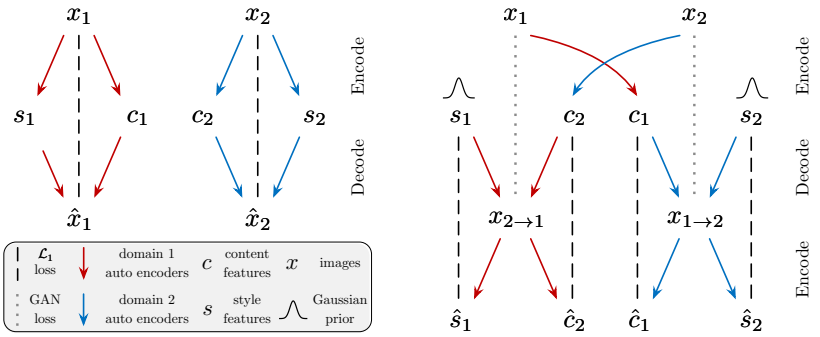
To tackle this problem, we make a *partially shared latent space assumption*. Specifically, we assume that each image $x_i \in \mathcal{X}_i$ is generated from a content latent code $c \in \mathcal{C}$ that is shared by both domains, and a style latent code $s_i \in \mathcal{S}_i$ that is specific to the individual domain. In other words, a pair of corresponding images (x_1, x_2) from the joint distribution is generated by $x_1 = G_1^*(c, s_1)$ and $x_2 = G_2^*(c, s_2)$, where c, s_1, s_2 are from some prior distributions and G_1^*, G_2^* are the underlying generators. We further assume that G_1^* and G_2^* are deterministic functions and have their inverse encoders $E_1^* = (G_1^*)^{-1}$ and $E_2^* = (G_2^*)^{-1}$. Our goal is to learn the underlying generator and encoder functions with neural networks. Note that although the encoders and decoders are deterministic, $p(x_2|x_1)$ is a continuous distribution due to the dependency of s_2.

Our assumption is closely related to the shared latent space assumption proposed in UNIT [15]. While UNIT assumes a fully shared latent space, we postulate that only part of the latent space (the content) can be shared across domains whereas the other part (the style) is domain specific, which is a more reasonable assumption when the cross-domain mapping is *many-to-many*.

Model Fig. 2 shows an overview of our model and its learning process. Similar to Liu *et al.* [15], our translation model consists of an encoder E_i and a decoder G_i for each domain $\mathcal{X}_i$ ($i = 1, 2$). As shown in Fig. 2 (a), the latent code of each auto-encoder is factorized into a content code c_i and a style code s_i, where $(c_i, s_i) = (E_i^c(x_i), E_i^s(x_i)) = E_i(x_i)$. Image-to-image translation is performed by swapping encoder-decoder pairs, as illustrated in Fig. 2 (b). For example, to translate an image $x_1 \in \mathcal{X}_1$ to $\mathcal{X}_2$, we first extract its content latent code $c_1 = E_1^c(x_1)$ and randomly draw a style latent code s_2 from the prior distribution $q(s_2) \sim \mathcal{N}(0, \mathbf{I})$. We then use G_2 to produce the final output image $x_{1\to2} = G_2(c_1, s_2)$. We note that although the prior distribution is unimodal, the output image distribution can be multimodal thanks to the nonlinearity of the decoder.

Our loss function comprises a *bidirectional reconstruction loss* that ensures the encoders and decoders are inverses, and an *adversarial loss* that matches the distribution of translated images to the image distribution in the target domain.

Bidirectional reconstruction loss. To learn pairs of encoder and decoder that are inverses of each other, we use objective functions that encourage reconstruction in both image → latent → image and latent → image → latent directions:



(a) Within-domain reconstruction (b) Cross-domain translation

Fig. 2. Model overview. Our image-to-image translation model consists of two auto-encoders (denoted by red and blue arrows respectively), one for each domain. The latent code of each auto-encoder is composed of a content code c and a style code s. We train the model with adversarial objectives (dotted lines) that ensure the translated images to be indistinguishable from real images in the target domain, as well as bidirectional reconstruction objectives (dashed lines) that reconstruct both images and latent codes.

– **Image reconstruction.** Given an image sampled from the data distribution, we should be able to reconstruct it after encoding and decoding.

$$\mathcal{L}_{\text{recon}}^{x_1} = \mathbb{E}_{x_1 \sim p(x_1)}[||G_1(E_1^c(x_1), E_1^s(x_1)) - x_1||_1] \tag{1}$$

– **Latent reconstruction.** Given a latent code (style and content) sampled from the latent distribution at translation time, we should be able to reconstruct it after decoding and encoding.

$$\mathcal{L}_{\text{recon}}^{c_1} = \mathbb{E}_{c_1 \sim p(c_1), s_2 \sim q(s_2)}[||E_2^c(G_2(c_1, s_2)) - c_1||_1] \tag{2}$$
$$\mathcal{L}_{\text{recon}}^{s_2} = \mathbb{E}_{c_1 \sim p(c_1), s_2 \sim q(s_2)}[||E_2^s(G_2(c_1, s_2)) - s_2||_1] \tag{3}$$

where $q(s_2)$ is the prior $\mathcal{N}(0, \mathbf{I})$, $p(c_1)$ is given by $c_1 = E_1^c(x_1)$ and $x_1 \sim p(x_1)$.

We note the other loss terms $\mathcal{L}_{\text{recon}}^{x_2}$, $\mathcal{L}_{\text{recon}}^{c_2}$, and $\mathcal{L}_{\text{recon}}^{s_1}$ are defined in a similar manner. We use $\mathcal{L}_1$ reconstruction loss as it encourages sharp output images.

The style reconstruction loss $\mathcal{L}_{\text{recon}}^{s_i}$ is reminiscent of the latent reconstruction loss used in the prior works [11, 31, 44, 57]. It has the effect on encouraging diverse outputs given different style codes. The content reconstruction loss $\mathcal{L}_{\text{recon}}^{c_i}$ encourages the translated image to preserve semantic content of the input image.

Adversarial loss. We employ GANs to match the distribution of translated images to the target data distribution. In other words, images generated by our model should be indistinguishable from real images in the target domain.

$$\mathcal{L}_{\text{GAN}}^{x_2} = \mathbb{E}_{c_1 \sim p(c_1), s_2 \sim q(s_2)}[\log(1 - D_2(G_2(c_1, s_2)))] + \mathbb{E}_{x_2 \sim p(x_2)}[\log D_2(x_2)] \tag{4}$$

where D_2 is a discriminator that tries to distinguish between translated images and real images in $\mathcal{X}_2$. The discriminator D_1 and loss $\mathcal{L}_{\text{GAN}}^{x_1}$ are defined similarly.

Total loss. We jointly train the encoders, decoders, and discriminators to optimize the final objective, which is a weighted sum of the adversarial loss and the bidirectional reconstruction loss terms.

$$\min_{E_1,E_2,G_1,G_2} \max_{D_1,D_2} \mathcal{L}(E_1,E_2,G_1,G_2,D_1,D_2) = \mathcal{L}_{\text{GAN}}^{x_1} + \mathcal{L}_{\text{GAN}}^{x_2} + \lambda_x(\mathcal{L}_{\text{recon}}^{x_1} + \mathcal{L}_{\text{recon}}^{x_2}) + \lambda_c(\mathcal{L}_{\text{recon}}^{c_1} + \mathcal{L}_{\text{recon}}^{c_2}) + \lambda_s(\mathcal{L}_{\text{recon}}^{s_1} + \mathcal{L}_{\text{recon}}^{s_2}) \quad (5)$$

where λ_x, λ_c, λ_s are weights that control the importance of reconstruction terms.

4 Theoretical Analysis

We now establish some theoretical properties of our framework. Specifically, we show that minimizing the proposed loss function leads to 1) matching of latent distributions during encoding and generation, 2) matching of two joint image distributions induced by our framework, and 3) enforcing a weak form of cycle consistency constraint. All the proofs are given in the supplementary material.

First, we note that the total loss in Eq. (5) is minimized when the translated distribution matches the data distribution and the encoder-decoder are inverses.

Proposition 1. *Suppose there exists* E_1^*, E_2^*, G_1^*, G_2^* *such that: 1)* $E_1^* = (G_1^*)^{-1}$ *and* $E_2^* = (G_2^*)^{-1}$, *and 2)* $p(x_{1\to 2}) = p(x_2)$ *and* $p(x_{2\to 1}) = p(x_1)$. *Then* E_1^*, E_2^*, G_1^*, G_2^* *minimizes* $\mathcal{L}(E_1,E_2,G_1,G_2) = \max_{D_1,D_2} \mathcal{L}(E_1,E_2,G_1,G_2,D_1,D_2)$ *(Eq. (5)).*

Latent Distribution Matching For image generation, existing works on combining auto-encoders and GANs need to match the encoded latent distribution with the latent distribution the decoder receives at generation time, using either KLD loss [15, 40] or adversarial loss [17, 42] in the latent space. The auto-encoder training would not help GAN training if the decoder received a very different latent distribution during generation. Although our loss function does not contain terms that explicitly encourage the match of latent distributions, it has the effect of matching them implicitly.

Proposition 2. *When optimality is reached, we have:*

$$p(c_1) = p(c_2),\ p(s_1) = q(s_1),\ p(s_2) = q(s_2)$$

The above proposition shows that at optimality, the encoded style distributions match their Gaussian priors. Also, the encoded content distribution matches the distribution at generation time, which is just the encoded distribution from the other domain. This suggests that the content space becomes domain-invariant.

Joint Distribution Matching Our model learns two conditional distributions $p(x_{1\to 2}|x_1)$ and $p(x_{2\to 1}|x_2)$, which, together with the data distributions, define two joint distributions $p(x_1, x_{1\to 2})$ and $p(x_{2\to 1}, x_2)$. Since both of them are designed to approximate the same underlying joint distribution $p(x_1, x_2)$, it is desirable that they are consistent with each other, *i.e.*, $p(x_1, x_{1\to 2}) = p(x_{2\to 1}, x_2)$.

Joint distribution matching provides an important constraint for unsupervised image-to-image translation and is behind the success of many recent methods. Here, we show our model matches the joint distributions at optimality.

Proposition 3. *When optimality is reached, we have* $p(x_1, x_{1\to2}) = p(x_{2\to1}, x_2)$.

Style-augmented Cycle Consistency Joint distribution matching can be realized via cycle consistency constraint [8], assuming deterministic translation models and matched marginals [43, 67, 68]. However, we note that this constraint is too strong for multimodal image translation. In fact, we prove in the supplementary material that the translation model will degenerate to a deterministic function if cycle consistency is enforced. In the following proposition, we show that our framework admits a weaker form of cycle consistency, termed as style-augmented cycle consistency, between the image–style joint spaces, which is more suited for multimodal image translation.

Proposition 4. *Denote* $h_1 = (x_1, s_2) \in \mathcal{H}_1$ *and* $h_2 = (x_2, s_1) \in \mathcal{H}_2$. h_1, h_2 *are points in the joint spaces of image and style. Our model defines a deterministic mapping* $F_{1\to2}$ *from* $\mathcal{H}_1$ *to* $\mathcal{H}_2$ *(and vice versa) by* $F_{1\to2}(h_1) = F_{1\to2}(x_1, s_2) \triangleq (G_2(E_1^c(x_1), s_2), E_1^s(x_1))$. *When optimality is achieved, we have* $F_{1\to2} = F_{2\to1}^{-1}$.

Intuitively, style-augmented cycle consistency implies that if we translate an image to the target domain and translate it back *using the original style*, we should obtain the original image. Note that we do not use any explicit loss terms to enforce style-augmented cycle consistency, but it is implied by the proposed bidirectional reconstruction loss.

5 Experiments

5.1 Implementation Details

Fig. 3 shows the architecture of our auto-encoder. It consists of a content encoder, a style encoder, and a joint decoder. More detailed information and hyperparameters are given in the supplementary material. We will provide an open-source implementation in PyTorch [69].

Content encoder. Our content encoder consists of several strided convolutional layers to downsample the input and several residual blocks [70] to further process it. All the convolutional layers are followed by Instance Normalization (IN) [71].

Style encoder. The style encoder includes several strided convolutional layers, followed by a global average pooling layer and a fully connected (FC) layer. We do not use IN layers in the style encoder, since IN removes the original feature mean and variance that represent important style information [54].

Decoder. Our decoder reconstructs the input image from its content and style code. It processes the content code by a set of residual blocks and finally produces the reconstructed image by several upsampling and convolutional layers.

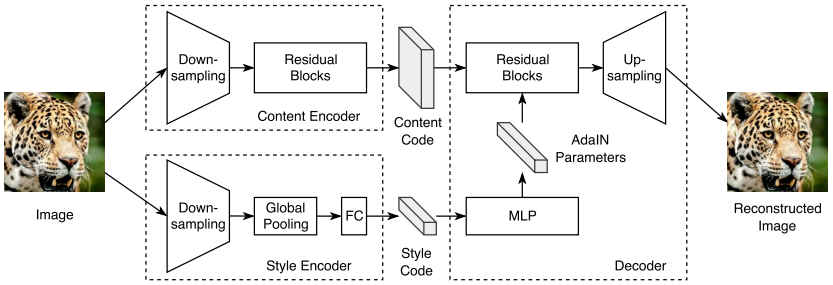


Fig. 3. Our auto-encoder architecture. The content encoder consists of several strided convolutional layers followed by residual blocks. The style encoder contains several strided convolutional layers followed by a global average pooling layer and a fully connected layer. The decoder uses a MLP to produce a set of AdaIN [54] parameters from the style code. The content code is then processed by residual blocks with AdaIN layers, and finally decoded to the image space by upsampling and convolutional layers.

Inspired by recent works that use affine transformation parameters in normalization layers to represent styles [54, 72–74], we equip the residual blocks with Adaptive Instance Normalization (AdaIN) [54] layers whose parameters are dynamically generated by a multilayer perceptron (MLP) from the style code.

$$\text{AdaIN}(z, \gamma, \beta) = \gamma \left(\frac{z - \mu(z)}{\sigma(z)} \right) + \beta \tag{6}$$

where z is the activation of the previous convolutional layer, μ and σ are channel-wise mean and standard deviation, γ and β are parameters generated by the MLP. Note that the affine parameters are produced by a learned network, instead of computed from statistics of a pretrained network as in Huang *et al.* [54].

Discriminator. We use the LSGAN objective proposed by Mao *et al.* [38]. We employ multi-scale discriminators proposed by Wang *et al.* [20] to guide the generators to produce both realistic details and correct global structure.

Domain-invariant perceptual loss. The perceptual loss, often computed as a distance in the VGG [75] feature space between the output and the reference image, has been shown to benefit image-to-image translation when paired supervision is available [13, 20]. In the unsupervised setting, however, we do not have a reference image in the target domain. We propose a modified version of perceptual loss that is more domain-invariant, so that we can use the input image as the reference. Specifically, before computing the distance, we perform Instance Normalization [71] (without affine transformations) on the VGG features in order to remove the original feature mean and variance, which contains much domain-specific information [54, 76]. We find it accelerates training on high-resolution ($\geq 512 \times 512$) datasets and thus employ it on those datasets.

5.2 Evaluation Metrics

Human Preference. To compare the realism and faithfulness of translation outputs generated by different methods, we perform human perceptual study on Amazon Mechanical Turk (AMT). Similar to Wang *et al.* [20], the workers are given an input image and two translation outputs from different methods. They are then given unlimited time to select which translation output looks more accurate. For each comparison, we randomly generate 500 questions and each question is answered by 5 different workers.

LPIPS Distance. To measure translation diversity, we compute the average LPIPS distance [77] between pairs of randomly-sampled translation outputs from the same input as in Zhu *et al.* [11]. LPIPS is given by a weighted $\mathcal{L}_2$ distance between deep features of images. It has been demonstrated to correlate well with human perceptual similarity [77]. Following Zhu *et al.* [11], we use 100 input images and sample 19 output pairs per input, which amounts to 1900 pairs in total. We use the ImageNet-pretrained AlexNet [78] as the deep feature extractor.

(Conditional) Inception Score. The Inception Score (IS) [34] is a popular metric for image generation tasks. We propose a modified version called Conditional Inception Score (CIS), which is more suited for evaluating multimodal image translation. When we know the number of modes in $\mathcal{X}_2$ as well as the ground truth mode each sample belongs to, we can train a classifier $p(y_2|x_2)$ to classify an image x_2 into its mode y_2. Conditioned on a single input image x_1, the translation samples $x_{1\to2}$ should be mode-covering (thus $p(y_2|x_1) = \int p(y|x_{1\to2})p(x_{1\to2}|x_1)\,dx_{1\to2}$ should have high entropy) and each individual sample should belong to a specific mode (thus $p(y_2|x_{1\to2})$ should have low entropy). Combing these two requirements we get:

$$\text{CIS} = \mathbb{E}_{x_1\sim p(x_1)}[\mathbb{E}_{x_{1\to2}\sim p(x_{2\to1}|x_1)}[\text{KL}(p(y_2|x_{1\to2})||p(y_2|x_1))]] \tag{7}$$

To compute the (unconditional) IS, $p(y_2|x_1)$ is replaced with the unconditional class probability $p(y_2) = \iint p(y|x_{1\to2})p(x_{1\to2}|x_1)p(x_1)\,dx_1\,dx_{1\to2}$.

$$\text{IS} = \mathbb{E}_{x_1\sim p(x_1)}[\mathbb{E}_{x_{1\to2}\sim p(x_{2\to1}|x_1)}[\text{KL}(p(y_2|x_{1\to2})||p(y_2))]] \tag{8}$$

To obtain a high CIS/IS score, a model needs to generate samples that are both high-quality and diverse. While IS measures diversity of all output images, CIS measures diversity of outputs conditioned on a single input image. A model that deterministically generates a single output given an input image will receive a zero CIS score, though it might still get a high score under IS. We use the Inception-v3 [79] fine-tuned on our specific datasets as the classifier and estimate Eq. (7) and Eq. (8) using 100 input images and 100 samples per input.

5.3 Baselines

UNIT [15]. The UNIT model consists of two VAE-GANs with a fully shared latent space. The stochasticity of the translation comes from the Gaussian encoders as well as the dropout layers in the VAEs.

CycleGAN [8]. CycleGAN consists of two residual translation networks trained with adversarial loss and cycle reconstruction loss. We use Dropout during both training and testing to encourage diversity, as suggested in Isola *et al.* [6].
CycleGAN* [8] with noise. To test whether we can generate multimodal outputs within the CycleGAN framework, we additionally inject noise vectors to both translation networks. We use the U-net architecture [11] with noise added to input, since we find the noise vectors are ignored by the residual architecture in CycleGAN [8]. Dropout is also utilized during both training and testing.
BicycleGAN [11]. BicycleGAN is the only existing image-to-image translation model we are aware of that can generate continuous and multimodal output distributions. However, it requires paired training data. We compare our model with BicycleGAN when the dataset contains pair information.

5.4 Datasets

Edges ↔ shoes/handbags. We use the datasets provided by Isola *et al.* [6], Yu *et al.* [80], and Zhu *et al.* [81], which contain images of shoes and handbags with edge maps generated by HED [82]. We train one model for edges ↔ shoes and another for edges ↔ handbags without using paired information.
Animal image translation. We collect images from 3 categories/domains, including house cats, big cats, and dogs. Each domain contains 4 modes which are fine-grained categories belonging to the same parent category. Note that the modes of the images are not known during learning the translation model. We learn a separate model for each pair of domains.
Street scene images. We experiment with two street scene translation tasks:

- **Synthetic ↔ real.** We perform translation between synthetic images in the SYNTHIA dataset [83] and real-world images in the Cityscape dataset [84]. For the SYNTHIA dataset, we use the SYNTHIA-Seqs subset which contains images in different seasons, weather, and illumination conditions.
- **Summer ↔ winter.** We use the dataset from Liu et al. [15], which contains summer and winter street images extracted from real-world driving videos.

Yosemite summer ↔ winter (HD). We collect a new high-resolution dataset containing 3253 summer photos and 2385 winter photos of Yosemite. The images are downsampled such that the shortest side of each image is 1024 pixels.

5.5 Results

First, we qualitatively compare MUNIT with the four baselines above, and three variants of MUNIT that ablate $\mathcal{L}^x_{\text{recon}}$, $\mathcal{L}^c_{\text{recon}}$, $\mathcal{L}^s_{\text{recon}}$ respectively. Fig. 4 shows example results on edges → shoes. Both UNIT and CycleGAN (with or without noise) fail to generate diverse outputs, despite the injected randomness. Without $\mathcal{L}^x_{\text{recon}}$ or $\mathcal{L}^c_{\text{recon}}$, the image quality of MUNIT is unsatisfactory. Without $\mathcal{L}^s_{\text{recon}}$, the model suffers from partial mode collapse, with many outputs being almost identical (*e.g.*, the first two rows). Our full model produces images that are both diverse and realistic, similar to BicycleGAN but does not need supervision.

Fig. 4. Qualitative comparison on edges → shoes. The first column shows the input and ground truth output. Each following column shows 3 random outputs from a method.

Table 1. Quantitative evaluation on edges → shoes/handbags. The diversity score is the average LPIPS distance [77]. The quality score is the human preference score, the percentage a method is preferred over MUNIT. For both metrics, the higher the better.

	edges → shoes		edges → handbags	
	Quality	Diversity	Quality	Diversity
UNIT [15]	37.4%	0.011	37.3%	0.023
CycleGAN [8]	36.0%	0.010	40.8%	0.012
CycleGAN* [8] with noise	29.5%	0.016	45.1%	0.011
MUNIT w/o $\mathcal{L}^x_{\text{recon}}$	6.0%	0.213	29.0%	0.191
MUNIT w/o $\mathcal{L}^c_{\text{recon}}$	20.7%	0.172	9.3%	0.185
MUNIT w/o $\mathcal{L}^s_{\text{recon}}$	28.6%	0.070	24.6%	0.139
MUNIT	50.0%	0.109	50.0%	0.175
BicycleGAN [11]†	56.7%	0.104	51.2%	0.140
Real data	N/A	0.293	N/A	0.371

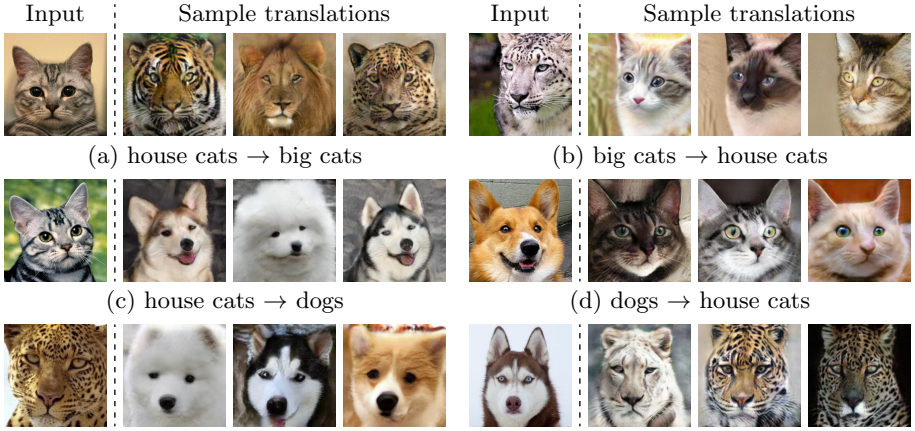
† Trained with paired supervision.

The qualitative observations above are confirmed by quantitative evaluations. We use human preference to measure quality and LPIPS distance to evaluate diversity, as described in Sec. 5.2. We conduct this experiment on the task of edges → shoes/handbags. As shown in Table 1, UNIT and CycleGAN produce very little diversity according to LPIPS distance. Removing $\mathcal{L}^x_{\text{recon}}$ or $\mathcal{L}^c_{\text{recon}}$ from MUNIT leads to significantly worse quality. Without $\mathcal{L}^s_{\text{recon}}$, both quality and diversity deteriorate. The full model obtains quality and diversity comparable to the fully supervised BicycleGAN, and significantly better than all unsupervised baselines. In Fig. 5, we show more example results on edges ↔ shoes/handbags.

We proceed to perform experiments on the animal image translation dataset. As shown in Fig. 6, our model successfully translate one kind of animal to an-

(a) edges ↔ shoes (b) edges ↔ handbags

Fig. 5. Example results of (a) edges ↔ shoes and (b) edges ↔ handbags.



(a) house cats → big cats (b) big cats → house cats

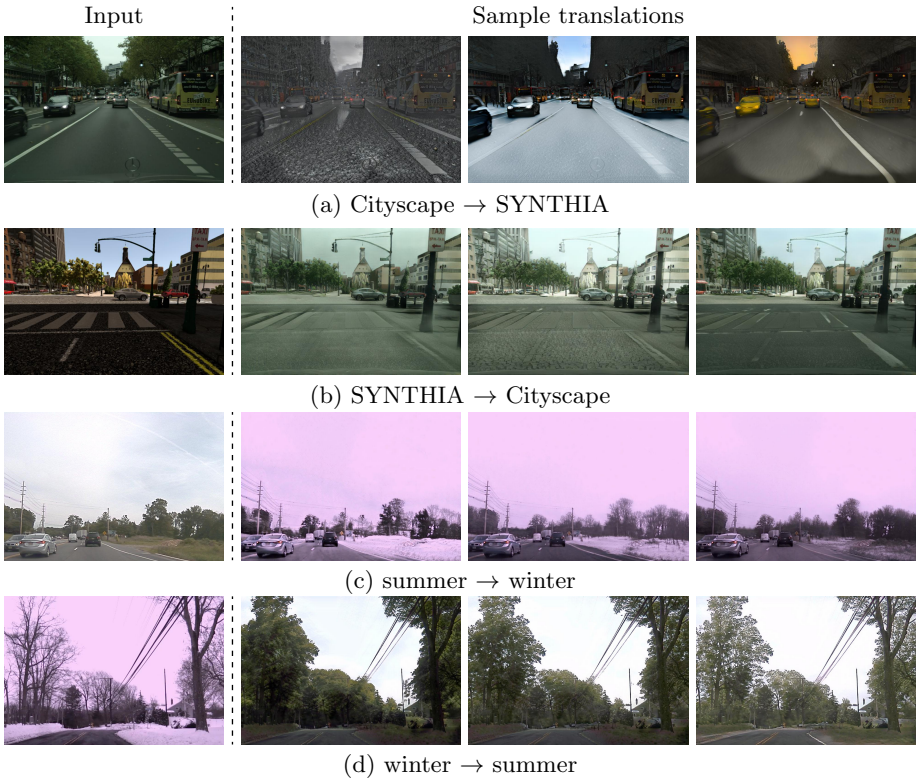
(c) house cats → dogs (d) dogs → house cats

(e) big cats → dogs (f) dogs → big cats

Fig. 6. Example results of animal image translation.

other. Given an input image, the translation outputs cover multiple modes, *i.e.*, multiple fine-grained animal categories in the target domain. The shape of an animal has undergone significant transformations, but the pose is overall preserved. As shown in Table 2, our model obtains the highest scores according to both CIS and IS. In particular, the baselines all obtain a very low CIS, indicating their failure to generate multimodal outputs from a given input. As the IS has been shown to correlate well to image quality [34], the higher IS of our method suggests that it also generates images of high quality than baseline approaches.

Fig. 7 shows results on street scene datasets. Our model is able to generate SYNTHIA images with diverse renderings (*e.g.*, rainy, snowy, sunset) from a given Cityscape image, and generate Cityscape images with different lighting, shadow, and road textures from a given SYNTHIA image. Similarly, it generates winter images with different amount of snow from a given summer image, and summer images with different amount of leafs from a given winter image.



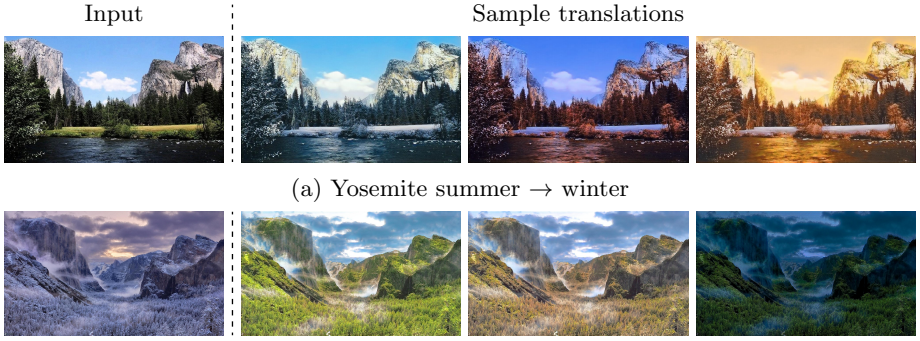
(a) Cityscape → SYNTHIA

(b) SYNTHIA → Cityscape

(c) summer → winter

(d) winter → summer

Fig. 7. Example results on street scene translations.



(a) Yosemite summer → winter

(b) Yosemite winter → summer

Fig. 8. Example results on Yosemite summer ↔ winter (HD resolution).

Fig. 8 shows example results of summer ↔ winter transfer on the high-resolution Yosemite dataset. Our algorithm generates output images with different lighting.

Example-guided Image Translation. Instead of sampling the style code from the prior, it is also possible to extract the style code from a reference image. Specifically, given a content image $x_1 \in \mathcal{X}_1$ and a style image $x_2 \in \mathcal{X}_2$, our

(a) edges → shoes (b) big cats → house cats

Fig. 9. image translation. Each row has the same content while each column has the same style. The color of the generated shoes and the appearance of the generated cats can be specified by providing example style images.

Table 2. Quantitative evaluation on animal image translation. This dataset contains 3 domains. We perform bidirectional translation for each domain pair, resulting in 6 translation tasks. We use CIS and IS to measure the performance on each task. To obtain a high CIS/IS score, a model needs to generate samples that are both high-quality and diverse. While IS measures diversity of all output images, CIS measures diversity of outputs conditioned on a single input image.

	CycleGAN		CycleGAN* with noise		UNIT		MUNIT	
	CIS	IS	CIS	IS	CIS	IS	CIS	IS
house cats → big cats	0.078	0.795	0.034	0.701	0.096	0.666	0.911	0.923
big cats → house cats	0.109	0.887	0.124	0.848	0.164	0.817	0.956	0.954
house cats → dogs	0.044	0.895	0.070	0.901	0.045	0.827	1.231	1.255
dogs → house cats	0.121	0.921	0.137	0.978	0.193	0.982	1.035	1.034
big cats → dogs	0.058	0.762	0.019	0.589	0.094	0.910	1.205	1.233
dogs → big cats	0.047	0.620	0.022	0.558	0.096	0.754	0.897	0.901
Average	0.076	0.813	0.068	0.762	0.115	0.826	1.039	1.050

model produces an image $x_{1\to 2}$ that recombines the content of the former and the style latter by $x_{1\to 2} = G_2(E_1^c(x_1), E_2^s(x_2))$. Examples are shown in Fig. 9.

6 Conclusions

We presented a framework for multimodal unsupervised image-to-image translation. Our model achieves quality and diversity superior to existing unsupervised methods and comparable to state-of-the-art supervised approach.

References

1. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: ECCV. (2014)
2. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV. (2016)
3. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR. (2016)
4. Laffont, P.Y., Ren, Z., Tao, X., Qian, C., Hays, J.: Transient attributes for high-level understanding and editing of outdoor scenes. TOG (2014)
5. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR. (2016)
6. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR. (2017)
7. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: ICCV. (2017)
8. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV. (2017)
9. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: ICML. (2017)
10. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised cross-domain image generation. In: ICLR. (2017)
11. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: NIPS. (2017)
12. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: NIPS. (2016)
13. Chen, Q., Koltun, V.: Photographic image synthesis with cascaded refinement networks. In: ICCV. (2017)
14. Liang, X., Zhang, H., Xing, E.P.: Generative semantic manipulation with contrasting gan. arXiv preprint arXiv:1708.00315 (2017)
15. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: NIPS. (2017)
16. Benaim, S., Wolf, L.: One-sided unsupervised domain mapping. In: NIPS. (2017)
17. Royer, A., Bousmalis, K., Gouws, S., Bertsch, F., Moressi, I., Cole, F., Murphy, K.: Xgan: Unsupervised image-to-image translation for many-to-many mappings. arXiv preprint arXiv:1711.05139 (2017)
18. Gan, Z., Chen, L., Wang, W., Pu, Y., Zhang, Y., Liu, H., Li, C., Carin, L.: Triangle generative adversarial networks. In: NIPS. (2017) 5253–5262
19. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: CVPR. (2018)
20. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: CVPR. (2018)
21. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: CVPR. (2017)
22. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: CVPR. (2017)
23. Wolf, L., Taigman, Y., Polyak, A.: Unsupervised creation of parameterized avatars. In: ICCV. (2017)

24. TAU, T.G., Wolf, L., TAU, S.B.: The role of minimal complexity functions in unsupervised learning of semantic mappings. In: ICLR. (2018)
25. Hoshen, Y., Wolf, L.: Identifying analogies across domains. In: ICLR. (2018)
26. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. In: ICLR. (2016)
27. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS. (2014)
28. Denton, E.L., Chintala, S., Fergus, R.: Deep generative image models using a laplacian pyramid of adversarial networks. In: NIPS. (2015)
29. Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: ECCV. (2016)
30. Yang, J., Kannan, A., Batra, D., Parikh, D.: Lr-gan: Layered recursive generative adversarial networks for image generation. In: ICLR. (2017)
31. Huang, X., Li, Y., Poursaeed, O., Hopcroft, J., Belongie, S.: Stacked generative adversarial networks. In: CVPR. (2017)
32. Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., Metaxas, D.: Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: ICCV. (2017)
33. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. In: ICLR. (2018)
34. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: NIPS. (2016)
35. Zhao, J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. In: ICLR. (2017)
36. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML. (2017)
37. Berthelot, D., Schumm, T., Metz, L.: Began: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717 (2017)
38. Mao, X., Li, Q., Xie, H., Lau, Y.R., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. In: ICCV. (2017)
39. Tolstikhin, I., Bousquet, O., Gelly, S., Schoelkopf, B.: Wasserstein auto-encoders. In: ICLR. (2018)
40. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. In: ICML. (2016)
41. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: NIPS. (2016)
42. Rosca, M., Lakshminarayanan, B., Warde-Farley, D., Mohamed, S.: Variational approaches for auto-encoding generative adversarial networks. arXiv preprint arXiv:1706.04987 (2017)
43. Li, C., Liu, H., Chen, C., Pu, Y., Chen, L., Henao, R., Carin, L.: Alice: Towards understanding adversarial learning for joint distribution matching. In: NIPS. (2017)
44. Srivastava, A., Valkoz, L., Russell, C., Gutmann, M.U., Sutton, C.: Veegan: Reducing mode collapse in gans using implicit variational learning. In: NIPS. (2017)
45. Ghosh, A., Kulharia, V., Namboodiri, V., Torr, P.H., Dokania, P.K.: Multi-agent diverse generative adversarial networks. arXiv preprint arXiv:1704.02906 (2017)
46. Bansal, A., Sheikh, Y., Ramanan, D.: Pixelnn: Example-based image synthesis. In: ICLR. (2018)
47. Almahairi, A., Rajeswar, S., Sordoni, A., Bachman, P., Courville, A.: Augmented cyclegan: Learning many-to-many mappings from unpaired data. arXiv preprint arXiv:1802.10151 (2018)

48. Lee, H.Y., Tseng, H.Y., Huang, J.B., Singh, M.K., Yang, M.H.: Diverse image-to-image translation via disentangled representation. In: ECCV. (2018)
49. Anoosheh, A., Agustsson, E., Timofte, R., Van Gool, L.: Combogan: Unrestrained scalability for image domain translation. arXiv preprint arXiv:1712.06909 (2017)
50. Hui, L., Li, X., Chen, J., He, H., Yang, J., et al.: Unsupervised multi-domain image translation with domain-specific encoders/decoders. arXiv preprint arXiv:1712.02050 (2017)
51. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. In: SIGGRAPH. (2001)
52. Li, C., Wand, M.: Combining markov random fields and convolutional neural networks for image synthesis. In: CVPR. (2016)
53. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV. (2016)
54. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: ICCV. (2017)
55. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Universal style transfer via feature transforms. In: NIPS. (2017) 385–395
56. Li, Y., Liu, M.Y., Li, X., Yang, M.H., Kautz, J.: A closed-form solution to photo-realistic image stylization. In: ECCV. (2018)
57. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: NIPS. (2016)
58. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: ICLR. (2017)
59. Tenenbaum, J.B., Freeman, W.T.: Separating style and content. In: NIPS. (1997)
60. Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D.: Domain separation networks. In: NIPS. (2016)
61. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. In: ICLR. (2017)
62. Mathieu, M.F., Zhao, J.J., Zhao, J., Ramesh, A., Sprechmann, P., LeCun, Y.: Disentangling factors of variation in deep representation using adversarial training. In: NIPS. (2016)
63. Denton, E.L., et al.: Unsupervised learning of disentangled representations from video. In: NIPS. (2017)
64. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. In: CVPR. (2018)
65. Donahue, C., Balsubramani, A., McAuley, J., Lipton, Z.C.: Semantically decomposing the latent spaces of generative adversarial networks. In: ICLR. (2018)
66. Shen, T., Lei, T., Barzilay, R., Jaakkola, T.: Style transfer from non-parallel text by cross-alignment. In: Advances in Neural Information Processing Systems. (2017) 6833–6844
67. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. In: ICLR. (2017)
68. Dumoulin, V., Belghazi, I., Poole, B., Lamb, A., Arjovsky, M., Mastropietro, O., Courville, A.: Adversarially learned inference. In: ICLR. (2017)
69. Automatic differentiation in PyTorch. In: NIPS Autodiff Workshop. (2017)
70. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016)

71. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In: CVPR. (2017)
72. Dumoulin, V., Shlens, J., Kudlur, M.: A learned representation for artistic style. In: ICLR. (2017)
73. Wang, H., Liang, X., Zhang, H., Yeung, D.Y., Xing, E.P.: Zm-net: Real-time zero-shot image manipulation network. arXiv preprint arXiv:1703.07255 (2017)
74. Ghiasi, G., Lee, H., Kudlur, M., Dumoulin, V., Shlens, J.: Exploring the structure of a real-time, arbitrary neural artistic stylization network. In: BMVC. (2017)
75. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR. (2015)
76. Li, Y., Wang, N., Shi, J., Liu, J., Hou, X.: Revisiting batch normalization for practical domain adaptation. arXiv preprint arXiv:1603.04779 (2016)
77. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR. (2018)
78. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012)
79. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: CVPR. (2016)
80. Yu, A., Grauman, K.: Fine-grained visual comparisons with local learning. In: CVPR. (2014)
81. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: ECCV. (2016)
82. Xie, S., Tu, Z.: Holistically-nested edge detection. In: ICCV. (2015)
83. Ros, G., Sellart, L., Materzynska, J., Vazquez, D., Lopez, A.M.: The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In: CVPR. (2016)
84. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR. (2016)