HAL
open science

Geometry-Consistent Generative Adversarial Networks for One-Sided Unsupervised Domain Mapping

Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, Kun Zhang, Dacheng Tao

To cite this version:

Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, Kun Zhang, et al.. Geometry-Consistent Generative Adversarial Networks for One-Sided Unsupervised Domain Mapping. CVPR 2019 - IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun 2019, Long Beach, CA, United States. pp.2422-2431, 10.1109/CVPR.2019.00253 . hal-03132263

HAL Id: hal-03132263

https://hal.science/hal-03132263

Submitted on 4 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Geometry-Consistent Generative Adversarial Networks for One-Sided Unsupervised Domain Mapping

Huan Fu*[1] Mingming Gong* [2,3] Chaohui Wang[4] Kayhan Batmanghelich[2] Kun Zhang[3] Dacheng Tao[1]
[1]UBTECH Sydney AI Centre, School of Computer Science, FEIT, University of Sydney, Darlington, NSW 2008, Australia
[2]Department of Biomedical Informatics, University of Pittsburgh
[3]Department of Philosophy, Carnegie Mellon University
[4]Université Paris-Est, LIGM (UMR 8049), CNRS, ENPC, ESIEE Paris, UPEM, Marne-la-Vallée, France
{hufu6371@uni., dacheng.tao@}sydney.edu.au {mig73, kayhan}@pitt.edu chaohui.wang@u-pem.fr kunz1@cmu.edu

Abstract

Unsupervised domain mapping aims to learn a function G_{XY} to translate domain $\mathcal{X}$ to $\mathcal{Y}$ in the absence of paired examples. Finding the optimal G_{XY} without paired data is an ill-posed problem, so appropriate constraints are required to obtain reasonable solutions. While some prominent constraints such as cycle consistency and distance preservation successfully constrain the solution space, they overlook the special properties of images that simple geometric transformations do not change the image's semantic structure. Based on this special property, we develop a geometry-consistent generative adversarial network (GcGAN), which enables one-sided unsupervised domain mapping. GcGAN takes the original image and its counterpart image transformed by a predefined geometric transformation as inputs and generates two images in the new domain coupled with the corresponding geometry-consistency constraint. The geometry-consistency constraint reduces the space of possible solutions while keep the correct solutions in the search space. Quantitative and qualitative comparisons with the baseline (GAN alone) and the state-of-the-art methods including CycleGAN [66] and DistanceGAN [5] demonstrate the effectiveness of our method. Codes will be made available at https://github.com/hufu6371/GcGAN.

1. Introduction

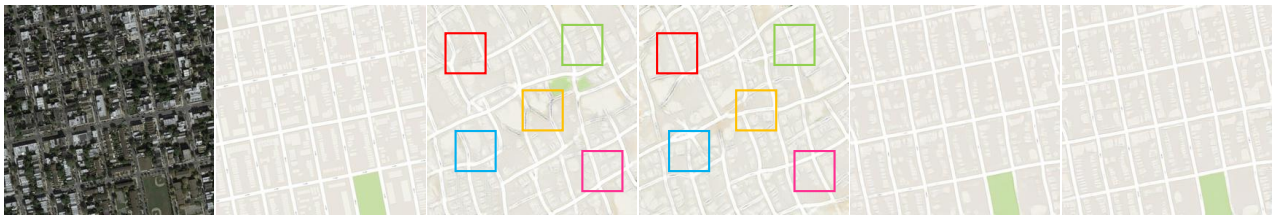
Domain mapping or image-to-image translation, which targets at translating an image from one domain to another, has been intensively investigated over the past few years. Let $X \in \mathcal{X}$ denote a random variable representing source domain images and $Y \in \mathcal{Y}$ represent target domain images. According to whether we have access to a paired sample $\{(x_i, y_i)\}_{i=1}^N$, domain mapping can be studied in a supervised or unsupervised manner. While several works have successfully produced high-quality translations by focusing on supervised domain mapping with constraints provided by cross-domain image pairs [46, 26, 59, 58], the progress of unsupervised domain mapping is relatively slow.

In unsupervised domain mapping, the goal is to model the joint distribution P_{XY} given samples drawn from the marginal distributions P_X and P_Y in individual domains. Since the two marginal distributions can be inferred from an infinite set of possible joint distributions, it is difficult to guarantee that an individual input $x \in X$ and the output $G_{XY}(x)$ are paired up in a meaningful way without additional assumptions or constraints.

To address this problem, recent approaches have exploited the cycle-consistency assumption, *i.e.*, a mapping G_{XY} and its inverse mapping G_{YX} should be bijections [66, 28, 61]. Specifically, when feeding an example $x \in X$ into the networks $G_{XY} \circ G_{YX} : X \to Y \to X$, the output should be a reconstruction of x and vise versa for y, *i.e.*, $G_{YX}(G_{XY}(x)) \approx x$ and $G_{XY}(G_{YX}(y)) \approx y$. Further, DistanceGAN [5] showed that maintaining the distances between images within domains allows one-sided unsupervised domain mapping.

Existing constraints overlook the special properties of images that simple geometric transformations (global geometric transformations without shape deformation) do not change the image's semantic structure. Here, semantic structure refers to the information that distinguishes different object/staff classes, which can be easily perceived by humans regardless of trivial geometric transformations such as rotation. Based on this property, we develop a geometry-consistency constraint, which helps in reducing the search space of possible solutions while still keeping the correct set of solutions under consideration, and results in a geometry-

*equal contribution



Input | Ground Truth | GAN alone | GAN alone (*rot*) | GcGAN | GcGAN (*rot*)

Figure 1: **Geometry consistency.** The original input image is denoted by x, and the predefined function $f(\cdot)$ is a 90° clockwise rotation (*rot*). GAN alone: $G^1_{XY}(x)$. GAN alone (*rot*): $f^{-1}(G^1_{\tilde{X}\tilde{Y}}(f(x)))$. GcGAN: $G^2_{XY}(x)$. GcGAN (*rot*): $f^{-1}(G^2_{\tilde{X}\tilde{Y}}(f(x))$. It can be seen that GAN alone produces geometrically-inconsistent output images, indicating that the learned G_{XY} and $G_{\tilde{X}\tilde{Y}}$ are far away from the correct mapping functions. By enforcing geometry consistency, our method results in more sensible domain mapping.

consistent generative adversarial network (GcGAN).

Our geometry-consistency constraint is motivated by the fact that a given geometric transformation $f(\cdot)$ between the input images should be preserved by related translators G_{XY} and $G_{\tilde{X}\tilde{Y}}$, if $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ are the domains obtained by applying $f(\cdot)$ on the examples of X and Y, respectively. Mathematically, given a random example x from the source domain $\mathcal{X}$ and a predefined geometric transformation function $f(\cdot)$, geometry consistency can be expressed as $f(G_{XY}(x)) \approx G_{\tilde{X}\tilde{Y}}(f(x))$ and $f^{-1}(G_{\tilde{X}\tilde{Y}}(f(x))) \approx G_{XY}(x)$, where $f^{-1}(\cdot)$ is the inverse function of $f(\cdot)$. Because it is unlikely that G_{XY} and $G_{\tilde{X}\tilde{Y}}$ always fail in the same location, G_{XY} and $G_{\tilde{X}\tilde{Y}}$ co-regularize each other by the geometry-consistency constraint and thus correct each others' failures in local regions of their respective translations (see Figure 1 for an illustration). Our geometry-consistency constraint allows one-sided unsupervised domain mapping, *i.e.*, G_{XY} can be trained independently from G_{YX}. In this paper, we employ two simple but representative geometric transformations as examples, *i.e.*, vertical flipping (*vf*) and 90 degrees clockwise rotation (*rot*), to illustrate geometry consistency. Quantitative and qualitative comparisons with the baseline (GAN alone) and the state-of-the-art methods including CycleGAN [66] and DistanceGAN [5] demonstrate the effectiveness of our method.

2. Related Work

Generative Adversarial Networks. Generative adversarial networks (GANs) [21, 45, 14, 47, 51, 3] learn two networks, *i.e.*, a generator and a discriminator, in a staged zero-sum game fashion to generate images from inputs. Many tasks have recently been developed based on deep convolutional GANs, such as image inpainting, style transfer, and domain adaptation [7, 62, 46, 48, 31, 60, 9, 52, 23, 53, 64, 27, 50, 19, 18, 35, 63]. The key component enabling GANs is the adversarial constraint, which enforces the generated images to be indistinguishable from real images.

Domain Mapping. Recent adversarial domain mapping has been studied in a supervised or unsupervised manner with respect to paired or unpaired inputs. There are a variety of literatures [46, 31, 26, 59, 56, 58, 25, 37, 4, 10] on supervised domain mapping. One representative example is conditional GAN [26], which learned the discriminator to distinguish (x, y) and $(x, G_{XY}(x))$ instead of y and $G_{XY}(x)$, where (x, y) is a meaningful pair across domains. Further, Wang *et al*. [59] showed that conditional GANs can be used to generate high-resolution images with a novel feature matching loss, as well as multi-scale generator and discriminator architectures. While there has been significant progress in supervised domain mapping, many real-word applications can not provide aligned images across domains because data preparation is expensive. Thus, different constraints and frameworks have been proposed for image-to-image translation in the absence of training pairs.

In unsupervised domain mapping, only unaligned examples in individual domains are provided, making the task more practical but more difficult. Unpaired domain mapping has a long history, and some successes in adversarial networks have recently been presented [40, 66, 5, 39, 42, 38, 6, 11]. For example, Liu and Tuzel [40] introduced coupled GAN (CoGAN) to learn cross-domain representations by enforcing a weight-sharing constraint. Subsequently, CycleGAN [66], DiscoGAN [28], and DualGAN [61] enforced that translators G_{XY} and G_{YX} should be bijections. Thus, jointly learning G_{XY} and G_{YX} by enforcing cycle consistency can help to produce convincing mappings. Since then, many constraints and assumptions have been proposed to improve cycle consistency [8, 17, 24, 32, 34, 11, 2, 67, 20, 44, 39, 36, 1]. Recently, Benaim and Wolf [5] reported that maintaining the distances between samples within domains allows one-sided unsupervised domain mapping. GcGAN is also a one-sided framework coupled with our geometry-consistency constraint, and produces competitive and even better translations than the two-sided CycleGAN in various applications.

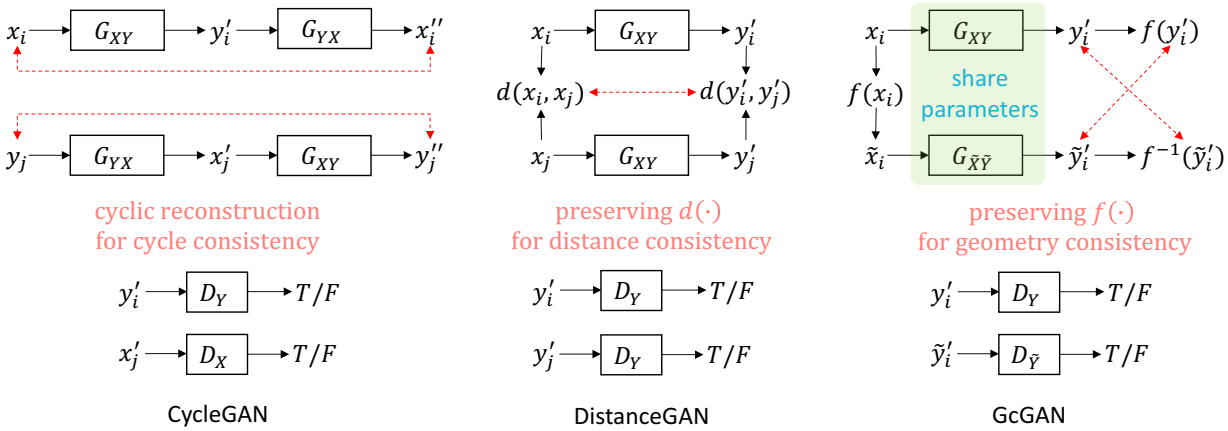


Figure 2: **An illustration of the differences between CycleGAN [66], DistanceGAN [5], and our GcGAN**. x and y are random examples from domain $\mathcal{X}$ and $\mathcal{Y}$, respectively. $d(x_i, x_j)$ is the distance between images x_i and x_j. $f(\cdot)$ is a predefined geometric transformation function for images, which satisfies $f^{-1}(f(x)) = f(f^{-1}(x)) = x$. G_{XY} and $G_{\tilde{X}\tilde{Y}}$ are the generators (or translators) which target the domain translation tasks from $\mathcal{X}$ to $\mathcal{Y}$ and $\tilde{\mathcal{X}}$ to $\tilde{\mathcal{Y}}$, where $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ are two domains obtained by applying $f(\cdot)$ on all the images in $\mathcal{X}$ and $\mathcal{Y}$, respectively. D_Y is an adversarial discriminator in domain $\mathcal{Y}$. The red dotted lines denote the unsupervised constraints with respect to cycle consistency ($x \approx G_{YX}(G_{XY}(x))$), distance consistency ($x \approx G_{YX}(G_{XY}(x))$), and our geometry consistency ($f(G_{XY}(x)) \approx G_{\tilde{X}\tilde{Y}}(f(x))$), respectively.

3. Preliminaries

Let $\mathcal{X}$ and $\mathcal{Y}$ be two domains with unpaired training examples $\{x_i\}_{i=1}^N$ and $\{y_j\}_{j=1}^M$, where x_i and y_j are drawn from the marginal distributions P_X and P_Y, where X and Y are two random variables associated with $\mathcal{X}$ and $\mathcal{Y}$, respectively. In the paper, we exploit style transfer without undesirable semantic distortions, and have two goals. First, we need to learn a mapping G_{XY} such that $G_{XY}(X)$ has the same distribution as Y, *i.e.*, $P_{G_{XY}(X)} \approx P_Y$. Second, the learned mapping function only changes the image style without distorting the semantic structures.

While many works have modeled the invertibility between G_{XY} and G_{YX} for convincing mappings since the success of CycleGAN, here we propose to enforce geometry consistency as a constraint that allows one-sided domain mapping. Let $f(\cdot)$ be a predefined geometric transformation. We can obtain two extra domains $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$ with examples $\{\tilde{x}_i\}_{i=1}^N$ and $\{\tilde{y}_j\}_{j=1}^M$ by applying $f(\cdot)$ on X and Y, respectively. We learn an additional translator $G_{\tilde{X}\tilde{Y}} : \tilde{X} \rightarrow \tilde{Y}$ while learning $G_{XY} : X \rightarrow Y$, and introduce our geometry-consistency constraint based on the predefined transformation such that the two networks can co-regularize each other. Our framework enforces that $G_{XY}(x)$ and $G_{\tilde{X}\tilde{Y}}(\tilde{x})$ should keep the same geometric transformation with the one between x and $\tilde{x}$, *i.e.*, $f(G_{XY}(x)) \approx G_{\tilde{X}\tilde{Y}}(\tilde{x})$, where $\tilde{x} = f(x)$. We denote the two adversarial discriminators as D_Y and $D_{\tilde{Y}}$ with respect to domains $\mathcal{Y}$ and $\tilde{\mathcal{Y}}$, respectively.

4. Proposed Method

We present our geometry-consistency constraint and GcGAN beginning with a review of the cycle-consistency constraint and the distance constraint. An illustration of the differences between these constraints is shown in Figure 2.

4.1. Unsupervised Constraints

Cycle-consistency constraint. Following the cycle-consistency assumption [28, 66, 61], through the translators $G_{XY} \circ G_{YX} : X \rightarrow Y \rightarrow X$ and $G_{YX} \circ G_{XY} : Y \rightarrow X \rightarrow Y$, the examples x and y in domain $\mathcal{X}$ and $\mathcal{Y}$ should recover the original images, *i.e.*, $x \approx G_{YX}(G_{XY}(x))$ and $y \approx G_{XY}(G_{YX}(y))$. Cycle consistency is implemented by a bidirectional reconstruction process that requires G_{XY} and G_{YX} to be jointly learned, as shown in Figure 2 (CycleGAN). The cycle consistency loss $\mathcal{L}_{cyc}(G_{XY}, G_{YX}, X, Y)$ takes the form as:

$$\begin{aligned}\mathcal{L}_{cyc} = &\mathbb{E}_{x\sim P_X}[\|G_{YX}(G_{XY}(x)) - x\|_1] \\ &+ \mathbb{E}_{y\sim P_Y}[\|G_{XY}(G_{YX}(y)) - y\|_1].\end{aligned} \tag{1}$$

Distance constraint. The assumption behind the distance constraint is that the distance between two examples x_i and x_j in domain X should be preserved after mapping to domain Y, *i.e.*, $d(x_i, x_j) \approx a \cdot d(G_{XY}(x_i), G_{XY}(x_j)) + b$, where $d(\cdot)$ is a predefined function to measure the distance between two examples and a and b are the linear coefficient and bias. In DistanceGAN [5], the distance consistency loss $\mathcal{L}_{dis}(G_{XY}, X, Y)$ is the exception to the absolute differ-

ences between distances:

$$\begin{aligned}\mathcal{L}_{dis} &= \mathbb{E}_{x_i,x_j\sim P_X}[|\phi(x_i,x_j)-\psi(x_i,x_j)|],\\ \phi(x_i,x_j) &= \frac{1}{\sigma_X}(\|x_i-x_j\|_1-\mu_X),\\ \psi(x_i,x_j) &= \frac{1}{\sigma_Y}(\|G_{XY}(x_i)-G_{XY}(x_j)\|_1-\mu_Y),\end{aligned} \tag{2}$$

where μ_X, μ_Y (σ_X, σ_Y) are the means (standard deviations) of distances of all the possible pairs of (x_i, x_j) within domain $\mathcal{X}$ and (y_i, y_j) within domain $\mathcal{Y}$, respectively.

4.2. Geometry-consistent Generative Adversarial Networks

Adversarial constraint. Taking G_{XY} as an example, an adversarial loss $\mathcal{L}_{gan}(G_{XY}, D_Y, X, Y)$ [21] enforces G_{XY} and D_Y to simultaneously optimize each other in an minimax game, *i.e.*, $\min_{G_{XY}}\max_{D_Y}\mathcal{L}_{gan}(G_{XY}, D_Y, X, Y)$. In other words, D_Y aims to distinguish real examples $\{y\}$ from translated samples $\{G_{XY}(x)\}$. By contrast, G_{XY} aims to fool D_Y so that D_Y can label a fake example $y' = G_{XY}(x)$ as a sample satisfying $y' \sim P_Y$. The objective can be expressed as:

$$\begin{aligned}\mathcal{L}_{gan} &= \mathbb{E}_{y\sim P_Y}[\log D_Y(y)]\\ &\quad+\mathbb{E}_{x\sim P_X}[\log(1-D_Y(G_{XY}(x)))].\end{aligned} \tag{3}$$

In the transformed domains $\tilde{\mathcal{X}}$ and $\tilde{\mathcal{Y}}$, we employ the adversarial loss $\mathcal{L}_{gan}(G_{\tilde{X}\tilde{Y}}, D_{\tilde{Y}}, \tilde{X}, \tilde{Y})$ that has the same form to $\mathcal{L}_{gan}(G_{XY}, D_Y, X, Y)$.

Geometry-consistency constraint. As shown in Figure 2 (GcGAN), given a predefined geometric transformation function $f(\cdot)$, we feed the images $x \in X$ and $\tilde{x} = f(x)$ into the translators G_{XY} and $G_{\tilde{X}\tilde{Y}}$, respectively. Following our geometry-consistency constraint, the outputs $y' = G_{XY}(x)$ and $\tilde{y}' = G_{\tilde{X}\tilde{Y}}(\tilde{x})$ should also satisfy $\tilde{y}' \approx f(y')$ like x and $\tilde{x}$. Considering both $f(\cdot)$ and the inverse geometric transformation function $f^{-1}(\cdot)$, our complete geometry consistency loss $\mathcal{L}_{geo}(G_{XY}, G_{\tilde{X}\tilde{Y}}, X, Y)$ has the following form:

$$\begin{aligned}\mathcal{L}_{geo} &= \mathbb{E}_{x\sim P_X}[\|G_{XY}(x)-f^{-1}(G_{\tilde{X}\tilde{Y}}(f(x)))\|_1]\\ &\quad+\mathbb{E}_{x\sim P_X}[\|G_{\tilde{X}\tilde{Y}}(f(x))-f(G_{XY}(x))\|_1].\end{aligned} \tag{4}$$

This geometry-consistency loss can be seen as a reconstruction loss that relies on the predefined geometric transformation function $f(\cdot)$. In this paper, we only take two common geometric transformations as examples, namely vertical flipping (*vf*) and 90° clockwise rotation (*rot*), to demonstrate the effectiveness of our geometry-consistency constraint. Note that, G_{XY} and $G_{\tilde{X}\tilde{Y}}$ have the same architecture and share all the parameters.

Full objective. By combining our geometry-consistency constraint with the standard adversarial constraint, a remarkable one-sided unsupervised domain mapping can be targeted. The full objective for our GcGAN $\mathcal{L}_{GcGAN}(G_{XY}, G_{\tilde{X}\tilde{Y}}, D_Y, D_{\tilde{Y}}, X, Y)$ will be:

$$\begin{aligned}\mathcal{L}_{GcGAN} &= \mathcal{L}_{gan}(G_{XY}, D_Y, X, Y)\\ &\quad+\mathcal{L}_{gan}(G_{\tilde{X}\tilde{Y}}, D_{\tilde{Y}}, X, Y)\\ &\quad+\lambda\mathcal{L}_{geo}(G_{XY}, G_{\tilde{X}\tilde{Y}}, X, Y),\end{aligned} \tag{5}$$

where λ ($\lambda = 20.0$ in all the experiments) is a trade-off hyperparameter to weight the contribution of $\mathcal{L}_{gan}$ and $\mathcal{L}_{geo}$ during the model training. Carefully tuning λ may give preferable results to specific translation tasks.

Network architecture. The full framework of our GcGAN is illustrated in Figure 2. Our experimental settings, network architectures, and learning strategies follow CycleGAN. We employ the same discriminator and generator as CycleGAN depending on the specific tasks. Specifically, the generator is a standard encoder-decoder, where the encoder contains two convolutional layers with stride 2 and several residual blocks [22] (6 / 9 blocks with respect to 128×128 / 256×256 of input resolution), and the decoder contains two deconvolutional layers also with stride 2. The discriminator distinguishes images at the patch level following PatchGANs [26, 33]. Like CycleGAN, we also use an identity mapping loss [55] in all of our experiments (except SVHN $\rightarrow$ MNIST), including our baseline (GAN alone). For other details, we use LeakyReLU as nonlinearity for the discriminators and instance normalization [57] to normalize convolutional feature maps.

Learning and inference. We use the Adam solver [29] with a learning rate of 0.0002 and coefficients of (0.5, 0.999), where the latter is used to compute running averages of gradients and their squares. The learning rate is fixed in the initial 100 epochs, and linearly decays to zero over the next 100 epochs. Following CycleGAN, the negative log likelihood objective is replaced with a more stable and effective least-squares loss [43] for $\mathcal{L}_{gan}$. The discriminator is updated with random samples from a history of generated images stored in an image buffer [54] of size 50. The generator and discriminator are optimized alternately. In the inference phase, we feed an image only into the learned generator G_{XY} to obtain a translated image.

5. Experiments

We apply our GcGAN to a wide range of applications and make both quantitative and qualitative comparisons with the baseline (GAN alone) and previous state-of-the-art methods including DistanceGAN and CycleGAN. We also study different ablations (based on *rot*) to analyze

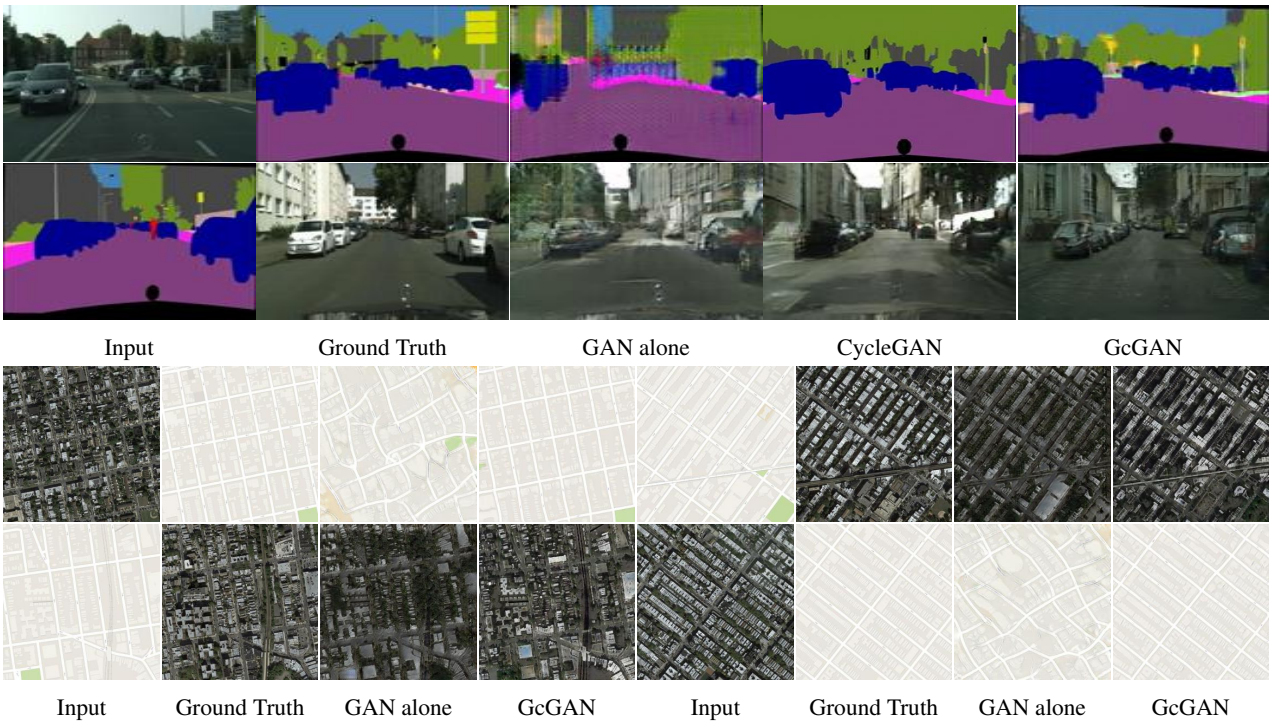


Figure 3: Qualitative comparison on Cityscapes (Parsing $\rightleftharpoons$ Image) and Google Maps (Map $\rightleftharpoons$ Aerial photo). GAN alone suffers from mode collapse. Translated images by GcGAN contain more details. $GcGAN = GAN\ alone + geometry\ consistency$.

our geometry-consistency constraint. Since adversarial networks are not always stable, every independent experiment could result in slightly different scores. The scores in the quantitative analysis are computed by the average on three independent experiments.

5.1. Quantitative Analysis

The results demonstrate that our geometry-consistency constraint can not only partially filter out the candidate solutions having mode collapse or semantic distortions and thus produce more sensible translations, but also compatible with other unsupervised constraints such as cycle consistency [66] and distance preservation [5].

Cityscapes. Cityscapes [12] contains 3975 image-label pairs, with 2975 used for training and 500 for validation (test in this paper). For a fair comparison with CycleGAN, the translators are trained at a resolution of 128×128 in an unaligned fashion. We evaluate our domain mappers using FCN scores and scene parsing metrics following previous works [41, 12, 66]. Specifically, for parsing $\rightarrow$ image, we assume that a high-quality translated image should produce qualitative semantic segmentation like real images when feeding it into a scene parser. Thus, we employ the pretrained FCN-8s [41] provided by pix2pix [26] to predict semantic labels for the 500 translated images. The label maps are then resized to the original resolution of 1024×2048 and compared against the ground truth labels using some standard scene parsing metrics including pixel accuracy, class accuracy, and mean IoU [41]. For image $\rightarrow$ parsing, since the fake labels are in the RGB format, we simply convert them into class-level labels using the nearest neighbor search strategy. In particular, we have 19 (category labels) + 1 (ignored label) categories for Cityscapes, each with a corresponding color value (RGB). For a pixel i in a translated parsing, we compute the distances between the 20 groundtruth color values and the color value of pixel i. The label of pixel i should be the one with the smallest distance. Then, the aforementioned metrics are used to evaluate our mapping on the 19 category labels.

The parsing scores are presented in Table 1. Our GcGAN outperforms the baseline (GAN alone) by a large margin. We take the average of pixel accuracy, class accuracy, and mean IoU as the final score for analysis [65], *i.e.*, score $=$ (pixel acc $+$ class acc $+$ mean IoU)$/3$. For image $\rightarrow$ parsing, GcGAN (32.6%) yields a slightly higher score than CycleGAN (32.0%). For parsing $\rightarrow$ image, GcGAN ($29.0\% \sim 29.5\%$) obtains a convincing improvement of $1.3\% \sim 1.8\%$ over distanceGAN (27.7%).

We next perform ablation studies to further discuss GcGAN. The scores are reported in Table 1. Specifically, GcGAN-*rot*-Seperate shows that the generator G_{XY} employed in GcGAN is sufficient to handle both the style transfers (without shape deformation) $X \rightarrow Y$ and $\tilde{X} \rightarrow \tilde{Y}$.

method	image → parsing			parsing → image		
	pixel acc	class acc	mean IoU	pixel acc	class acc	mean IoU
Benchmark Performance						
CoGAN [40]	0.45	0.11	0.08	0.40	0.10	0.06
BiGAN/ALI [15, 16]	0.41	0.13	0.07	0.19	0.06	0.02
SimGAN [54]	0.47	0.11	0.07	0.20	0.10	0.04
CycleGAN (Cycle) [66]	**0.58**	0.22	0.16	0.52	0.17	0.11
DistanceGAN [5]	-	-	-	0.53	0.19	0.11
GAN alone (baseline)	0.514	0.160	0.104	0.437	0.161	0.098
GcGAN-*rot*	0.574	**0.234**	0.170	**0.551**	**0.197**	**0.129**
GcGAN-*vf*	0.576	0.232	**0.171**	0.548	0.196	0.127
Ablation Studies (Robustness & Compatibility)						
$\mathcal{L}_{GcGAN}$ w/o $\mathcal{L}_{geo}$ ($\lambda = 0$)	0.486	0.163	0.102	0.396	0.148	0.088
$\mathcal{L}_{GcGAN}$ w/o $\mathcal{L}_{gan}(\tilde{X}, \tilde{Y})$	0.549	0.199	0.139	0.526	0.184	0.111
GcGAN-*rot*-Seperate	0.575	0.233	0.170	0.545	0.196	0.124
GcGAN-Mix-comb	0.573	0.229	0.168	0.545	0.197	0.128
GcGAN-Mix-rand	0.564	0.217	0.156	0.547	0.192	0.123
GcGAN-*rot* + Cycle	**0.587**	**0.246**	**0.182**	**0.557**	**0.201**	**0.132**

Table 1: **Parsing scores on Cityscapes.** $\mathcal{L}_{GcGAN}$: The objective in Eqn. 5 with *rot*. GcGAN-*rot*-Separate: G_{XY} and $G_{\tilde{X}\tilde{Y}}$ do not share parameters. GcGAN-Mix-comb: Training GcGAN with both *vf* and *rot* in each iteration. GcGAN-Mix-rand: Training GcGAN with randomly chosen *vf* and *rot* in each iteration. GcGAN-*rot* + Cycle: GcGAN-*rot* with the cycle-consistency constraint.

method	class acc (%)
Benchmark Performance	
DistanceGAN (Dist.) [5]	26.8
CycleGAN (Cycle) [66]	26.1
Self-Distance [5]	25.2
GcGAN-*rot*	32.5
GcGAN-*vf*	**33.3**
Ablation Studies (Compatibility)	
Cycle + Dist. [5]	18.0
GcGAN-*rot* + Dist.	**34.0**
GcGAN-*rot* + Cycle	33.8
GcGAN-*rot* + Dist. + Cycle	33.2

Table 2: Quantitative scores for SVHN → MNIST.

GcGAN-Mix-{comb, rand} demonstrate that persevering a geometric transformation can filter out most of the candidate solutions having mode collapse or undesired shape deformation, but preserving more ones can not leach more. Besides, GcGAN-Mix-rand performs slightly worse than GcGAN-Mix-comb. One of the possible reasons is that neither $X_{rot} \rightarrow Y_{rot}$ nor $X_{vf} \rightarrow Y_{vf}$ are sufficiently trained in the random case, which would decrease the effect of the aforementioned co-regularization mechanism. For GcGAN-*rot* + Cycle, we set the trade-off parameter for $\mathcal{L}_{cyc}$ to 10.0 as published in CycleGAN. The consistent improvement is a credible support that our geometry-consistency constraint is compatible with the widely-used cycle-consistency constraint. Moreover, when setting $\lambda = 0$ in $\mathcal{L}_{GcGAN}$, both G_{XY} and G_{YX} perform badly. One of the possible reasons is that, without the geometry consistency constraint, jointly modeling $X \rightarrow Y$ and $\tilde{X} \rightarrow \tilde{Y}$ with the shared generator G_{XY} would decrease the performance due to domain diversities caused by the geometric transformations. When removing $\mathcal{L}_{gan}(G_{\tilde{X}\tilde{Y}}, D_{\tilde{Y}})$, the obtained scores are much higher than baseline (GAN alone) because Y' can partially correct $\tilde{Y}'$ so that G_{XY} is able to handle the mapping $\tilde{X} \rightarrow \tilde{Y}$, and then $\tilde{Y}'$ can constrain the mapping $X \rightarrow Y$. As the analysis, when learning both $\mathcal{L}_{gan}(G_{XY}, D_Y)$ and $\mathcal{L}_{gan}(G_{\tilde{X}\tilde{Y}}, D_{\tilde{Y}})$ with $\mathcal{L}_{geo}$, the co-regularization help generate preferable translations.

SVHN → MNIST. We apply our approach to the SVHN → MNIST translation task. The translation models are trained on 73257 and 60000 training images of resolution 32×32 contained in the SVHN and MNIST training sets, respectively. The experimental settings follow DistanceGAN [5], including the default trade-off parameters for $\mathcal{L}_{cyc}$ and $\mathcal{L}_{dis}$. We compare our GcGAN with both DistanceGAN and CycleGAN in this translation task. To obtain quantitative results, we feed the translated images into a pretrained classifier trained on the MNIST training split, as done in [5].

Classification accuracies are reported in Table 2. Both GcGAN-*rot* and GcGAN-*vf* outperform DistanceGAN and CycleGAN by a large margin (about $6\% \sim 7\%$). From the ablations, adding our geometry-consistency constraint to current unsupervised domain mapping frameworks will achieve different levels of improvements against the original ones. Note that, it seems that the distance-preservation

constraint is not compatible with the cycle-consistency constraint on this task, but our geometry-consistency constraint can improve both ones.

DistanceGAN [5] GcGAN

Figure 4: Qualitative comparison for SVHN → MNIST.

Google Maps. We obtain 2194 (map, aerial photo) pairs of images in and around New York City from Google Maps [26], and split them into training and test sets with 1096 and 1098 pairs, respectively. We train Map ⇌ Aerial photo translators with an image size of 256×256 using the training set in an unsupervised manner (unpaired) by ignoring the pair information. For Aerial photo → Map, we make comparisons with CycleGAN using average RMSE and pixel accuracy (%). Given a pixel i with the ground-truth RGB value (r_i, g_i, b_i) and the predicted RGB value (r'_i, g'_i, b'_i), if $max(|r_i - r'_i|, |g_i - g'_i|, |b_i - b'_i|) < \delta$, we consider this is an accurate prediction. Since maps only contain a limited number of different RGB values, it is reasonable to compute pixel accuracy using this strategy ($\delta_1 = 5$ and $\delta_2 = 10$ in this paper). For Map → Aerial photo, we only show some qualitative results in Figure 3.

method	RMSE	acc (δ_1)	acc (δ_2)
Benchmark Performance			
CycleGAN [66]	**28.15**	**41.8**	**63.7**
GAN alone (baseline)	33.27	19.3	42.0
GcGAN-*rot*	28.31	41.2	63.1
GcGAN-*vf*	28.50	37.3	58.9
Ablation Studies (Robustness & Compatibility)			
GcGAN-*rot*-Separate	30.25	40.7	60.8
GcGAN-Mix-comb	**27.98**	**42.8**	**64.6**
GcGAN-*rot* + Cycle	28.21	40.6	63.5

Table 3: Quantitative scores for Aerial photo → Map.

From the scores presented in Table 3, GcGAN produces superior translations to the baseline (GAN alone). In particular, GcGAN yields an $18.0\% \sim 21.9\%$ improvement over the baseline with respect to pixel accuracy when $\delta = 5.0$, demonstrating that the fake maps obtained by our GcGAN contain more details. In addition, GcGANs achieve competitive scores compared with CycleGAN.

5.2. Qualitative Evalutation

The qualitative results are shown in Figure 3, Figure 4, and Figure 5. Our geometry-consistency constraint improve the training of GAN alone, and helps to generate empirically more impressive translations on various applications. The following applications are trained in the image size of 256×256 with the *rot* geometric transformation.

Horse → Zebra. We apply GcGAN to the widely studied object transfiguration application task, *i.e.*, Horse → Zebra. The images are randomly sampled from ImageNet [13] using the keywords (*i.e.*, wild horse and zebra). The numbers of training images are 939 and 1177 for horse and zebra, respectively. We find that training GcGAN without parameter sharing would produce preferable translations for the task.
Synthetic ⇌ Real. We employ the 2975 training images from Cityscapes as the real-world scenes, and randomly select 3060 images from SYNTHIA-CVPR16 [49], which is a virtual urban scene benchmark, as the synthetic images.
Summer ⇌ Winter. The images used for the season translation tasks are provided by CycleGAN. The training set sizes for Summer and Winter are 1273 and 854.
Photo ⇌ Artistic Painting. We translate natural images to artistic paintings with different art styles, including Monet, Cezanne, Van Gogh, and Ukiyo-e. We also perform GcGAN on the translation task of Monet's paintings → photographs. We use the photos and paintings (Monet: 1074, Cezanne: 584, Van Gogh: 401, Ukiyo-e: 1433, and Photographs: 6853) collected by CycleGAN for training.
Day ⇌ Night. We randomly extract 4500 training images for both Day and Night from the 91 webcam sequences captured by [30].

6. Conclusion

In this paper, we propose to enforce geometry consistency as a constraint for unsupervised domain mapping, which can be viewed as a predefined geometric transformation $f(\cdot)$ preserving the geometry of a scene. The geometry-consistency constraint makes the translation networks on the original images and transformed images co-regularize each other, which not only provides an effective remedy to the mode collapse problem suffered by standard GANs, but also reduces the semantic distortions in the translation. We evaluate our model, *i.e.*, the geometry-consistent generative adversarial network (GcGAN), both qualitatively and quantitatively in various applications. The experimental results demonstrate that GcGAN achieves competitive and sometimes even better translations than the state-of-the-art methods including DistanceGAN and CycleGAN. Finally, our geometry-consistency constraint is compatible with other well-studied unsupervised constraints.

7. Acknowledgement

This research was supported by Australian Research Council Projects FL-170100117, DP-180103424, and IH-

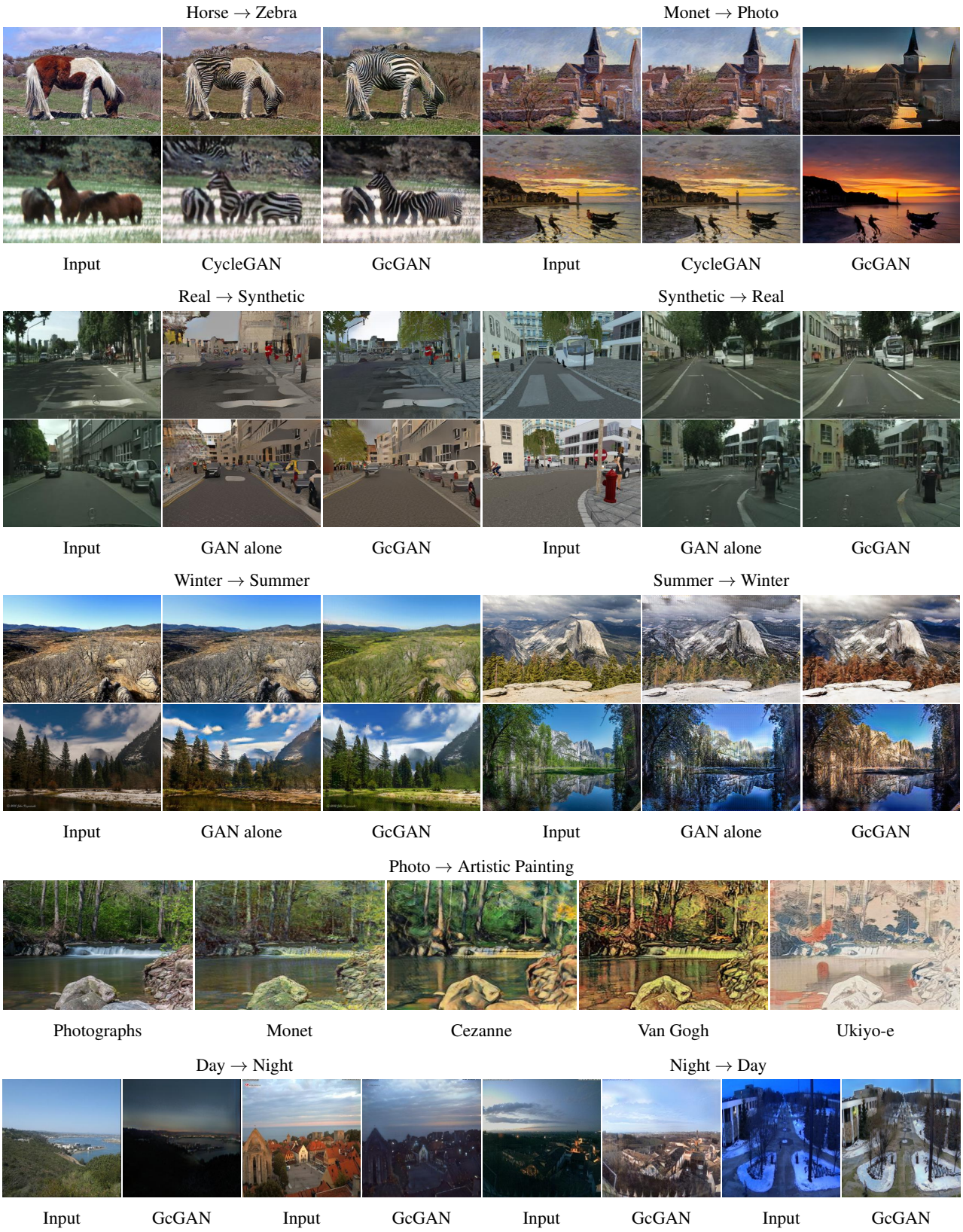


Figure 5: Qualitative results on different applications, including Horse → Zebra, Monet → Photo, Synthetic ⇌ Real, Summar ⇌ Winter, Photo → Artist Painting, and Day ⇌ Night. GcGAN has the potential to produce realistic images. Zoom in for better view.

180100002. This work was partially supported by SAP SE and CNRS INS2I-JCJC-INVISANA. This work is partially supported by NIH Award Number 1R01HL141813-01, NSF 1839332 Tripod+X, and SAP SE. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. We were also grateful for the computational resources provided by Pittsburgh Super Computing grant number TG-ASC170024.

References

[1] Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. Augmented cyclegan: Learning many-to-many mappings from unpaired data. *ICML*, 2018. 2

[2] Asha Anoosheh, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Combogan: Unrestrained scalability for image domain translation. In *CVPRW*, 2018. 2

[3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 2

[4] Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. Multi-content gan for few-shot font style transfer. In *CVPR*, 2018. 2

[5] Sagie Benaim and Lior Wolf. One-sided unsupervised domain mapping. In *NIPS*, 2017. 1, 2, 3, 5, 6, 7

[6] Sagie Benaim and Lior Wolf. One-shot unsupervised cross domain translation. *NIPS*, 2018. 2

[7] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *NIPS*, 2016. 2

[8] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In *CVPR*, 2018. 2

[9] Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stereoscopic neural style transfer. In *CVPR*, 2018. 2

[10] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 2

[11] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018. 2

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR)*, 2016. 5

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 7

[14] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a? laplacian pyramid of adversarial networks. In *NIPS*, 2015. 2

[15] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016. 6

[16] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016. 6

[17] Aaron Gokaslan, Vivek Ramanujan, Daniel Ritchie, Kwang In Kim, and James Tompkin. Improving shape deformation in unsupervised image-to-image translation. *ECCV*, 2018. 2

[18] Mingming Gong, Kun Zhang, Biwei Huang, Clark Glymour, Dacheng Tao, and Kayhan Batmanghelich. Causal generative domain adaptation networks. *arXiv preprint arXiv:1804.04333*, 2018. 2

[19] Mingming Gong, Kun Zhang, Tongliang Liu, Dacheng Tao, Clark Glymour, and Bernhard Schölkopf. Domain adaptation with conditional transferable components. In *ICML*, pages 2839–2848, 2016. 2

[20] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. *NIPS*, 2018. 2

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 2, 4

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[23] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *ICML*, 2018. 2

[24] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. *ECCV*, 2018. 2

[25] Takahiro Isokane, Fumio Okura, Ayaka Ide, Yasuyuki Matsushita, and Yasushi Yagi. Probabilistic plant modeling via multi-view image-to-image translation. *CVPR*, 2018. 2

[26] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 2, 4, 5, 7

[27] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 2

[28] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*, 2017. 1, 2, 3

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[30] Pierre-Yves Laffont, Zhile Ren, Xiaofeng Tao, Chao Qian, and James Hays. Transient attributes for high-level understanding and editing of outdoor scenes. *ACM TOG*, 33(4):149, 2014. 7

[31] Christian Ledig, Lucas Theis, Ferenc Huszar, 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 *CVPR*, 2017. 2

[32] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *ECCV*, 2018. 2

[33] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *ECCV*, 2016. 4

[34] Minjun Li, Haozhi Huang, Lin Ma, Wei Liu, Tong Zhang, and Yugang Jiang. Unsupervised image-to-image translation with stacked cycle-consistent adversarial networks. *ECCV*, 2018. 2

[35] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, pages 624–639, 2018. 2

[36] Xiaodan Liang, Hao Zhang, and Eric P Xing. Generative semantic manipulation with contrasting gan. *NIPS*, 2017. 2

[37] Jianxin Lin, Yingce Xia, Tao Qin, Zhibo Chen, and Tie-Yan Liu. Conditional image-to-image translation. In *CVPR*, 2018. 2

[38] Alexander Liu, Yen-Chen Liu, Yu-Ying Yeh, and Yu-Chiang Frank Wang. A unified feature disentangler for multi-domain image translation and manipulation. *NIPS*, 2018. 2

[39] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017. 2

[40] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *NIPS*, 2016. 2, 6

[41] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 5

[42] Shuang Ma, Jianlong Fu, Chang Wen Chen, and Tao Mei. Da-gan: Instance-level image translation by deep attention generative adversarial networks. In *CVPR*, 2018. 2

[43] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *ICCV*, 2017. 4

[44] Youssef A Mejjati, Christian Richardt, James Tompkin, Darren Cosker, and Kwang In Kim. Unsupervised attention-guided image to image translation. *NIPS*, 2018. 2

[45] Aäron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. In *NIPS*, 2016. 2

[46] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 1, 2

[47] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016. 2

[48] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 2

[49] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016. 7

[50] Amélie Royer, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Moressi, Forrester Cole, and Kevin Murphy. Xgan: Unsupervised image-to-image translation for many-to-many mappings. *ICLR*, 2018. 2

[51] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, 2016. 2

[52] Falong Shen, Shuicheng Yan, and Gang Zeng. Neural style transfer via meta networks. In *CVPR*, 2018. 2

[53] Lu Sheng, Ziyi Lin, Jing Shao, and Xiaogang Wang. Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In *CVPR*, 2018. 2

[54] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017. 4, 6

[55] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation, 2016. 4

[56] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. Texture networks: feed-forward synthesis of textures and stylized images. In *ICML*, 2016. 2

[57] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 4

[58] Chao Wang, Haiyong Zheng, Zhibin Yu, Ziqiang Zheng, Zhaorui Gu, and Bing Zheng. Discriminative region proposal adversarial networks for high-quality image-to-image translation. In *ECCV*, 2018. 1, 2

[59] 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. *arXiv preprint arXiv:1711.11585*, 2017. 1, 2

[60] Yaxing Wang, Joost van de Weijer, and Luis Herranz. Mix and match networks: encoder-decoder alignment for zero-pair image translation. In *CVPR*, 2018. 2

[61] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *CVPR*, 2017. 1, 2, 3

[62] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stack-gan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. 2

[63] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *ICML*, pages 819–827, 2013. 2

[64] Yexun Zhang, Ya Zhang, and Wenbin Cai. Separating style and content for generalized style transfer. In *CVPR*, 2018. 2

[65] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 5

[66] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7

[67] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NIPS*, 2017. 2

8. Network Architecture

The generator and discriminator (except for SVHN → MNIST) presented before are shown in Tab. 4. For convenience, we use the following abbreviation: C = Feature channel, K = Kernel size, S = Stride size, Deconv/Conv = Deconvolutional/Convolutional layer, and ResBlk = A residual block.

Generator				
Index	Layer	C	K	S
1	Conv + ReLU	64	7	1
2	Conv + ReLU	128	3	2
3	Conv + ReLU	256	3	2
4	ResBlk + ReLU	256	3	1
5	ResBlk + ReLU	256	3	1
6	ResBlk + ReLU	256	3	1
7	ResBlk + ReLU	256	3	1
8	ResBlk + ReLU	256	3	1
9	ResBlk + ReLU	256	3	1
10	ResBlk + ReLU	256	3	1
11	ResBlk + ReLU	256	3	1
12	ResBlk + ReLU	256	3	1
12	Deconv + ReLU	128	3	2
13	Deconv + ReLU	64	3	2
14	Conv	3	7	1
15	Tanh	-	-	-
Discriminator				
1	Conv + LeakyReLU	64	4	2
2	Conv + LeakyReLU	128	4	2
3	Conv + LeakyReLU	256	4	2
4	Conv + LeakyReLU	512	4	1
5	Conv	512	4	1

Table 4: The generator and discriminator used in our experiments (256×256).

The network architecture for SVHN → MNIST is reported in Tab. 5.

Generator				
Index	Layer	C	K	S
1	Conv + LeakyReLU	64	4	2
2	Conv + LeakyReLU	128	4	2
3	Conv + LeakyReLU	128	3	1
4	Conv + LeakyReLU	128	3	1
5	Deconv + LeakyReLU	64	4	2
5	Deconv + LeakyReLU	1	4	2
15	Tanh	-	-	-
Discriminator				
1	Conv + LeakyReLU	64	4	2
2	Conv + LeakyReLU	128	4	2
3	Conv + LeakyReLU	256	4	2
4	Conv + LeakyReLU	512	4	1
5	Conv	512	4	1

Table 5: The network architecture for SVHN → MNIST.

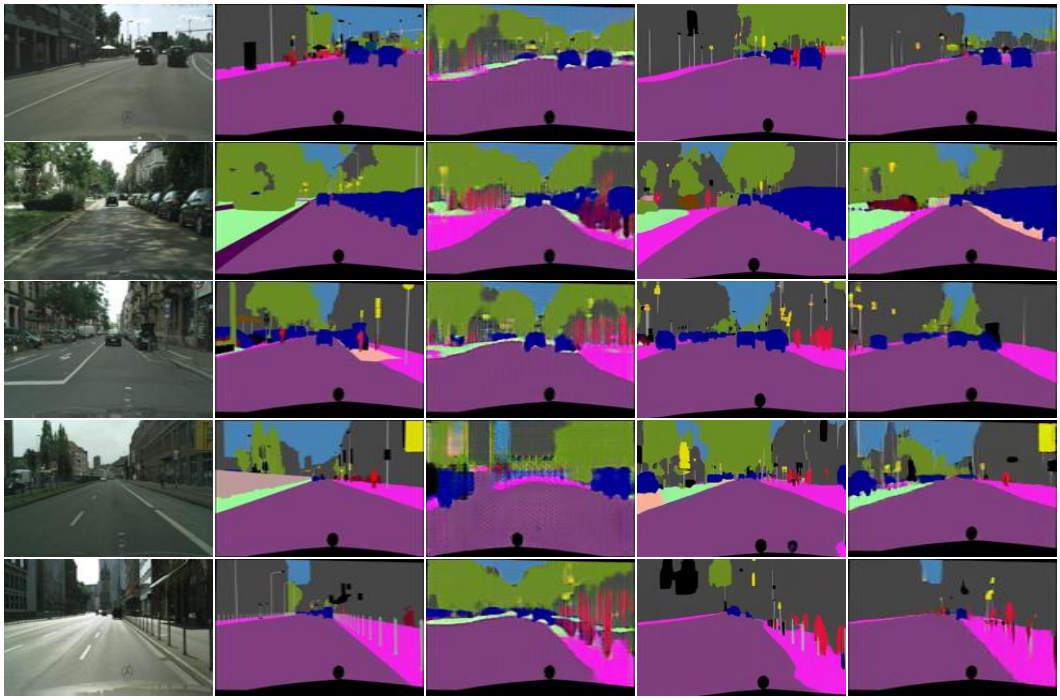
9. Other Qualitative Results

We show other qualitative results for the fourteen applications in our submission (Figure 6 to Figure 14). Our geometry-consistency constraint makes the translation networks on the original images and transformed images co-regularize each other, which not only provides an effective remedy to the mode collapse problem suffered by standard GANs, but also reduces the semantic distortions in the translation. As a result, GcGAN can produce empirically impressive translations.

10. Failure Cases

Unluckily, GcGAN cannot guarantee reasonable translations for all the cases as previous works. Thus, more assumptions and constraints should be investigated to improve unsupervised domain mapping. Some failure cases are shown in Figure 15.

Photo → Parsing



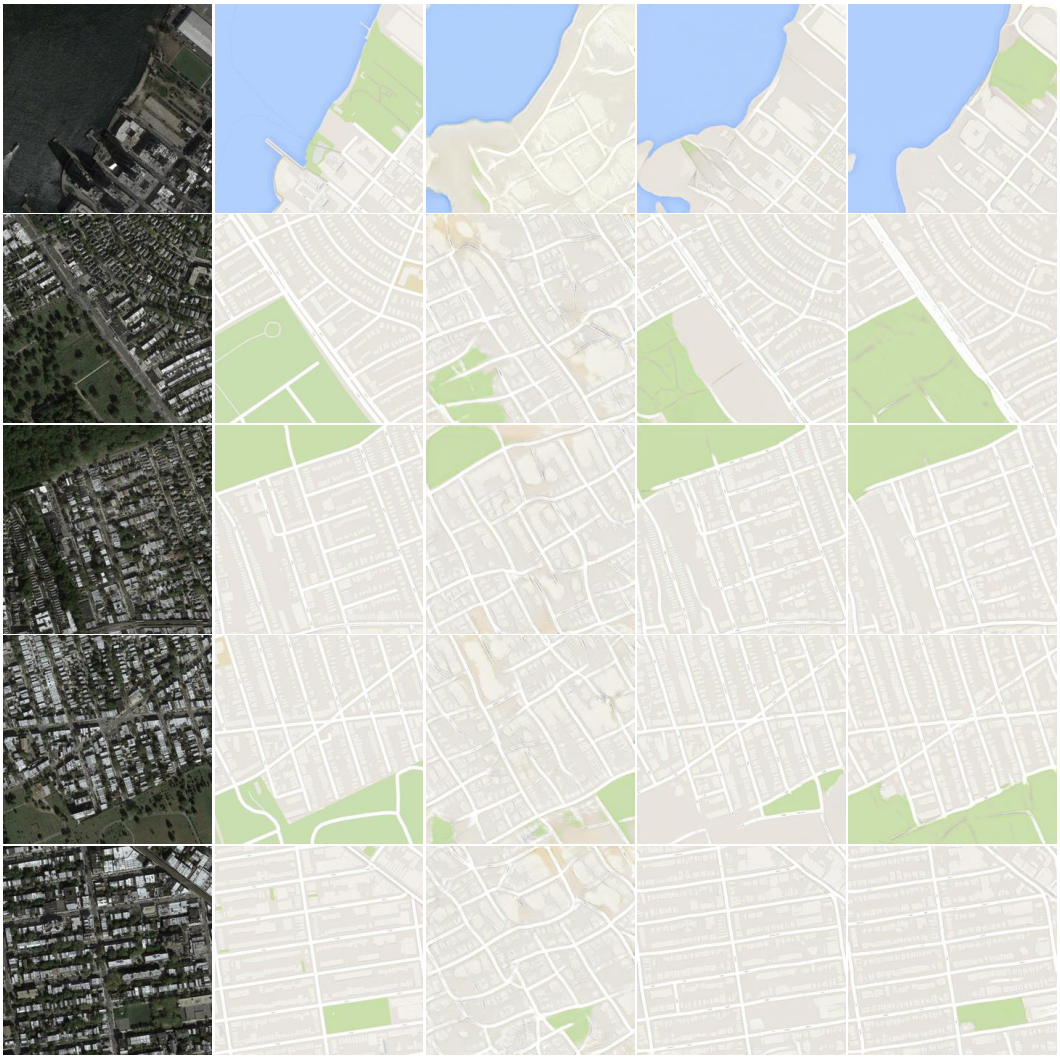
Input | Ground Truth | GAN alone | CycleGAN | GcGAN

Parsing → Photo

Input | Ground Truth | GAN alone | CycleGAN | GcGAN

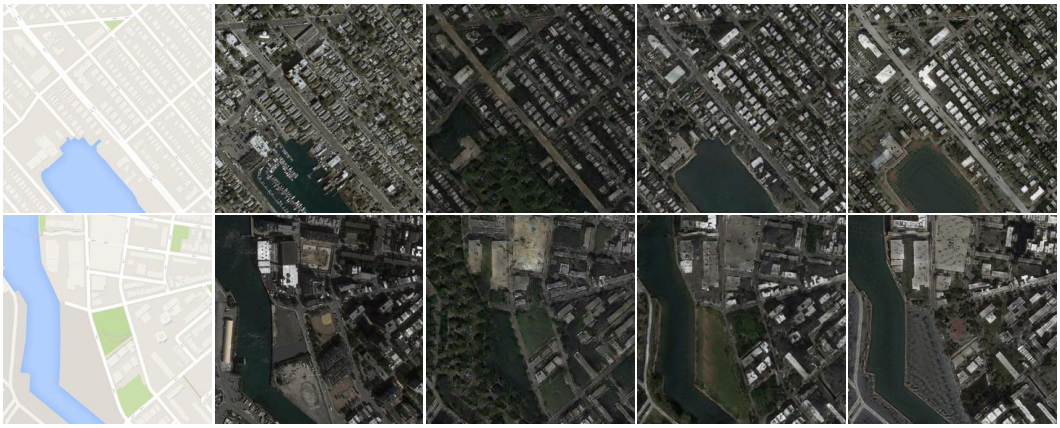
Figure 6: **Cityscapes (Parsing ⇌ Image).** The results for CycleGAN [62] are produced by the officially provided pretrained PyTorch models. GcGAN denotes GcGAN-*rot*.

Aerial photo → Map



Input | Ground Truth | GAN alone | CycleGAN | GcGAN

Map → Aerial photo



Input | Ground Truth | GAN alone | CycleGAN | GcGAN

Figure 7: **Google Mpas (Aerial photo ⇌ Map).** For Map → Aerial photo, GcGAN produces competitive translations compared with CycleGAN [62].

SVHN → MNIST

CycleGAN (24/64)

DistanceGAN (26/64)

GcGAN (35/64)

Figure 8: **SVHN → MNIST.** The qualitative results for both CycleGAN [62] and DistanceGAN [5] come from DistanceGAN [5]. The correct translations are about 24, 26, and 35 for CycleGAN, DistanceGAN, and GcGAN, respectively.

Horse → Zebra

Input | GAN alone | CycleGAN | GcGAN

Figure 9: **Horse → Zebra.** For this task, GcGAN generates slightly better translations for some images, but can not perform better than CycleGAN [62] generally.

Monet → Photo

Input | GAN alone | CycleGAN | GcGAN

Figure 10: **Monet → Photo.** GcGAN is superior in generating realistic images. Zoom in for better view.

Synthetic → Real

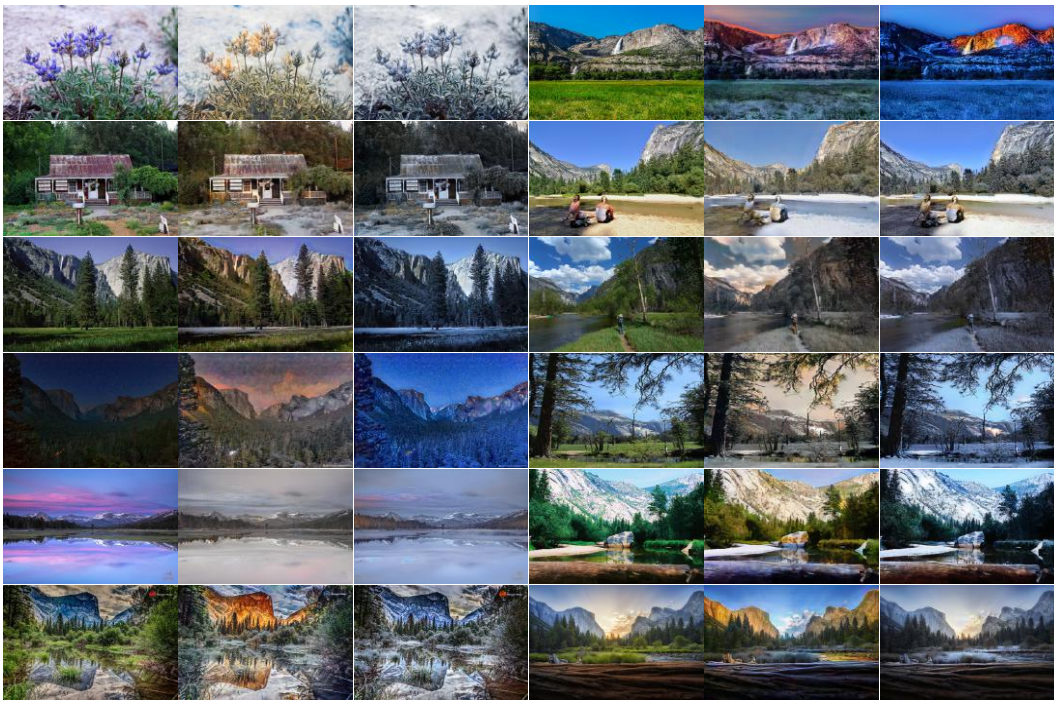
Input | GAN alone | CycleGAN | GcGAN

Real → Synthetic

Input | GAN alone | CycleGAN | GcGAN

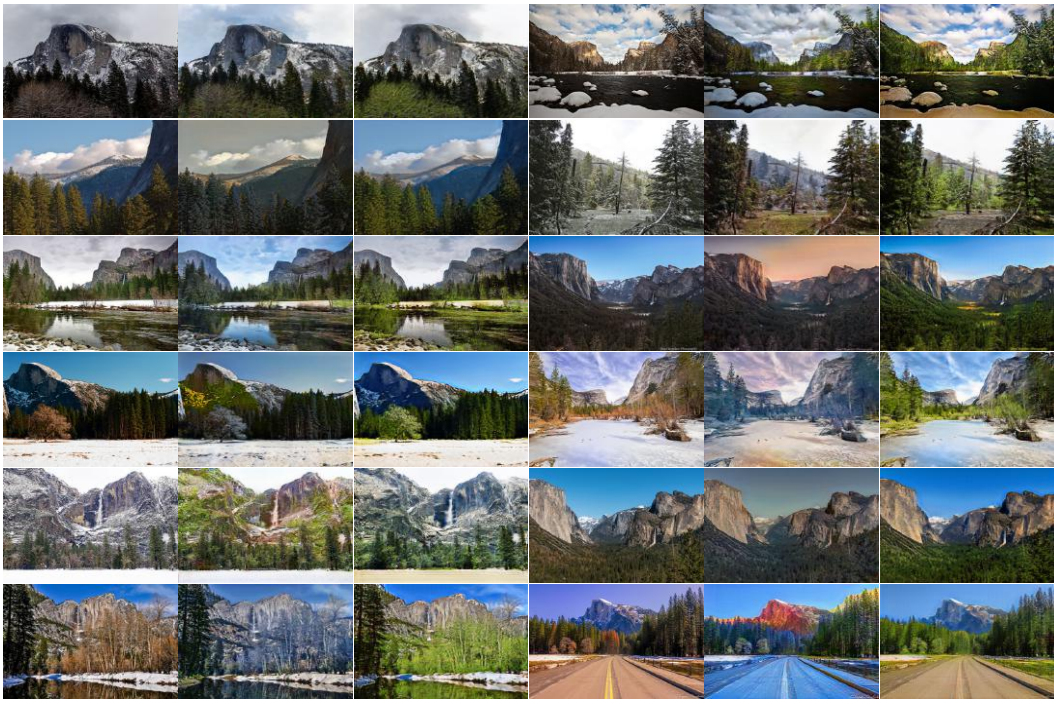
Figure 11: **Synthetic ⇌ Real.** We train CycleGAN [62] using the released PyTorch codes. The results produced by GcGAN contain more details. Zoom in for better view.

Summer → Winter



Input CycleGAN GcGAN Input CycleGAN GcGAN

Winter → Summer



Input CycleGAN GcGAN Input CycleGAN GcGAN

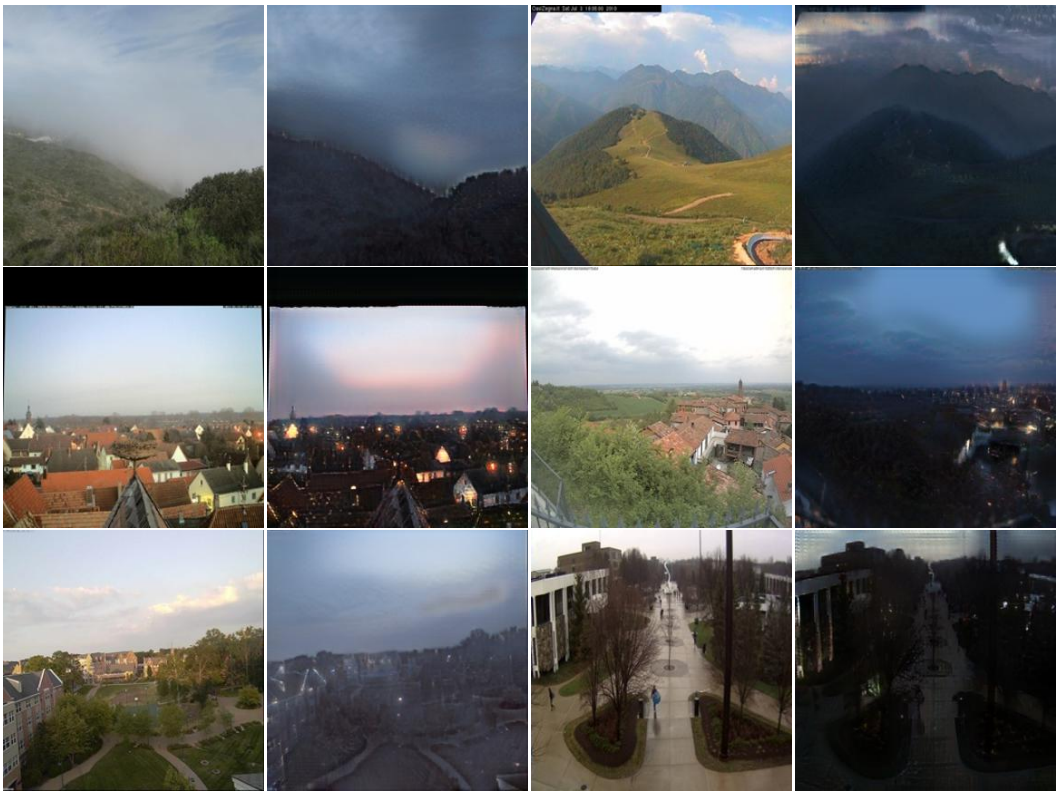
Figure 12: **Summer ⇌ Winter.** Here, GcGAN represents GcGAN-*rot*-Seperate. Zoom in for better view.

Photographs → Artist paintings

Input | Monet | Van Gogh | Cezanne | U-kiyoe

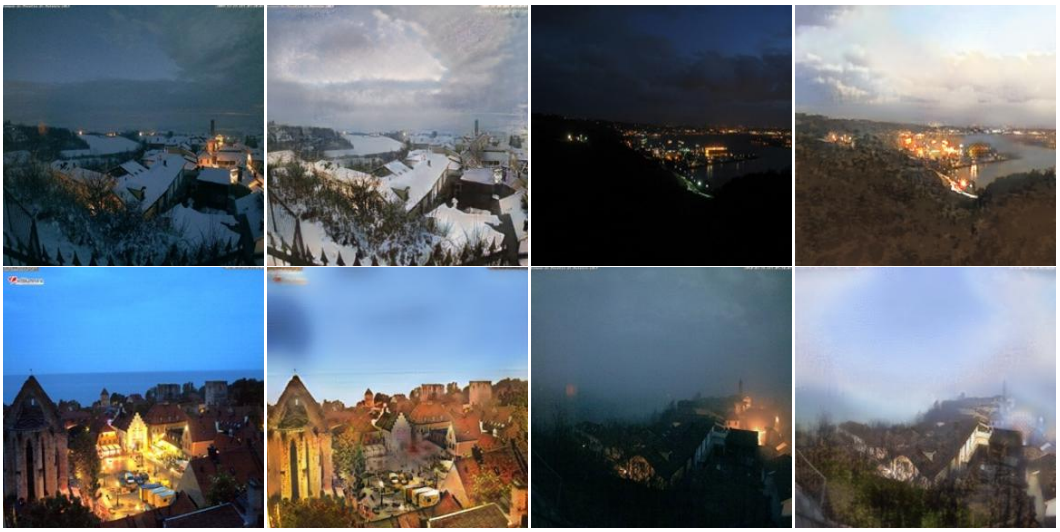
Figure 13: **Photographs → Artist paintings.** We translate a photo to the artistic styles of Monet, Van Gogh, Cezanne, and U-kiyoe.

Day → Night



Input GcGAN Input GcGAN

Night → Day



Input GcGAN Input GcGAN

Figure 14: **Day ⇌ Night.**

Failure Cases

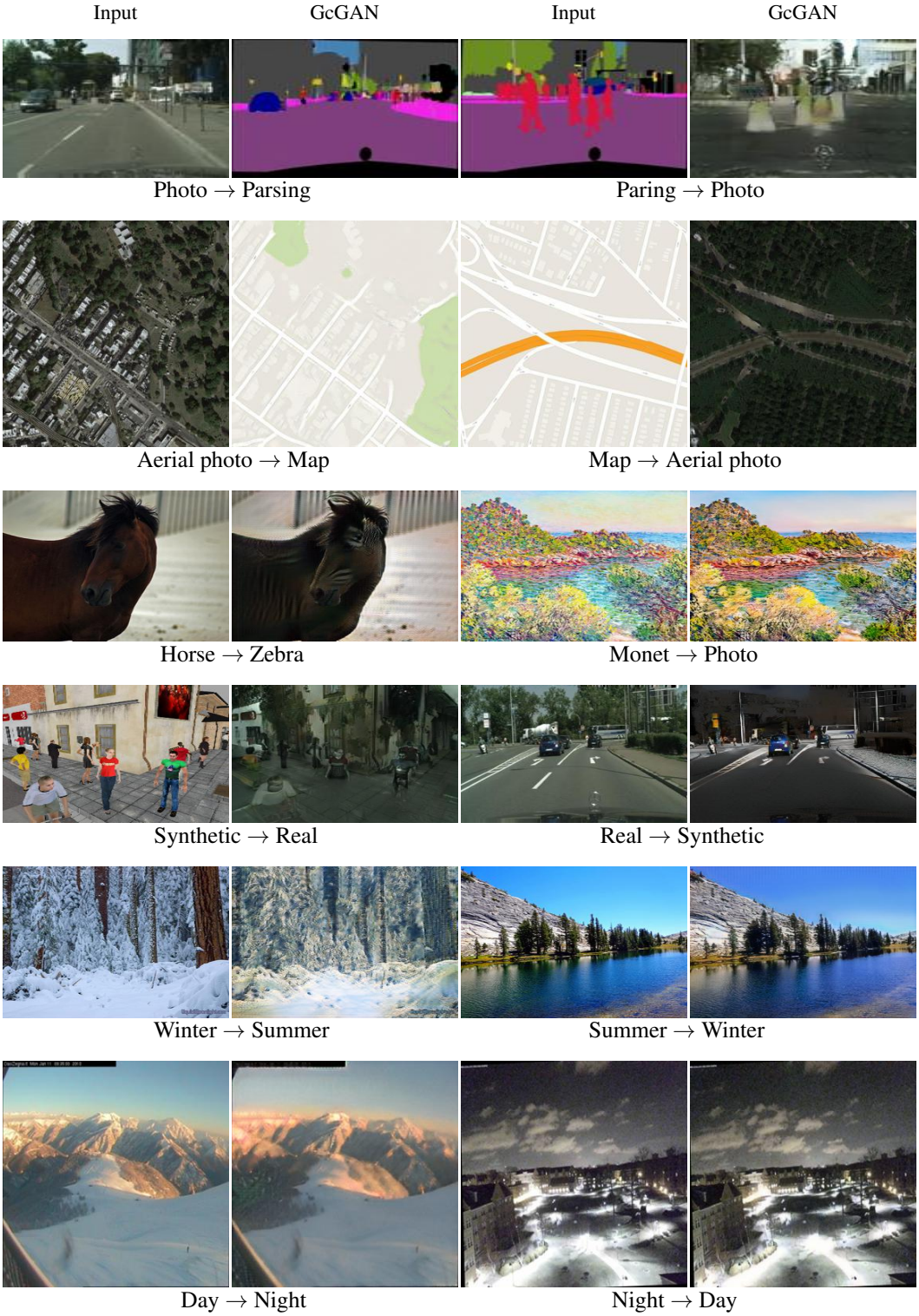


Figure 15: **Failure Cases.** GcGAN cannot guarantee reasonable translations for all the cases as previous works. Thus, more assumptions and constraints should be investigated to improve unsupervised domain mapping.