Sem-GAN: Semantically-Consistent Image-to-Image Translation

Cherian, A.; Sullivan, A.

Abstract

Unpaired image-to-image translation is the problem of mapping an image in the source domain to one in the target domain, without requiring corresponding image pairs. To ensure the translated images are realistically plausible, recent works, such as Cycle-GAN, demands this mapping to be invertible. While, this requirement demonstrates promising results when the domains are unimodal, its performance is unpredictable in a multi-modal scenario such as in an image segmentation task. This is because, invertibility does not necessarily enforce semantic correctness. To this end, we present a semantically-consistent GAN framework, dubbed Sem-GAN, in which the semantics are defined by the class identities of image segments in the source domain as produced by a semantic segmentation algorithm. Our proposed framework includes consistency constraints on the translation task that, together with the GAN loss and the cycle-constraints, enforces that the images when translated will inherit the appearances of the target domain, while (approximately) maintaining their identities from the source domain. We present experiments on several imageto-image translation tasks and demonstrate that Sem-GAN improves the quality of the translated images significantly, sometimes by more than 20% on the FCN score. Further, we show that semantic segmentation models, trained with synthetic images translated via Sem-GAN, leads to significantly better segmentation results than other variants.

IEEE Winter Conference on Applications of Computer Vision (WACV)

# Sem-GAN: Semantically-Consistent Image-to-Image Translation

Anoop Cherian   Alan Sullivan

Mitsubishi Electric Research Labs (MERL), Cambridge, MA

{cherian, sullivan}@merl.com

Abstract

Unpaired image-to-image translation is the problem of mapping an image in the source domain to one in the target domain, without requiring corresponding image pairs. To ensure the translated images are realistically plausible, recent works, such as Cycle-GAN, demands this mapping to be invertible. While, this requirement demonstrates promising results when the domains are unimodal, its performance is unpredictable in a multi-modal scenario such as in an image segmentation task. This is because, invertibility does not necessarily enforce semantic correctness. To this end, we present a semantically-consistent GAN framework, dubbed Sem-GAN, in which the semantics are defined by the class identities of image segments in the source domain as produced by a semantic segmentation algorithm. Our proposed framework includes consistency constraints on the translation task that, together with the GAN loss and the cycle-constraints, enforces that the images when translated will inherit the appearances of the target domain, while (approximately) maintaining their identities from the source domain. We present experiments on several image-to-image translation tasks and demonstrate that Sem-GAN improves the quality of the translated images significantly, sometimes by more than 20% on the FCN score. Further, we show that semantic segmentation models, trained with synthetic images translated via Sem-GAN, leads to significantly better segmentation results than other variants.

1. Introduction

The recent advancements in several fundamental computer vision tasks [17, 4, 5] are unequivocally associated with the availability of huge annotated datasets for training deep architectures of increasing sophistication [9, 18, 25]. However, collecting or annotating such datasets are often challenging or expensive. An alternative, which is cheap and manageable, is to resort to computer gaming software [8, 39, 40] to render realistic virtual worlds; such software could supply unlimited amounts of training data and could also simulate real world scenarios that may be oth-
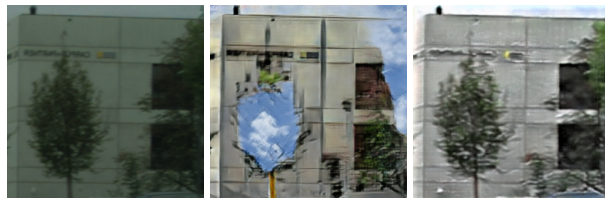
(a) Real→Synthetic   (b) Cycle GAN   (c) Sem-GAN (Ours)

Figure 1. Translation of an image from the Cityscapes dataset (left-most) to a synthetic domain. The state-of-the-art model Cycle-GAN [57] incorrectly maps 'trees' to 'sky'. Using Sem-GAN, that enforces semantic consistency in the translation process, results in more realistic translations.

erwise difficult to observe. Unfortunately, using data from a synthetic domain often introduces biases in the learned model, resulting in a domain shift that might hurt the performance of a downstream task [34, 50].

A standard way to account for domain shift is to adapt the synthetic images so that their statistics match that of the real domain. This is the classical domain adaptation problem [3, 15, 13], commonly called image-to-image translation when done on image pixels [41, 57, 47]. Most such translation algorithms require corresponding image pairs from the two domains [20, 12, 21]. However, thanks to generative adversarial networks (GAN), recent years have seen breakthroughs in unsupervised translations that do not require paired examples, instead need only sets of examples from the two domains, which are much easier to obtain [47, 27, 26, 57]. However, the lack of correspondences results in a harder problem to solve, as one needs to estimate the image distributions and the adaptation function from the two sets – an ill-posed problem, since an infinite number of marginal distributions are possible that could have generated the finite examples in each of these sets.

To ameliorate this intractability, recent methods make assumptions on the problem domains or the mapping function. For example, in Liu et al. [27, 26], the two domains are assumed to share a common latent space. In Cycle GAN [57], the mapping is assumed invertible, i.e., a translated image when mapped back must be the same as the input image. Learning such a mapping could avoid some

well-known pitfalls in GAN training such as mode collapse, and could allow learning a bijective mapping between the domains.

When dealing with real-world tasks, bijective mappings might not be sufficient to generate meaningful translations. For example, consider the real-to-synthetic translation task depicted in Figure 1. Here, the Cycle-GAN is trained on real images from the Cityscapes dataset [7] and synthetic road scene images are produced by the Mitsubishi Precision Simulator.[1] As is clear, Cycle-GAN has learned an incorrect mapping between the classes 'trees' and 'sky', resulting in implausible translations. Nevertheless, such a mapping is invertible as per the Cycle-GAN cost function. This problem happens because, in typical translation tasks the images in the two sets are assumed to be samples from the joint distribution of all their respective sub-classes (object segments), and the translation is a mapping between such joint distributions. Such a mapping (even with cycles) does not ensure that the marginals of the sub-classes (modes) are assigned correctly (e.g., sky→sky). To this end, we try to look beyond cyclic dependencies, and incorporate semantic consistency as well in the translation process.

In this paper, we propose a novel GAN architecture for pixel-level domain adaptation, coined semantically-consistent GAN (Sem-GAN), that takes as input two unpaired sets, each set consisting of tuples of images and their semantic segment labels, and learns a domain mapping function by optimizing over the standard min-max generator-discriminator GAN objective [16]. However, differently from the standard GAN, we pass the generated images through a semantic segmentation network [29] (in addition to the discriminator); this network trained to segment images in the target domain. It is expected that if the translation is ideal, the objects being translated will inherit their appearance from the target domain while maintaining their identity from the source domain. For example, when a 'car' class in one domain is translated, a segmentation model trained to identify the 'car' class in the target domain should identify the translated object in the same class. We use the discrepancy between the ground-truth semantic classes and their predictions as error cues (via cross-entropy loss) for improving the generator via backpropagation.

Given that semantic segmentation by itself is a difficult (and unsolved) computer vision problem, a natural question would be how useful it can be to include such an imperfect module in a GAN setup. Our experiments show that using a segmentation scheme that performs reasonably well (such as FCN [29]) is sufficient to ensure semantic consistency, leading to better translations. Further, we also propose to train the segmentation module jointly with the discriminator; as a result, its accuracy improves along with

the generator-discriminator pair. A careful reader would have noticed, we are in fact solving a *chicken-egg problem*: on the one hand, we use GAN to improve semantic segmentation, while on the other hand, we are using segmentation to improve image-to-image translation. To clarify, we do not assume an accurate segmentation model, instead some model that performs reasonably well will suffice, which could be obtained via training a semantic segmentation model in a supervised setup using limited data; for example, we use about 1K annotated images in our experiments on the Cityscapes dataset. Our goal is to use this model to improve domain adaptation, so that we can adapt a large number of synthetic images to the target domain, to train a better segmentation model on the target domain.

The use of the segmentation models could help us even better. As alluded to above, the main challenge in standard image translation models is the inability of the network to find the correct mode mapping. We explore this facet of our framework by introducing semantic dropouts by stochastically blanking out semantic classes from the inputs so that the network can learn to map specific classes independently. We present experiments on a variety of image-to-image translation tasks and show that our scheme outperforms those using cycle-GAN significantly.

Before moving on, we summarize below our main contributions:

- We introduce a novel feedback to the generator in GANs using predictions from a semantic segmentation model.

- We propose a GAN architecture that includes a segmentation module and the whole framework trained in an end-to-end manner.

- We introduce semantic dropout for improving our consistency loss.

- We present experiments on several image-to-image translation tasks, demonstrating state-of-the-art results (sometimes by more than 20% in FCN score against Cycle-GAN) Further, we provide experiments using the proposed translation for training semantic segmentation models using large synthetic datasets, and show that our translations lead to significantly better segmentation models than state-of-the-art models (by 4-6% in mean IoU score).

2. Related Work

GANs [16, 44, 36, 1] allow learning complex data distributions automatically by pitting two CNNs (a generator and a discriminator) against each other using an adversarial loss. The optimum for this non-convex min-max game is when the generator produces data which the discriminator cannot distinguish from real data. This key idea has

led to major advancements in applications requiring data synthesis such as: representation learning [36], image generation [51, 52, 36], text-to-image synthesis [37], inpainting [35], super-resolution [23], face-synthesis [28], style-transfer [21, 52, 41], and image editing [56].

The basic GAN [16, 36] framework is extended in Liu and Tuzel [27] to model the joint distribution of paired images from two distinct domains by coupling two GANs by sharing their weights. This scheme is extended in VAE-GAN [26] for image-to-image translation using auto-encoders to embed images to a shared space, on which the generators are conditioned. SimGAN [47] replaces the noise input in traditional GANs by synthetic images, and asking the generator to refine these images to look as real as possible. Similar to ours, SimGAN uses an FCN for translation consistency; however, this FCN is used to preserve the holistic structure of the images and not to preserve the class identities. In Cycle-GAN [57] and dual-GAN [54], the translations are required to be bijective. However, as noted earlier, such bijective mappings need not preserve semantic consistency. Other forms of consistencies have been explored in recent works. In Sangkloy et al. [45], consistency between sketch boundaries is presented, perceptual consistency is enforced in DeepSim [11], while geometric consistency is explored in Xu et al. [53]. Self-similarity is used in Deng et al. [10], and feature-level consistency is assumed in X-GAN [42]. In Luo et al. [30], domain alignment is used.

There are several works concurrent to ours that explore other ways for semantic consistency. In PixelDA [3], DTN [48], and CyCADA [19], the translation task is required to be invariant under a pre-defined criteria – such as a classifier performance. However, their tasks are different from ours and do not assume the availability of target segmenters like we do. Semantic consistency using attention is presented in DA-GAN [31]. In Sankaranarayan et al. [46], a neighborhood-preserving feature embedding is introduced. Similarly Li et al. [24] uses a semantic-aware discriminator to preserve the high-level appearances. In triangle GAN [14], a semi-supervised approach is presented for paired examples. In bidirectional-GAN [43], the source class labels are used for semantic consistency. In one-sided GAN [2], the cycle-loss is replaced by neighborhood constraints. In comparison to these works, we tackle a different problem in which we assume to have access to (approximate) segmentation networks that can extract the source and target class labels.

To summarize, while semantic consistency has been explored from multiple facets in prior works, we are unaware of any work that explores segmentation consistency in the way we propose. We believe ours is the first work that leverages advances in the segmentation arena into the GAN framework for the image-to-image translation problem.

3. Proposed Method

In this section, we present our Sem-GAN framework. To set the stage, we first review in the following sections, important previous work on GAN and Cycle-GAN on which our scheme is based.

3.1. Problem Setup

Let $\mathcal{D}_A, \mathcal{D}_B$ be two image domains and let $\mathcal{X}_A, \mathcal{X}_B$ be sets of samples (images) from each domain respectively. Further, let $x_A \in \mathcal{X}_A$ and $x_B \in \mathcal{X}_B$ denote data samples. We assume $\mathcal{X}_A$ and $\mathcal{X}_B$ are unpaired, however, the two domains share K semantic segment classes (with plausibly varied appearances). Assuming $\mathcal{S}$ is the space of all segmentation masks with K classes, let $\mathcal{S}_A : \mathcal{D}_A \to \mathcal{S}$ and $\mathcal{S}_B : \mathcal{D}_B \to \mathcal{S}$ be two functions mapping each pixel in each input image to their respective class label in the segmentation mask. In case, if we have access to the ground-truth masks, then we use g_A and g_B to denote these masks for inputs x_A and x_B respectively. Ideally, $g_A = \mathcal{S}_A(x_A)$ and $g_B = \mathcal{S}_B(x_B)$. In this case, (x_A, g_A) and (x_B, g_B) form an image-ground-truth pair from each domain (however note that the pairs (x_A, g_A) and (x_B, g_B) remain unpaired).

3.2. Generative Adversarial Networks

A standard GAN [16] consists of two convolutional neural networks (CNN), termed a *generator* and a *discriminator*; the former takes random noise as input to produce an image, while the latter identifies if its input is a true or a generated image. The parameters of the generator and discriminator CNNs are optimized against an adversarial loss in a min-max game [1, 16, 36].

Extending this idea to an image-to-image translation setting, we define two generators G_{AB} and G_{BA} defined as $G_{AB} : \mathcal{X}_A \to \mathcal{D}_B$ and $G_{BA} : \mathcal{X}_B \to \mathcal{D}_A$ and two binary discriminators D_A and D_B, where $D_A(x) = 1, \forall x \in \mathcal{X}_A$ and $D_A(x) = 0, \forall x \in G_{BA}(\mathcal{X}_B)$. Similarly, $D_B(x) = 1, \forall x \in \mathcal{X}_B$ and $D_B(x) = 0, \forall x \in G_{AB}(\mathcal{X}_A)$. Here, with a slight abuse of notation, we assume $G(\mathcal{X})$ is the set of fake images produced by a generator G from domain $\mathcal{X}$. To learn the parameters of the generators and the discriminators, we define the following adversarial losses using binary cross-entropy:

$$\min_{G_{AB}} \max_{D_B} \quad \ell_{GAN}^{AB} := \mathbb{E}_{x_B \sim \mathcal{D}_B} \log(D_B(x_B)) +$$

$$\mathbb{E}_{x_A \sim \mathcal{D}_A} \log(1 - D_B(G_{AB}(x_A))), \quad (1)$$

$$\min_{G_{BA}} \max_{D_A} \quad \ell_{GAN}^{BA} := \mathbb{E}_{x_A \sim \mathcal{D}_A} \log(D_A(x_A)) +$$

$$\mathbb{E}_{x_B \sim \mathcal{D}_B} \log(1 - D_A(G_{BA}(x_B))). \quad (2)$$

The GAN architecture for these objectives is graphically illustrated in Figure 2(a). While, (1) and (2) represent a

non-convex game whose optimum parameters correspond to saddle points (and thus typically difficult to optimize) it is often seen that with suitable heuristics [16, 36, 44] and careful choices for the loss [1, 32], the problem converges to practically useful solutions.
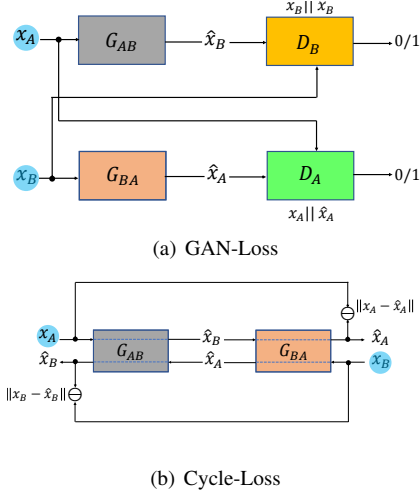

(a) GAN-Loss

(b) Cycle-Loss

Figure 2. Illustrations of the objectives. The inputs are highlighted in 'blue' circles.

3.3. Cycle-Consistent GAN

An important problem that one often encounters when training GANs is that of mode collapse which happens when the generators G learn to produce a few samples from the true data domain without completely spanning it. Given that the discriminator loss does not enforce diversity in its inputs; the optimization may converge to such local solutions. Among several workarounds proposed to tackle this problem [1, 36, 44], one that has been very promising in the image-translation setting is that of Cycle-GAN [57] that puts additional constraints to the GAN objective enforcing diversity implicitly. Specifically, the Cycle-GAN loss asks for the translated data to be re-translated back to their original inputs. Mathematically, this loss can be written as:

$$\ell_{cycle} := \mathbb{E}_{x_A \sim \mathcal{D}_A} \|x_A - G_{BA}(G_{AB}(x_A))\| + \\ \mathbb{E}_{x_B \sim \mathcal{D}_B} \|x_B - G_{AB}(G_{BA}(x_B))\|, \quad (3)$$

where $\| . \|$ is a suitable norm. Optimizing this requirement within a GAN formulation (1, 2) automatically demands that the generators G_{AB} and G_{BA} learn unique mappings such that they may be invertible to the original inputs; thereby elegantly avoiding the collapse of the data modes. The cyclic-constraints are depicted in Figure 2(b).

While, the cyclic constraint has resulted in several compelling image-to-image translation results on domains that are often uni-modal (such as horse-zebra, day-night etc.), it is theoretically unclear how the scheme may perform in a multi-modal setup. This is because, when there are multiple modes in either domains, any bijective mapping between the modes will satisfy the invertibility constraint in (3), such as in the example depicted in Figure 1. Recently, this assignment problem has been looked at in the work of Gananti et al. [49], hypothesizing that by constraining the space of possible translation functions (by controlling the capacity of the CNNs) may be allowing them to learn minimal-complexity mappings. While, this result is interesting, it may be practically difficult to achieve. Instead, we seek to constrain the possible mappings by guiding the generators to learn mappings that are semantically consistent with respect to a segmentation loss, viz. Sem-GAN.

3.4. Semantically-Consistent GAN

In this section, we present our Sem-GAN framework. Re-using notation from Section 3.1, we assume to have segmentation functions S_A and S_B that are trained to segment images from their respective domains. We do not assume that these segmenters are trained on $\mathcal{X}_A$ and $\mathcal{X}_B$ necessarily, but could be trained on external datasets that are similar to our domains. Using these segmenters, our semantic consistency constraint is written as:

$$\ell_{seg} := \mathbb{E}_{x_A \in \mathcal{D}_A} \mathcal{L}(S_A(x_A), S_B(G_{AB}(x_A))) + \\ \mathbb{E}_{x_B \in \mathcal{D}_B} \mathcal{L}(S_B(x_B), S_A(G_{BA}(x_B))), \quad (4)$$

where $\mathcal{L}(P, Q)$ is a suitable loss comparing two segmentation masks $P, Q \in \mathcal{S}$, e.g., the cross-entropy loss used in FCN [29]. Specifically, in (4), we enforce that the segmentation of x_A by a function S_A trained using images from domain $\mathcal{D}_A$ should be preserved by a segmentation function S_B trained on images from $\mathcal{D}_B$ when applied to the translated image. When ground truth semantic labels g_A and g_B are available for either domains, we replace $S_A(x_A)$ by g_A and $S_B(x_B)$ by g_B in (4). In this case, Eq. (4) can be written as:

$$\ell^g_{seg} := \mathbb{E}_{(x_A, g_A) \in \mathcal{D}_A \times \mathcal{S}} \quad \mathcal{L}(g_A, S_B(G_{AB}(x_A))) \\ + \mathbb{E}_{(x_B, g_B) \in \mathcal{D}_B \times \mathcal{S}} \quad \mathcal{L}(g_B, S_A(G_{BA}(x_B))). \quad (5)$$

Figure 3(a) and 3(b) illustrate these two variants of our Seg-GAN losses. For convenience in the depiction, we have introduced additional variables $\hat{g}$ to denote the outputs of the segmentation modules.

3.5. Overall Architecture

In Figure 4, we present our complete Sem-GAN architecture, combining the three losses. Notably, we include the two segmentation modules, S_A, S_B, as part of the setup; which are trained end-to-end alongside the generators and the discriminators. Using hyperparameters λ_1 and λ_2, our full loss is given by:

$$\ell := \ell^{AB}_{GAN} + \ell^{BA}_{GAN} + \lambda_1 \ell_{cycle} + \lambda_2 \ell_{seg}. \quad (6)$$

(a) Seg-Loss w/o ground truth
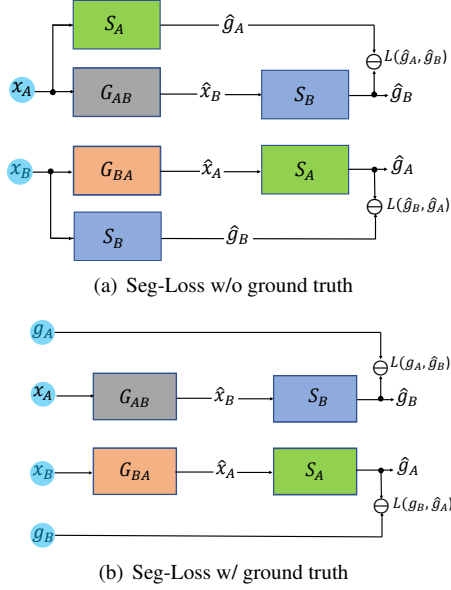

(b) Seg-Loss w/ ground truth

Figure 3. Two variants of our segmentation consistency model. Left: when the ground truth annotations are not used. Right: when they are used.
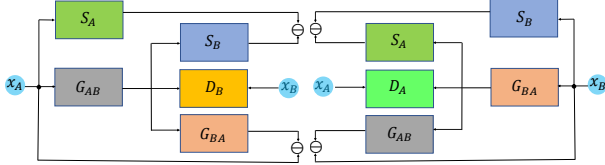

Figure 4. Overall architecture. Note that some modules are repeated on the left and the right parts of our illustration to avoid cluttered cross-connections. Thus, blocks with the same color represent the same module.

With such an end-to-end architecture, that learns the segmenters and the discriminators together, there is a subtle but important risk. Note that, since G_{AB} and G_{BA} are learned alongside S_A and S_B, it is often observed empirically that the segmenters will learn unrealistic appearances as valid ground truth classes when the generators are still in the learning phase. For example, suppose say in early epochs, the generator has not started translating valid 'car' images. Instead, it translates the appearance of a 'person' to a 'car' class (this is possible as we do not have paired data). Now, when back-propagating the error to update the parameters of the segmenter using the ground truth mask for the translated image, the segmenter may incorrectly learn to map the 'person' appearance to 'car' class. This crucial issue may fail the semantic consistency. To circumvent this problem, we propose to optimize the segmenters using their ground-truth labels alongside updating the discriminator parameters; i.e., we do not use the generator outputs to train the segmenters until they are accurate.

3.6. Semantic Dropout

The availability of segmenters S_A and S_B allow for applying some modifications to our inputs such that the translation networks can be trained more effectively. Specifically, we could arbitrarily mask out object classes from both the input images. As a result, a generator could learn to map corresponding classes individually (mode-to-mode translation rather than translating the joint distribution of all labels together). Precisely, let M^k denote a segment mask for class $k \in \{1, 2, \cdots, K\}$, which has all zeros for all classes, except class k (which is unity). To make the GAN learn class-to-class translation, we propose to transform input images x_A and x_B to $x'_A = M^k_A \odot x_A$ and $x'_B = M^k_B \odot x_B$, where $\odot$ is the element-wise product. Next, we use these x''s in the above losses.

A problem with this scheme is that, while the network learns to transform individual classes using semantic dropout, it may miss learning the inter-class context within images. To this end, we propose to use the dropout stochastically, i.e., with a probability p, we select a label k randomly from classes common to a pair of tuples (x_A, g_A) and (x_B, g_B). Next, using the respective ground truth masks, we create new semantic masks M^A_k and M^B_k, which are then used to select the respective image pixels to generate x'_A and x'_B as described above. The full dropout pipeline is provided in Algorithm 1.

Input: tuples (x_A, g_A) and (x_B, g_B); dropout p
$l_A \leftarrow get\_labels(g_A)$; // get labels in g_A
$l_B \leftarrow get\_labels(g_B)$;
$l_{AB} \leftarrow l_A \cap l_B$; // find common labels
if $uniform(0, 1) <= p$ **then**
$\quad k \sim$ multinomial$(|l_{AB}|)$; // $|\ .\ |$ is the
$\quad$ set cardinality.
$\quad M^k_A \leftarrow get\_mask(l_{AB}[k], g_A)$; // generate
$\quad$ a mask on g_A with $l_{AB}[k]$
$\quad M^k_B \rightarrow get\_mask(l_{AB}[k], g_B)$;
$\quad x'_A \rightarrow M^k_A \odot x_A$;
$\quad x'_B \rightarrow M^k_B \odot x_B$;
$\quad$ **return** $(x'_A, M^A_k \odot g_A), (x'_B, M^B_k \odot g_B)$;
$\quad$ // assuming g's are 1-indexed.
else
$\quad$ **return** $(x_A, g_A), (x_B, g_B)$
end

Algorithm 1: Semantic dropout.

4. Experiments

We use three datasets and six image translation tasks to demonstrate the improvements afforded by Sem-GAN. Details of these datasets, tasks, network architectures, and our evaluation protocols follow. We also report results on improving the semantic segmentation accuracy.

4.1. Datasets

Cityscapes (CS) [7]: consists of 5K densely annotated real-world road scene images collected from 50 European cities and annotated for 30 semantic classes. The dataset has moderate diversity in weather and lighting conditions.

Mitsubishi Precision (MP): consists of about 20K road scene images generated by the Mitsubishi Precision Co. simulator and densely annotated for 36 semantic classes. The dataset has high-resolution images from varied weather (summer, winter, rain), lighting conditions (dawn, dusk, night), and object appearances.

VIPER [38]: (which is a recent version of the popular GTA5 dataset [39]) consists of 250K frames from driving videos in realistic virtual worlds generated by the Unity gaming engine. The dataset is densely annotated for 31 semantic classes and includes images from varied weather and lighting conditions.

4.2. Data Preparation

As the images in our datasets are of different resolutions, we resize them to a common size of 540×860 pixels. Further, since the synthetic images are from video sequences, nearby frames might be very similar. To this end, we uniformly sample 5K frames from each of the synthetic datasets. We map the semantic classes from all the datasets to a common subset, with the Cityscapes annotations as the reference. We find that 19 classes are common, and use only these to enforce semantic consistency. Details are available in the supplementary material. We report experiments on five bi-directional translation tasks, namely (i) CS↔MP, (ii) CS↔Viper, (iii) CS Summer↔ MP Winter, (iv) CS Day ↔ MP night, and (v) MP Summer↔ MP Winter. We also present experiments on the task of mapping segmentation masks to real images (Seg → CS) to show that conditioning the generators directly on the segment labels may not be a replacement to our scheme. In this case, we use unpaired translations, that is we have sets of masks and images, without correspondences. Thus, our setting is different from that of pix2pix [20]. Note that, even though we use limited ground truth segmentation masks on the Cityscapes dataset, our problem remains unpaired as we do not assume correspondences between such image-label pairs across domains.

4.3. Network Architectures

We implemented Sem-GAN from the code shared as part of Cycle-GAN[2] using PyTorch [33]. For the generators and discriminators in GAN, we use a sequence of 9 residual network blocks. The Adam optimizer [22] is used for training

the networks, with an initial learning rate of 0.0002. The validation accuracy seem to saturate in about 50 epochs for all our tasks, except for the Seg→CS task, for which we use 200 epochs. For our segmentation networks, we use the FCN [29] implementation in Pytorch.[3], which is cheaper and faster to train alongside other modules in our framework in comparison to other deeper networks such as Deeplab [5] and PSP-Net [55]. In FCN, we use a VGG-16 network and use cross-entropy loss on the final output layer for enforcing the Sem-GAN criteria.

4.4. Training, Testing, and Evaluation

We define training, validation, and test sets by random sampling each dataset in 85:5:10 splits. The images are cropped to 256×256 pixels; the training inputs are cropped randomly during training (as part of data augmentation), while the validation and test images are center-cropped. The segmentation networks are pre-trained on the respective training sets to recognize 19 semantic classes. Note that, we use only images from ideal weather conditions (well-lit and good weather) for this training, while networks for other conditions (day, night, winter, etc.) are learned jointly with other modules in Sem-GAN. For training the segmenters, we fine-tune a VGG-16 model with batches of 16 images, optimizing the parameters using stochastic gradient descent with a learning rate of 0.0001 and a momentum of 0.9. During testing, we do not use the segmentation pipeline, instead directly forward pass the source images through the generators and gather the translated images for evaluation.

For quantitative evaluations, we use the semantic segmentation accuracy of the translated images by a segmentation model trained on the respective domain. To ensure unbiased evaluation, we report results using two segmentation networks, namely (i) FCN and (ii) PSP-Net [55]. The evaluation networks are trained separately from Sem-GAN on training sets from the respective domains. Using these models, we report results on (i) the overall accuracy (Over. Acc) – which is the number of pixels correctly predicted against the total number of annotated pixels, (ii) the average class accuracy (Avg. Acc) – which is the average of per-class accuracy, and (iii) the mean intersection-over-union (mIoU) score [29] over all the classes. On the 19 evaluation classes, FCN achieves mIoU of 64.1%, 56.2%, and 51.7% on the test sets of MP, Viper, and CS datasets respectively, while PSP-net gets 73.4%, 71.1% and 61.1%. We use 1K images randomly sampled from the Cityscapes dataset for training the respective segmentation models on this dataset.

4.5. Semantic Dropout

We first analyze the merit of semantic-dropout. This scheme has a parameter p, which is the probability of dropping segments; a higher-value of p drops segments too fre-

---

A↔B Task	Scheme	A→B Avg. Acc	A→B Over. Acc	A→B mIoU	B→A Avg. Acc	B→A Over. Acc	B→A mIoU
MP↔CS	VAE-GAN	42.6	59.4	13.2	13.2	30.9	6.1
	Style-Trans.	44.5	72.1	26.3	15.6	27.8	6.3
	Cycle-GAN	36.7	56.2	16.2	19.5	36.9	7.2
	Sem-GAN	51.5	71.9	34.1	29.1	58.4	18.3
	Sem-GAN+SM	60.7	80.2	**40.2**	29.4	67.8	**19.3**
Viper↔CS	VAE-GAN	31.3	54.5	13.4	18.9	36.4	8.6
	Style Trans.	29.3	64.7	13.7	17.9	61.6	11.0
	Cycle-GAN	23.6	54.1	9.2	21.0	63.9	13.4
	Sem-GAN	38.8	82.0	24.0	27.4	80.3	20.4
	Sem-GAN+SM	42.5	84.2	**28.4**	27.7	81.6	**21.5**
MP(N)↔CS(D)	VAE-GAN	22.6	39.6	7.4	10.7	20.0	5.5
	Cycle-GAN	32.8	47.5	10.7	14.8	48.2	7.1
	Sem-GAN	54.1	79.7	32.7	27.6	78.3	**20.5**
	Sem-GAN+SM	56.2	80.0	**36.7**	28.6	78. 1	20.2
MP(W)↔CS(S)	VAE-GAN	26.2	48.5	8.0	9.08	41.1	4.7
	Cycle-GAN	27.1	65.9	13.2	12.8	51.9	7.1
	Sem-GAN	51.3	85.9	32.3	22.4	76.4	16.2
	Sem-GAN+SM	50.1	84.2	**34.2**	22.5	72.1	**16.9**
MP(S)↔MP(W)	VAE-GAN	53.0	87.1	41.8	57.6	75.9	45.2
	Cycle-GAN	60.0	74.6	47.4	61.8	91.0	51.0
	Sem-GAN	53.1	75.9	43.4	62.9	92.2	52.3
	Sem-GAN+SM	54.7	75.1	**45.5**	63.2	92.3	**53.3**
Seg→CS	VAE-GAN	7.36	48.0	4.0	NA	NA	NA
	Cycle-GAN	12.6	37.7	7.8	NA	NA	NA
	Sem-GAN	35.6	75.0	**26.6**	NA	NA	NA

Table 1. Results using our Sem-GAN and semantic dropout (SM) against the state-of-the-art Cycle-GAN [57], VAE-GAN [26], and style transfer model [21]. We use the FCN [29] for evaluation. All numbers are in %. W=Winter, S=Summer, D=Day, and N=Night.

A↔B Task	Scheme	A→B Avg. Acc	A→B Over. Acc	A→B mIoU	B→A Avg. Acc	B→A Over. Acc	B→A mIoU
MP↔CS	Cycle-GAN	45.6	62.6	21.5	23.7	55.9	11.6
	Sem-GAN	50.0	77.4	**34.2**	30.1	71.2	**18.6**
Viper↔CS	Cycle-GAN	29.8	57.3	15.4	24.6	70.7	17.3
	Sem-GAN	38.3	74.0	**23.2**	30.9	80.8	**23.5**
MP(N)↔CS(D)	Cycle-GAN	34.1	48.2	15.5	19.0	57.2	8.97
	Sem-GAN	49.8	74.0	**27.9**	29.3	77.5	**20.2**
MP(W)↔CS(S)	Cycle-GAN	33.9	63.3	14.8	13.4	52.1	7.39
	Sem-GAN	48.3	76.8	**26.2**	23.5	75.8	**16.9**
MP(S)↔MP(W)	Sem-GAN	64.6	76.7	51.1	65.6	91.6	**54.8**
	Sem-GAN	57.9	76.7	**57.2**	63.2	90.7	51.9
Seg→CS	Cycle-GAN	16.2	46.0	9.92	NA	NA	NA
	Sem-GAN	19.7	55.0	**13.4**	NA	NA	NA

Table 2. Comparisons between Sem-GAN and Cycle-GAN [57] using PSP-net [55] segmentation model for evaluation. W=Winter, S=Summer, D=Day, and N=Night.

quently; as a result, the generators may not be able to learn their spatial contexts. To this end, in Figures 5(a) and 5(b), we plot the mIoU for the MP↔CS tasks against varying $p = \{0, 0.2, 0.4, 0.6, 0.8\}$. As is clear from the plots, semantic dropout improves the performance of translation significantly; e.g., on the CS→MP task, the gap between $p = 0$ and $p = 0.2$ is ∼20%. We also see that a higher value $p = 0.8$ shows lower accuracy. Thus, we use $p = 0.2$ for CS→MP and $p = 0.4$ for MP→CS and Viper→CS.

4.6. State-of-the-Art Comparisons

In Tables 1 and 2, we compare Sem-GAN against three state-of-the-art image translators: (i) Cycle-GAN [57], (ii) VAE-GAN [26], and style-transfer using perceptual losses [21]. We also report performances with and without semantic dropout (SM). As is clear, Sem-GAN (+ SM) outperforms Cycle-GAN in almost all tasks, especially on the challenging mIoU criteria. Specifically, we find that on MP→CS and Viper→CS tasks, our scheme is nearly 20% better in classification accuracies. Similar results are observed on other tasks as well, except in

MP(Snow)→MP(Winter) translations. In this case, the source and target domains are inherently the same, except for simulated snow in the latter, which can be undone by the generator, thereby perfectly aligning the domains. In Figures 5(d) and 5(c), we analyze the per-class IoU for the MP↔CS task. Note that not all classes are present in our (randomly chosen) test set. We see that Sem-GAN almost always shows superior translations on most classes. In Figure 6, qualitative results are presented. On the Mask→CS task, Sem-GAN guides the error from the segmenters to be improve the appearance of the generated segments as demonstrated by results in Tables 1 and 2, leading to better results than the other models. More qualitative and quantitative results using Sem-GAN are provided in [6].

4.7. Improvements on Semantic Segmentation

Next, we analyze the merit of Sem-GAN for improving the original task, namely training semantic segmentation models via synthetic data. Our analysis is loosely based on [19], however using our datasets and evaluation models. We use 10K images from the two synthetic datasets and 200 images from the Cityscapes (CS) dataset. We translated the synthetic images (source) to the CS domain (as in [19]) and used source ground truth labels for training two segmentation models. We used a test set of 500 CS images for evaluating our models. All the algorithms are trained using SGD with a learning rate of 0.0001 for 50 epochs. As is clear from Table 3, Cycle GAN is sometimes seen to reduce the performance (e.g., Cy(VP)) against no adaptation likely due to the correspondence mismatch problems alluded to earlier. However, Sem-GAN improves image adaptation significantly compared to CycleGAN, and leads to more accurate segmentation models than when not using adaptation; e.g., "CS only" with FCN8s is 19.9% mIoU, while using Sem-GAN i.e., CS+Sem(VP), this improves to 34.4%. Similarly, using PSPNet, "CS only" to CS+Sem(VP) is 24.4% to 44.4%, a 20% improvement. Further, note that the improvement from CS+VP to CS+Sem(VP) is nearly 6%; the former is without any adaptation on VP images.

5. Conclusions

We presented an image-to-image translation framework that uses semantic consistency using segment class identities for achieving realistic translations. Modeling such a consistency as a novel loss, we presented an end-to-end learnable GAN architecture. We demonstrated the advantages of our framework on three datasets and six translation tasks. Our results clearly demonstrate that semantic consistency, as is proposed in this paper, is very important for ensuring the quality of the translation.

Method	n/w	Road	s.walk	Bldg	wall	fence	pole	t. light	t. sign	veg.	terrain	sky	person	rider	car	truck	bus	train	m.cycle	bicycle	**mIoU**	Ov. Acc
CS only	**FCN**	85.1	38.9	60.6	0.8	1.1	0.0	0.0	0.0	65.1	7.4	36.3	19.8	0.0	62.7	0.0	0.0	0.0	0.0	0.4	19.9	77.4
CS + MP	FCN	87.0	41.8	64.6	14.5	0.2	0.6	0.1	0.4	68.6	11.7	67.3	11.2	0.0	63.6	0.0	0.0	0.0	0.0	0.1	22.7	79.3
CS+Cy(MP)	FCN	85.5	40.3	63.6	6.9	0.0	3.2	0.4	3.5	69.0	7.8	52.2	11.7	0.0	62.5	0.0	0.0	0.0	0.0	2.6	21.5	77.8
CS+Sm(MP)	FCN	88.1	47.0	67.8	12.8	0.5	7.1	0.0	2.0	71.1	10.0	69.0	15.4	0.0	67.6	0.0	0.0	0.0	0.0	4.0	**24.3**	80.8
CS + VP	FCN	90.2	54.1	70.3	22.6	5.9	3.4	0.0	3.4	73.4	27.2	67.1	31.4	0.3	73.2	9.1	18.7	5.6	0.2	21.2	30.4	83.2
CS+Cy(VP)	FCN	90.3	53.2	67.4	9.4	15.3	2.6	0.1	4.8	70.6	26.4	60.9	36.4	0.9	74.2	21.1	24.9	3.4	5.0	30.4	31.4	82.8
CS+Sm(VP)	FCN	92.1	59.1	71.3	21.6	19.1	4.4	0.2	5.6	74.1	30.2	70.1	36.4	1.3	76.8	24.2	20.5	11.7	4.3	30.7	**34.4**	84.6
CS only	**PSP**	85.2	35.2	62.9	4.1	**15.4**	0.5	0.0	2.6	68.6	24.3	49.0	27.2	0.0	63.6	0.0	4.8	0.0	0.0	19.7	24.4	78.7
CS+MP	PSP	88.8	49.5	70.2	7.0	4.7	9.0	0.0	13.4	72.8	20.2	74.9	38.2	0.0	73.5	0.0	3.2	0.0	0.0	31.5	29.3	82.4
CS+Sm(MP)	PSP	90.3	54.2	72.4	17.4	8.0	16.6	0.1	17.9	75.8	23.6	74.2	42.8	8.5	74.3	0.0	17.3	0.0	0.0	36.1	**33.1**	84.3
CS+VP	PSP	91.5	54.2	74.8	23.8	7.4	18.3	3.0	13.7	76.9	24.8	66.5	48.6	22.2	82.1	35.7	19.7	28.2	6.4	42.7	39.0	85.6
CS+Sm(VP)	PSP	**93.4**	**63.4**	**76.4**	**27.3**	11.7	**23.6**	**15.8**	**23.6**	**78.2**	**32.0**	**78.4**	**52.2**	**26.2**	**84.0**	**33.4**	**33.4**	**30.1**	**18.1**	42.4	**44.4**	**87.4**

Table 3. Training segmentation models using adapted images. We adapt synthetic MP and VIPER (VP) datasets to Cityscapes (CS) domain. Sm(MP) and Cy(MP) denote adaptation of all images in MP with our Sem-GAN and Cycle GAN respectively to the CS domain. "only" refers to using images directly from that domain (without adaptation). We use two segmentation CNNs: "FCN" is VGG-FCN8s and "PSP" is PSPNet for segmentation evaluation. We also show per-class accuracy on the 19 common classes across the three datasets.
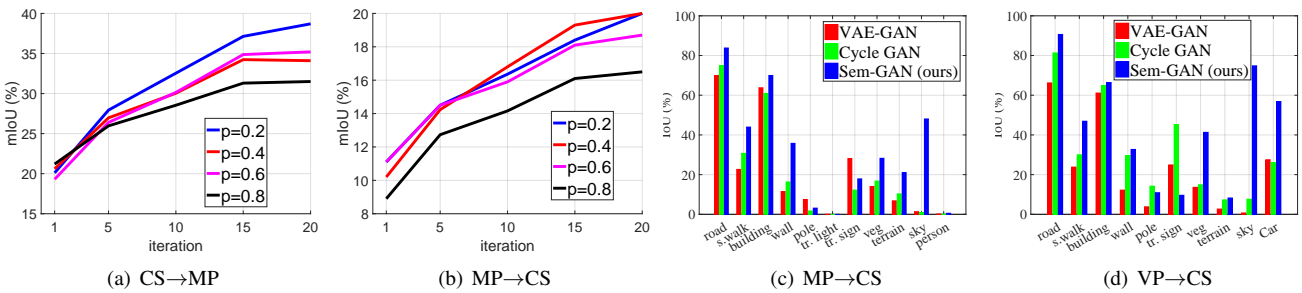

Figure 5. Effect of semantic dropout on image translation accuracy. Right: Detailed analysis of accuracies when translating each class.
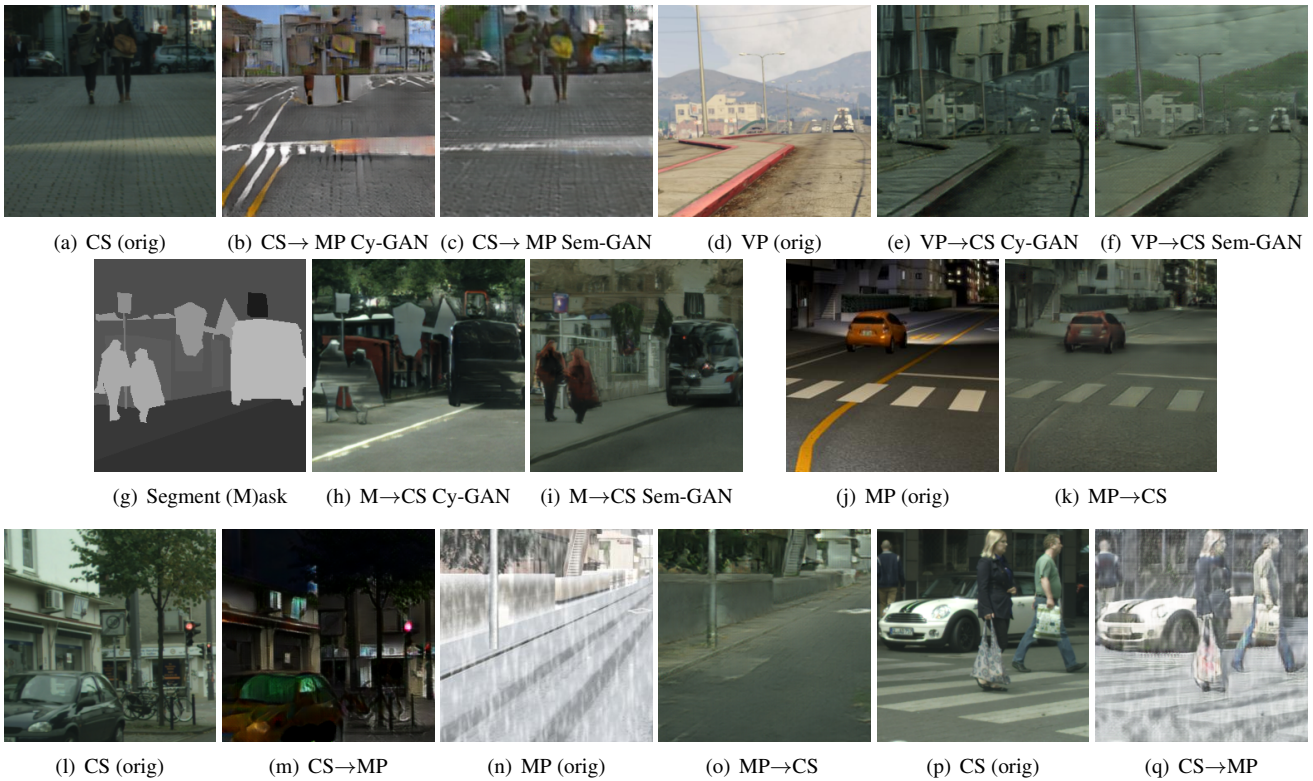
(a) CS→MP (b) MP→CS (c) MP→CS (d) VP→CS

(a) CS (orig) (b) CS→ MP Cy-GAN (c) CS→ MP Sem-GAN (d) VP (orig) (e) VP→CS Cy-GAN (f) VP→CS Sem-GAN

(g) Segment (M)ask (h) M→CS Cy-GAN (i) M→CS Sem-GAN (j) MP (orig) (k) MP→CS

(l) CS (orig) (m) CS→MP (n) MP (orig) (o) MP→CS (p) CS (orig) (q) CS→MP

Figure 6. Qualitative results; (a–c) CS→ MP, (d–f) Viper→CS, (g–i) segment mask→ CS, (j–m) Night↔Day, and (n–q) Winter↔Summer. More results in the supplementary material.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[2] S. Benaim and L. Wolf. One-sided unsupervised domain mapping. In *NIPS*, 2017.

[3] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *CVPR*, 2017.

[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *arXiv preprint arXiv:1606.00915*, 2016.

[6] A. Cherian and A. Sullivan. Sem-gan: Semantically-consistent image-to-image translation. *arXiv preprint arXiv:1807.04409*, 2018.

[7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016.

[8] C. R. de Souza, A. Gaidon, Y. Cabon, and A. L. Pena. Procedural generation of videos to train deep action recognition networks. In *CVPR*, 2017.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[10] W. Deng, L. Zheng, G. Kang, Y. Yang, Q. Ye, and J. Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. *arXiv preprint arXiv:1711.07027*, 2017.

[11] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. In *NIPS*, 2016.

[12] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015.

[13] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*, 2013.

[14] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin. Triangle generative adversarial networks. In *NIPS*, 2017.

[15] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[18] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[19] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017.

[20] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017.

[21] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016.

[24] P. Li, X. Liang, D. Jia, and E. P. Xing. Semantic-aware grad-gan for virtual-to-real urban scene adaption. *arXiv preprint arXiv:1801.01726*, 2018.

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

[26] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017.

[27] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *NIPS*, 2016.

[28] S. Liu, Y. Sun, D. Zhu, R. Bao, W. Wang, X. Shu, and S. Yan. Face aging with contextual generative adversarial nets. In *ACM on Multimedia Conference*, 2017.

[29] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[30] Z. Luo, Y. Zou, J. Hoffman, and L. F. Fei-Fei. Label efficient learning of transferable representations acrosss domains and tasks. In *NIPS*, pages 164–176, 2017.

[31] S. Ma, J. Fu, C. W. Chen, and T. Mei. Da-gan: Instance-level image translation by deep attention generative adversarial networks. *arXiv preprint arXiv:1802.06454*, 2018.

[32] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *ICCV*, 2017.

[33] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

[34] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3):53–69, 2015.

[35] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[36] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[37] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. *arXiv preprint arXiv:1605.05396*, 2016.

[38] S. R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In *ICCV*, 2017.

[39] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016.

[40] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016.

[41] R. Rosales, K. Achan, and B. J. Frey. Unsupervised image translation. In *ICCV*, 2003.

[42] A. Royer, K. Bousmalis, S. Gouws, F. Bertsch, I. Moressi, F. Cole, and K. Murphy. Xgan: Unsupervised image-to-image translation for many-to-many mappings. *arXiv preprint arXiv:1711.05139*, 2017.

[43] P. Russo, F. M. Carlucci, T. Tommasi, and B. Caputo. From source to target and back: symmetric bi-directional adaptive gan. *arXiv preprint arXiv:1705.08824*, 2017.

[44] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *NIPS*, 2016.

[45] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *CVPR*, 2017.

[46] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa. Unsupervised domain adaptation for semantic segmentation with gans. *arXiv preprint arXiv:1711.06969*, 2017.

[47] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017.

[48] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. *arXiv preprint arXiv:1611.02200*, 2016.

[49] T. G. TAU, L. Wolf, and S. B. TAU. The role of minimal complexity functions in unsupervised learning of semantic mappings. *arXiv preprint arXiv:1709.00074*, 2017.

[50] M. Wang and W. Deng. Deep visual domain adaptation: A survey. *arXiv preprint arXiv:1802.03601*, 2018.

[51] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. *arXiv preprint arXiv:1711.11585*, 2017.

[52] X. Wang and A. Gupta. Generative image modeling using style and structure adversarial networks. In *ECCV*, 2016.

[53] W. Xu, Y. Li, and C. Lu. Generating instance segmentation annotation by geometry-guided gan. *arXiv preprint arXiv:1801.08839*, 2018.

[54] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *CVPR*, 2017.

[55] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[56] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *ECCV*, 2016.

[57] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.