GAN-Based Day-to-Night Image Style Transfer for Nighttime Vehicle Detection

Che-Tsung Lin, Sheng-Wei Huang, Yen-Yi Wu, and Shang-Hong Lai, *Member, IEEE*

***Abstract*— Data augmentation plays a crucial role in training a CNN-based detector. Most previous approaches were based on using a combination of general image-processing operations and could only produce limited plausible image variations. Recently, GAN (Generative Adversarial Network) based methods have shown compelling visual results. However, they are prone to fail at preserving image-objects and maintaining translation consistency when faced with large and complex domain shifts, such as day-to-night. In this paper, we propose AugGAN, a GAN-based data augmenter which could transform on-road driving images to a desired domain while image-objects would be well-preserved. The contribution of this work is three-fold: (1) we design a structure-aware unpaired image-to-image translation network which learns the latent data transformation across different domains while artifacts in the transformed images are greatly reduced; (2) we quantitatively prove that the domain adaptation capability of a vehicle detector is not limited by its training data; (3) our object-preserving network provides significant performance gain in the difficult day-to-night case in terms of vehicle detection. AugGAN could generate more visually plausible images compared to competing methods on different on-road image translation tasks across domains. In addition, we quantitatively evaluate different methods by training Faster R-CNN and YOLO with datasets generated from the transformed results and demonstrate significant improvement on the object detection accuracies by using the proposed AugGAN model.**

***Index Terms*— Vehicle detection, generative adversarial network, image-to-image translation, semantic segmentation, domain adaptation.**

I. Introduction

THE major cause of traffic accidents is mainly due to improper following distance and distracted driving. The most critical function in the advanced driver assistance systems (ADAS) and autonomous vehicles is vehicle detection. One expects that vehicles around host driver could be detected as accurately as possible by an ADAS all day, including day and night. However, vehicle's appearance at daytime is quite different from its counterpart at nighttime. Even in the era of deep learning, a daytime vehicle detector could not function well at nighttime if only standard monocular camera is adopted.

A number of vehicle detection methods were proposed under a Hypothesis Generation (HG) + Hypothesis Verification (HV) framework [1], [2]. The former is to generate region proposals and the latter applies a pair of feature extractor and classifier to eliminate false positives. Detecting vehicles partially occluded or seen by arbitrary viewing angle is a great challenge because its appearance varies to a great extent. Early works in this area were done by detecting a combination of independent vehicle parts. The Deformable Part Model (DPM) [3], using HOG features and latent SVM, can successfully handle deformable object detection even when the target is partially occluded and it has been used for daytime on-road vehicle detection in [4]–[7].

The vast majority of vision-based vehicle detection works were designed to detect vehicles at daytime. However, detecting vehicle at nighttime is difficult in that vehicle's body could not be fully seen consistently even without being occluded. When a vehicle drives in a scenario that street/highway lights are absent, whether or not the body of an un-occluded vehicle could be entirely seen is related to several issues, such as what color its body is, how far it is and most importantly, if its body is lit up by the head-light of the host vehicle or other vehicles. As some features of vehicles, such as headlights and taillights, are more visible at nighttime, some studies [8], [9] proposed nighttime DPM models which are specifically-optimized for nighttime scenario.

Recent advances of object detection are driven by the success of two-stage detectors popularized by region-based convolutional neural network (R-CNN) [10]. The YOLO detector [11] made a break-through by realizing an end-to-end one-stage framework for object detection. Without specific preset rules, a CNN-based detector could easily detect a vehicle seen at arbitrary viewing angles. Indeed, the generalization capability of CNN-based detectors is way better than traditional machine learning approaches. However, performance still drops significantly when the detector is presented with data from a new deployment domain where the model did not see during training. In ADAS or autonomous vehicle, one of the most complex domain shift is between day and night because the object appearances such as vehicles at daytime are very different from their counterparts at nighttime. As indicated by [12] in pedestrian detection, training on daytime and testing on night-time gives significantly worse results than training and testing on the same time-of-day.

Manuscript received February 12, 2019; revised September 14, 2019; accepted December 16, 2019. Date of publication January 6, 2020; date of current version February 2, 2021. This work was supported in part by the Ministry of Science and Technology (MOST), Taiwan, R.O.C., under Grant MOST 108-2634-F-007-002. The Associate Editor for this article was S. S. Nedevschi. *(Corresponding author: Shang-Hong Lai.)*

Che-Tsung Lin is with the Mechanical and Mechatronics System Research Laboratory, Intelligent Vehicle Division, Safety Sensing and Control Department,Industrial Technology Research Institute, Hsinchu 31057, Taiwan, and also with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: alexofntu@gmail.com).

Sheng-Wei Huang and Yen-Yi Wu are with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: mlm4590027@gmail.com; jessicayywu@yahoo.com.tw).

Shang-Hong Lai is with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan, and also with the Microsoft AI R&D Center, Microsoft Corp., Taipei 11065, Taiwan (e-mail: lai@cs.nthu.edu.tw).

Digital Object Identifier 10.1109/TITS.2019.2961679

However, most datasets containing vehicles are captured at daytime. Most importantly, nighttime vehicle datasets in real-driving scenario are scarce in public domain.

A naive thought to overcome this problem is to apply traditional data augmentation techniques, which are usually related to image-processing operations, to enhance the generalization capability of an object detector when deployed in a new domain different from the source one. For example, one may expect that applying those techniques on the daytime training data would help train a vehicle detector to function well at nighttime. However, such kind of transformation could only provide limited plausible data variations. Nowadays, an end-to-end deep learning solution is proven effective qualitatively and quantitatively in many kinds of image translation tasks.

Recently, generative adversarial networks (GANs) [13], which consist of two networks (i.e., a generator and a discriminator) competing against each other, have emerged as a powerful framework for learning generative models of random data distributions. While GANs are expected to generate images in the conditional setting, using a GAN to directly generate object detection training data with automatically-generated bounding boxes in an expected scenario from random noise still sounds like a fantasy. Instead, learning to translate a labeled image to another style is more feasible.

Inspired by the advantages and drawbacks of previous works, in this paper, we propose AugGAN, a structure-aware unpaired image-to-image translation network, which allows us to directly benefit object detection by translating existing labeled data from its original domain to other ones. We particularly stress on day-to-night image translation not only for the importance of night-time detection, but also for the fact that it is one of the most difficult cross-domain transformations. However, our method is also capable of handling various domain pairs.

In the quantitative analysis, our network is trained on synthetic datasets (i.e., SYNTHIA [14], GTA dataset [15]). Compared to other competing methods, the domain translation results of our network significantly enhance the capability of the object detector for applications on both synthetic (i.e., SYNTHIA, GTA) and real-world (i.e., KITTI [16], ITRI-Day [17], ITRI-Night [17]) data. Finally, AugGAN is general in that it could deal with synthetic-to-synthetic (e.g. GTA-day to SYNTHIA, GTA-sunset, GTA-rain, and SYNTHIA), synthetic-to-real (e.g. SYNTHIA-day to ITRI-Night), real-to-real (e.g. Cityscape [18] to ITRI-Night), and even real-to-synthetic (e.g. Cityscape to SYNTHIA) transformations.

The preliminary study of this work including the earlier versions of AugGAN (AugGAN-1 and AugGAN-2) were published in [17]. This paper provides an extended and refined AugGAN model (AugGAN-3 denoted as AugGAN throughout this paper) capable of achieving better qualitative and quantitative results in our experiments on different cross-domain translation with various datasets. It is evident from the detailed nighttime detector training comparison between real and generated nighttime images, and a thorough subjective evaluation in assessing the transformed results done by other competing methods and our model variations.

II. Related Work

With the advent of R-CNN, a sequence of two-stage detectors including Fast R-CNN [19], Faster R-CNN [20], R-FCN [21], MS-CNN [22], etc., continuously achieve higher accuracy. YOLO regards object detection as a single regression problem as how CNNs are applied for image classification. Then, a multi-scale version of the one-stage detector, SSD [23], demonstrates significant improvements. YOLOv2 proposed in YOLO9000 [24] further boosts the mAP while the FPS is still very high. These detectors keep pushing the limits of object detection in general object detection datasets such as PASCAL VOC [25] and MSCOCO [26].

Recently, Pix2Pix [27] made a breakthrough in direct paired image-to-image translation. i.e., the generator learns to translate a given image to the expected output, such as labels to street scenes, labels to Façade, a B/W image to its color counterpart. However, to obtain paired labeled image pairs in distinct scenarios like day and night is very difficult in practice. More recently, unpaired image-to-image translation methods, such as CycleGAN [28], DiscoGAN [29] and DualGAN [30] have made the training of GAN possible without paired data by introducing the cycle-consistency constraint. CoGAN [31] is a model capable of working on unpaired images by using two weight-sharing generators to generate images of two domains with one random noise. Based on CoGAN, UNIT [32] further introduced the latent space assumption by encouraging two encoders to map images from two domains into the same latent space, which largely increases the translation consistency.

Most on-road vehicle datasets are captured at daytime. LISA 2010 dataset [33] is composed of three video sequences of vehicles. Urban Traffic Surveillance (UTS) dataset [34] provides vehicle images captured by surveillance system. Compcars [35] dataset consists of a large amount of labeled cars with bounding boxes, viewpoints as well as five attributes. The Cars [35] dataset contains more than 100 classes of cars. PASCAL VOC [25] also provides annotated cars and buses in more diverse and challenging scenarios in the sense that their appearances are fully or partially seen at different distances, aspect ratios, sizes and viewing angles. However, vehicles in these datasets are rarely captured from the front-view or in real-driving scenario. KITTI dataset is specifically designed for autonomous driving, having every image collected in practical driving scenario but every image is captured at daytime only. ITRI dataset [17] including ITRI-Day and ITRI-Night provides real-driving images captured at daytime and nighttime under similar driving scenarios. SYNTHIA dataset [14] is collected from synthetic driving scenarios including morning, dusk, evening and night, with four stereo cameras (i.e., font, back, left, right). GTA dataset [15] offers more realistic synthetic images in day, snow, rain, sunset, and night conditions.

III. Proposed Framework

For images in source domain to be properly translated to the target one while image-objects are well-preserved, we assume that the encoded information is required to contain (1) mutual

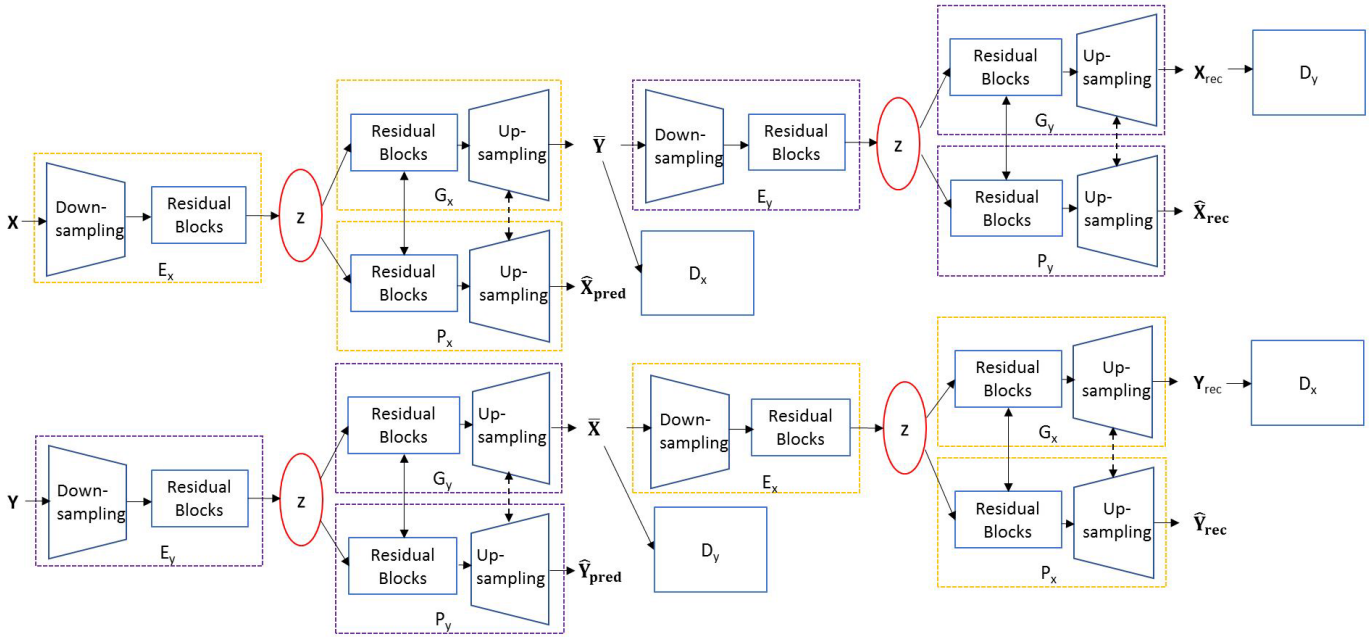

Fig. 1. Overall structure of the proposed image-to-image translation network: X, Y: image domain X and Y; Z: feature domain; $\bar{X}$, $\bar{Y}$: translated results; $\hat{X}_{pred}$, $\hat{Y}_{pred}$: predicted segmentation masks; $\hat{X}_{rec}$, $\hat{Y}_{rec}$: predicted segmentation masks for reconstructed images given $\bar{Y}$ and $\bar{X}$; Dotted & solid lines between blocks implicate soft-sharing and hard-sharing, respectively.

style information between source and target domains, and (2) structural information of the given input image.

Let X and Y denote the two image domains, $\hat{X}$ and $\hat{Y}$ represent the corresponding segmentation masks, and Z is the encoded feature space. Our network, as depicted in Fig. 1, consists of two encoders $E_x : \mathrm{X} \rightarrow \mathrm{Z}$ and $E_y : \mathrm{Y} \rightarrow \mathrm{Z}$, two image-translation generators, $G_x : \mathrm{Z} \rightarrow \bar{\mathrm{Y}}$ and $G_y : \mathrm{Z} \rightarrow \bar{\mathrm{X}}$, two parsing nets, $P_x : \mathrm{Z} \rightarrow \hat{\mathrm{X}}_{\mathrm{pred}}$, and $P_y : \mathrm{Z} \rightarrow \hat{\mathrm{Y}}_{\mathrm{pred}}$, and two discriminators D_x and D_y for the two image domains, respectively.

Our network learns the image translation and the segmentation subtasks in both forward and backward cycles, simultaneously. Take the forward cycle for example, the latent vector of an input $x \in X$ is extracted by E_x. Then, the encoded vector is processed to produce the translated output $\bar{y}$ via G_x and the segmentation result $\hat{x}_{pred}$ via P_x. For the translated output $\bar{y}$, its latent vector is encoded by E_y to yield the reconstructed image x_{rec} via G_y and the segmentation result $\hat{x}_{rec}$ via P_y.

However, as will be discussed in the model analysis section, although the final structure depicted in Fig. 1 leads to the best results quantitatively and qualitatively, our basic model-AugGAN-1, which only introduced the segmentation subtask to the image translation phase in the forward cycle, already outperforms other competing models. Detailed architecture of our proposed complete network is given in Table I. We take SYNTHIA case in describing the size of the feature maps in each layer. The input image size is re-sized to 5-times smaller in GAN learning throughout this work.

A. Structure-Aware Encoding and Segmentation Subtask

Our model actively guides the encoder networks to extract structure-aware features by regularizing them via segmentation subtask so that the extracted feature vector contains not only the mutual style information between X and Y domains, but also the intricate low-level semantic features of the input image that are valuable in the preservation of image-objects during the translation. We experimentally found that using both cross-entropy loss and L1 loss leads to better results and the segmentation loss in the image translation phase of the forward cycle could be formulated as:

$$\begin{aligned}&\mathcal{L}_{seg-x}(P_x, E_x, X, \hat{X})\\&= \lambda_{seg-L1}\mathbb{E}_{x,\hat{x}\sim p_{data(x,\hat{x})}}[\ell_{L1}(P_x(E_x(x)), \hat{x})]\\&\quad+\lambda_{seg-cross-entropy}\mathbb{E}_{x,\hat{x}\sim p_{data(x,\hat{x})}}[\ell_{mce}(P_x(E_x(x)), \hat{x})],\end{aligned} \quad (1)$$

where $\ell_{L1}(P_x(E_x(x)), \hat{x})$ denotes the L1 loss between a C-class $H \times W$ predicted segmentation probability map $\hat{X}_{pred} = P_x(E_x(x))$ and the target segmentation map $\hat{X}$ represented using a 1-hot encoding in the form of $\frac{1}{H\times W\times C}\sum_{i=1}^{H\times W}\sum_{c=1}^{C}|\hat{x}_{pred,c}(i) - \hat{x}_c(i)|$. For the multi-class cross-entropy loss, $\ell_{mce}(P_x(E_x(x)), \hat{x})$, it could be formulated as $-\frac{1}{H\times W}\sum_{i=1}^{H\times W}\sum_{c=1}^{C}\hat{x}_c(i)log(\hat{x}_{pred,c}(i))$.

For the backward cycle, the segmentation loss in the image translation phase is:

$$\begin{aligned}&\mathcal{L}_{seg-y}(P_y, E_y, Y, \hat{Y})\\&= \lambda_{seg-L1}\mathbb{E}_{y,\hat{y}\sim p_{data(y,\hat{y})}}[\ell_{L1}(P_y(E_y(y)), \hat{y})]\\&\quad+\lambda_{seg-cross-entropy}\mathbb{E}_{y,\hat{y}\sim p_{data(y,\hat{y})}}[\ell_{mce}(P_y(E_y(y)), \hat{y}))].\end{aligned} \quad (2)$$

Similarly, for the reconstruction phases of both cycles, the two segmentation losses could be formulated as $\mathcal{L}_{seg-y}(P_y, E_y, \bar{Y}, \hat{X})$ and $\mathcal{L}_{seg-x}(P_x, E_x, \bar{X}, \hat{Y})$, respectively.

TABLE I

NETWORK ARCHITECTURE OF ENCODERS, GENERATORS, PARSING NETS, AND DISCRIMINATORS IN OUR IMAGE-TO-IMAGE TRANSLATION MODEL: THE RESBLK IS A COMBINATION OF CONVOLUTION, BATCH NORMALIZATION, ELU, CONVOLUTION, AND BATCH NORMALIZATION LAYERS WITH SKIP CONNECTION. EVERY CONVOLUTION AND DECONVOLUTION LAYER IS FOLLOWED BY BATCH NORMALIZATION LAYER EXCEPT THE FIRST LAYER OF EACH GENERATOR AND THE LAST LAYER OF EACH GENERATOR, PARSING NET AND DISCRIMINATOR. HARD, SOFT, AND NSEG DENOTE HARD-SHARED, SOFT-SHARED, AND TASK-SPECIFIC NUMBER OF SEGMENTATION CLASSES, RESPECTIVELY

	Encoders				
	Type	Filters	Size	Output	Shared Type
	CONV	64	7×7	256×152	None
	CONV+ReLU	128	3×3/2	128×76	None
	CONV+ReLU	256	3×3/2	64×38	None
3×	RESBLK	512	3×3	64×38	None
	Generators				
	Type	Filters	Size	Output	Shared Type
6×	RESBLK	512	3×3	64×38	Hard
	DCONV+ReLU	128	3×3/2	128×76	Soft
	DCONV+ReLU	64	3×3/2	256×152	Soft
	CONV+Tanh	3	7×7	256×152	None
	Parsing Networks				
	Type	Filters	Size	Output	Shared Type
6×	RESBLK	512	3×3	64×38	Hard
	DCONV+ReLU	128	3×3/2	128×76	Soft
	DCONV+ReLU	64	3×3/2	256×152	Soft
	CONV+ReLU	Nseg	7×7	256×152	None
	CONV+Softmax	Nseg	1×1	256×152	None
	Discriminators				
	Type	Filters	Size	Output	Shared Type
	CONV+LeakyReLU	64	4×4/2	128×76	None
	CONV+LeakyReLU	128	4×4/2	64×38	None
	CONV+LeakyReLU	256	4×4/2	32×19	None
	CONV+LeakyReLU	512	4×4/2	16×9	None
	CONV+LeakyReLU	512	4×4	15×8	None
	CONV+Sigmoid	1	4×4	14×7	None

B. Adversarial Learning

We apply adversarial losses to the mapping functions. For the mapping function $E_x : \mathrm{X} \rightarrow \mathrm{Z}$, $G_x : \mathrm{Z} \rightarrow \mathrm{Y}$ and its discriminator D_x, we express the objective as:

$$\mathcal{L}_{GAN_1}(E_x, G_x, D_x, X, Y) = E_{y\sim p_{data(y)}}[\log D_x(y)] + E_{x\sim p_{data(x)}}[\log(1 - D_x(G_x(E_x(x))))], \quad (3)$$

where E_x and G_x try to generate transformed images $G_x(E_x(x))$ that look similar to images from domain Y, while D_x aims to distinguish between translated samples $G_x(E_x(x))$ and real samples y in terms of style. The same mapping function could be applied to the reconstruction phase of the backward cycle and the only difference is $E_x : \bar{\mathrm{X}} \rightarrow \mathrm{Z}$, $G_x : \mathrm{Z} \rightarrow \mathrm{Y}$ with the objective $\mathcal{L}_{GAN_1}(E_x, G_x, D_x, \bar{X}, Y)$.

Similarly, for the image translation phase of the backward cycle, the mapping function $E_y : \mathrm{Y} \rightarrow \mathrm{Z}$, $G_y : \mathrm{Z} \rightarrow \mathrm{X}$ and its discriminator D_y are related in the following adversial loss:

$$\mathcal{L}_{GAN_2}(E_y, G_y, D_y, Y, X) = E_{x\sim p_{data(x)}}[\log D_y(x)] + E_{y\sim p_{data(y)}}[\log(1 - D_y(G_y(E_y(y))))]. \quad (4)$$

and the reconstruction phase of the forward cycle is modeled as $\mathcal{L}_{GAN_2}(E_y, G_y, D_y, \bar{Y}, X)$.

C. Weight-Sharing for Multi-Task Network

Sharing weights between the generators and the parsing networks allows the generator to fully take advantage of the structure-aware feature vector. We hard-share the encoders and the residual blocks of the generator-parsing net pairs and soft-share the deconvolution layers in the net pairs. The soft weight-sharing is done by calculating the weight difference which is modeled as a cosine similarity loss targeting zero. The mathematical expression for the soft weight-sharing loss function is given by

$$\mathcal{L}_{\omega_x}(\omega_{G_x}, \omega_{P_x}) = -\log(\omega_{G_x} \cdot \omega_{P_x} / \|\omega_{G_x}\|_2 \|\omega_{P_x}\|_2), \quad (5)$$

$$\mathcal{L}_{\omega_y}(\omega_{G_y}, \omega_{P_y}) = -\log(\omega_{G_y} \cdot \omega_{P_y} / \|\omega_{G_y}\|_2 \|\omega_{P_y}\|_2). \quad (6)$$

where ω_G and ω_P denote the weight vectors formed by the deconvolution layers of the generator and parsing networks, respectively.

As will be quantitatively and visually analyzed in the model analysis section, in this work, we also experimented on other weight-sharing strategies, such as hard-sharing the encoders only, and hard-sharing both the encoders and decoders. However, both strategies could only provide inferior results. The best result is achieved by simultaneously applying hard weight-sharing and soft weight-sharing, as mentioned above.

D. Cycle Consistency

The cycle consistency loss has been proven quite effective in preventing network from generating random images in the target domain. We also enforce the cycle-consistency constraint in the proposed framework to further regularize the ill-posed unsupervised image-to-image translation problem. The loss function is given by

$$\mathcal{L}_{cyc1}(E_x, G_x, E_y, G_y, X) = \mathbb{E}_{x\sim p_{data(x)}}[\|G_y(E_y(G_x(E_x(x)))) - \mathrm{x}\|_1], \quad (7)$$

$$\mathcal{L}_{cyc2}(E_x, G_x, E_y, G_y, Y) = \mathbb{E}_{y\sim p_{data(y)}}[\|G_x(E_x(G_y(E_y(y)))) - y\|_1]. \quad (8)$$

E. Network Learning

We jointly solve the learning problems for the image translation subtask:$\{E_x, G_x, D_x\}$ and $\{E_y, G_y, D_y\}$, the image parsing subtask: $\{E_x, P_x\}$ and $\{E_y, P_y\}$, to be cycle-consistent.

The full objective function is given as follows:

$$\begin{aligned}\mathcal{L}_{full}&(E_x, G_x, E_y, G_y, P_x, P_y, D_x, D_y)\\ &= \mathcal{L}_{GAN_1}(E_x, G_x, D_x, X, Y)\\ &\quad +\mathcal{L}_{GAN_2}(E_y, G_y, D_y, Y, X)\\ &\quad +\mathcal{L}_{GAN_1}(E_x, G_x, D_x, \bar{X}, Y)\\ &\quad +\mathcal{L}_{GAN_2}(E_y, G_y, D_y, \bar{Y}, X)\\ &\quad +\lambda_{cyc} * (\mathcal{L}_{cyc1}(E_x, G_x, E_y, G_y, X)\\ &\quad +\mathcal{L}_{cyc2}(E_x, G_x, E_y, G_y, Y))\\ &\quad +\lambda_{seg} * (\mathcal{L}_{seg-x}(E_x, P_x, X, \hat{X}) + \mathcal{L}_{seg-y}(E_y, P_y, Y, \hat{Y})\\ &\quad +\mathcal{L}_{seg-x}(E_x, P_x, \bar{X}, \hat{X}) + \mathcal{L}_{seg-y}(E_y, P_y, \bar{Y}, \hat{Y}))\\ &\quad +\lambda_{\omega} * (\mathcal{L}_{\omega_x}(\omega_{G_x}, \omega_{P_x}) + \mathcal{L}_{\omega_y}(\omega_{G_y}, \omega_{P_y})),\end{aligned} \tag{9}$$

and we aim to solve:

$$\min_{\substack{E_x, G_x, E_y,\\ G_y, P_x, P_y}} \max_{D_x, D_y} \mathcal{L}_{full}(E_x, G_x, E_y, G_y, P_x, P_y, D_x, D_y). \tag{10}$$

IV. Experimental Results

We conducted the training of our network and competing methods on the aforementioned two synthetic datasets for the quantitative analysis, taking advantage of the labeled segmentation masks. In SYNTHIA, only images in stereo-left are adopted and the day-to-night GAN training is performed with the (13072) spring and (13208) night images in sequences other than sequence-1. The data used for detector training and testing would come from the (4756) spring images transformed by GANs and (3740) night images in sequence-1. In the experiments utilizing GTA dataset, all the (40237) daytime and (10277) nighttime images in training sets are involved in GAN training and the (31010) daytime images in validation set would be transformed by GANs to train the detectors to be assessed by (10277) nighttime validation images.

Real-world daytime datasets such as KITTI (7481 images) and our ITRI-Day (25104 images) datasets are used for nighttime detector training after they are transformed by GANs learning from SYNTHIA and GTA, respectively, so there is no train/val split. ITRI-Night (9366 images) is used for nighttime detector evaluation.

We applied both one-stage YOLO [11] and two-stage Faster R-CNN (VGG 16-based) [20] detectors in assessing how well the day-to-night transformation is done by each GAN model in terms of vehicle detection. Aside from revising both detectors to perform single-class vehicle detection, all hyper-parameters were the same as training on PASCAL VOC challenge. The IOU threshold for objects to be considered true-positives is 0.5, where we follow the standard for common object detection datasets. In the transformation of segmentation annotation to its detection counterpart, we exclude the bounding boxes whose heights are lower than 40 pixels or occluded for more than 75 percent in the subsequent AP estimation.

The inference time of running our model on a NVIDIA Tesla P100 GPU is 0.639s, 0.367s, and 0.293s for a 1280 × 760 (SYNTHIA) image, a 1242 × 375 (KITTI) image, and a 960 × 540 (GTA & ITRI) image, respectively. It is worth mentioning that the original image resolution of GTA and ITRI datasets is 1920 × 1080 but the GPU memory is not sufficient.

The demonstration video which summarizes all the subsequent experiments could be accessed from https://youtu.be/CtCwXmhvQMU.

TABLE II

Detection Accuracy Comparison (AP) - Detectors Trained With SD2N: SYNTHIA-Seq-1-Spring Day-to-Night-Transformed by GANs (Trained With SYNTHIA Spring & Night Sequences Other Than Seq-1), and Tested With SN: SYNTHIA-Seq-1-Night Sequence Excluded in GAN Training

Training	Testing	CycleGAN	UNIT	AugGAN	Detector
SD2N	SN	36.1	35.2	39.5	YOLO
SD2N	SN	65.9	57.2	73.1	FRCN

TABLE III

Detection Accuracy Comparison (AP) - Detectors Trained With GTA-val-D2N: GTA-val-Day Day-to-Night-Transformed by GANs (Trained With GTA-Train), and Tested With GTA-val-N: GTA-val Night Sequences

Training	Testing	CycleGAN	UNIT	AugGAN	Detector
GTA-val-D2N	GTA-val-N	20.5	23.6	26.9	YOLO
GTA-val-D2N	GTA-val-N	54.4	62.5	64.0	FRCN

A. Synthetic Datasets

We first assess the effectiveness of training nighttime detectors using day-to-night transformed images in synthetic datasets. As shown in Table II and Table III, AugGAN outperforms competing methods in both SYNTHIA and GTA datasets. Visually, the transformation results of AugGAN is clearly better in terms of image-object preservation and preventing the appearance of artifacts as shown in Fig. 2 and Fig. 3.

B. KITTI and ITRI-Night Datasets

The labeled images in KITTI dataset were collected in versatile driving scenarios and have been widely used in assessing the performance of an on-road object detector used for ADAS or autonomous driving although there is no nighttime version of it. However, since nighttime real-driving dataset is scarce in public domain, we use the self-collected ITRI-Night as the nighttime testing dataset. Besides, all KITTI training data is transformed by using different GANs which have learned the day-to-night transformation from SYNTHIA or GTA.

As the experimental results indicate, real-world data transformed by AugGAN quantitatively and visually provides better results even though AugGAN was trained with synthetic dataset, as shown in Table IV, Fig. 4 and Fig. 5.

C. ITRI Daytime and Nighttime Datasets

We collected a real-driving daytime dataset, ITRI-Day, captured mostly in the similar scenario as our nighttime dataset,

Fig. 2. SYNTHIA day-to-night transformation results - GANs trained with SYNTHIA: 1st row: SYNTHIA daytime testing images; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

TABLE IV

DETECTION ACCURACY COMPARISON (AP) - DETECTORS TRAINED WITH TRANSFORMED IMAGES PRODUCED BY GANS(TRAINED WITH GTA OR SYNTHIA DATASET), AND TESTED WITH REAL NIGHTTIME IMAGES: KITTI-D2N-S: KITTI DAY-TO-NIGHT TRAINING DATA TRANSFORMED BY GANS LEARNING FROM SYNTHIA; KITTI-D2N-G: KITTI DAY-TO-NIGHT TRAINING DATA TRANSFORMED BY GANS LEARNING FROM GTA; ITRIN: ITRI-NIGHT DATASET

Training	Testing	CycleGAN	UNIT	AugGAN	Detector
KITTI-D2N-S	ITRIN	20.2	19.0	31.5	YOLO
KITTI-D2N-G	ITRIN	28.5	20.5	32.2	YOLO
KITTI-D2N-S	ITRIN	59.6	49.2	70.1	FRCN
KITTI-D2N-G	ITRIN	72.0	64.0	81.1	FRCN

ITRI-Night. In Table V, the experiments demonstrate similar results as in other datasets.

The training images transformed by AugGAN are quantitatively better than the ones transformed by others because the object appearance in the images produced by using AugGAN is clearer, sharper and more real. Detector training data transformed by different GANs learning from SYNTHIA and GAN could be seen in Fig. 6 and Fig. 7, respectively.

D. Night-to-Day Transformation

While using AugGAN could bypass the tedious and difficult labeling for the nighttime images by performing day-to-night transformation on the daytime labeled images, it is also potentially-beneficial to make the most of the available (labeled) nighttime images by transforming them to daytime ones. An alternative application to alleviate the difficulties in nighttime image labeling is to label on their night-to-day counterparts.

In SYNTHIA dataset, the nighttime images are not really dark so the transformation could be easily done by AugGAN. Different from day-to-night transformation, in some scenarios in GTA-night and ITRI-Night datasets where no ambient light is present, performing night-to-day transformation becomes not meaningful because some of the information related to

Fig. 3. GTA day-to-night transformation results - GANs trained with GTA: 1st row: GTA daytime testing images; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

Fig. 4. KITTI day-to-night transformation results - GANs trained with SYNTHIA: 1st row: KITTI images; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

Fig. 5. KITTI dataset day-to-night transformation results - GANs trained with GTA dataset: 1st row: input images from KITTI dataset; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

TABLE V
DETECTION ACCURACY COMPARISON (AP) - DETECTORS TRAINED WITH TRANSFORMED IMAGES PRODUCED BY GANS(TRAINED WITH SYNTHIA OR GTA DATASET): ITRID-D2N-S/ITRID-D2N-G: ITRI-DAY DAY-TO-NIGHT TRAINING DATA GENERATED BY GANS TRAINED WITH SYNTHIA/GTA DATASETS; ITRIN: ITRI-NIGHT DATASET

Training	Testing	CycleGAN	UNIT	AugGAN	Detector
ITRID-D2N-S	ITRIN	35.5	41.3	47.2	YOLO
ITRID-D2N-G	ITRIN	37.9	42.6	48.6	YOLO
ITRID-D2N-S	ITRIN	72.4	74.5	82.2	FRCN
ITRID-D2N-G	ITRIN	86.2	85.9	86.5	FRCN

image structure has been totally lost. However, apart from those extreme cases, some nighttime images are still worthy of being transformed, as can be seen in Fig. 8. Most importantly, AugGAN still consistently outperforms other models in terms of better object preservation.

E. Transformations Other Than Daytime and Nighttime

AugGAN is capable of learning transformation across unpaired domain pairs while either domain could be real or synthetic although segmentation supervision is the prerequisite of using AugGAN. However, the minimum requirement is the segmentation annotation in the source domain and we denote this version as AugGAN-1 as will be discussed later in the model analysis section. This version increases the flexibility of learning cross-domain adaptation in terms of object detection. As shown in Fig. 9, our model could learn image-to-image translation from synthetic-to-synthetic (e.g. GTA-day to SYNTHIA, GTA-sunset, GTA-rain, and SYNTHIA), synthetic-to-real (e.g. SYNTHIA-day to ITRI-Night), real-to-real (e.g. Cityscape to ITRI-Night), and even real-to-synthetic (e.g. Cityscape to SYNTHIA). It is worth mentioning that all the quantitative results in the above sections are achieved by AugGAN-3 which needs the segmentation supervision from both domains.

Fig. 6. ITRI-Day dataset day-to-night transformation results - GANs trained with SYNTHIA: 1st row: input images from ITRI-Day dataset; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

F. On-Road Nighttime Vehicle Detection Result Analysis

Since a vehicle detector is expected to function in real-driving scenario, we demonstrate some detection results of training a vehicle detector (single-class Faster R-CNN) using three different kinds of training data (KITTI-D2N-S transformed by CycleGAN, UNIT, and AugGAN, respectively). As can be seen in Fig. 10, nighttime vehicle detector trained by using KITTI day-to-night-transformed images generated by CycleGAN and UNIT would normally provide poor results when vehicles look dark since both models would fail in preserving the object contour. The same detector trained by using the same images transformed by AugGAN could achieve quantitatively-better detection results because in the transformed images, there are fewer artifacts which are harmful for the detector to learn the essence of vehicle appearance.

G. Training Detectors With Generated Night Images v.s Real Night Images

To compare a detector trained with generated nighttime images and real ones, we train YOLO detector with images (i.e., 2k, 4k, 4.5k, etc.) randomly sampled from ITRI-Night-1 dataset which was captured under the same scenarios as ITRI-Night and contains 9k images, as in Table VI. However, the training in CNN is non-deterministic in the sense that the resulted AP would be slightly different every time. Therefore, we simply perform each training for five times and report the averaged results. Quantitative results show that the AP of YOLO vehicle detector trained with (ITRI-Day) day-to-night transformed images by CycleGAN (learning from

Fig. 7. ITRI-Day dataset day-to-night transformation results - GANs trained with GTA dataset: 1st row: input images from ITRI-Day dataset; 2nd row: results of CycleGAN; 3rd row: results of UNIT; 4th row: results of AugGAN.

SYNTHIA) is between the AP achieved by using 4k and 4.5k real nighttime images. AugGAN (learning from SYNTHIA) performs similarly to that with 9k real night images. Furthermore, in order to know if the superiority of AugGAN remains when fewer day-to-night images are given to train a nighttime vehicle detector, we further randomly sample 9k images from the day-to-night images generated by AugGAN and CycleGAN, respectively, and also report the averaged AP. In this experiment, AugGAN & CycleGAN reach the AP close to the ones done by using 4k & 2K real nighttime images. Roughly speaking, AugGAN is about two times better than CycleGAN in terms of the numbers of real nighttime images required to achieve the same AP. One could conclude that using day-to-night transformed images could make the most of the common daytime training data in training a night-time vehicle detector and AugGAN outperforms CycleGAN with respect to day-to-night domain adaptation.

V. Model Analysis and Subjective Evaluation

In the attempt to explore a better architecture in achieving visually-better and quantitatively-beneficial results, we performed some analysis targeting (1) the weight-sharing strategy and (2) the loss function. These experiments were all done on SYNTHIA dataset and gradually lead us to the best results in other datasets. Moreover, in order to holistically evaluate the visual quality of our results with other competing models, we conducted a semantic segmentation analysis and a subjective evaluation on the same dataset.

Fig. 8. Night-to-day transformation results: left column: SYNTHIA night and its night-to-day transformation results using UNIT (learning from SYNTHIA) and AugGAN (learning from SYNTHIA), respectively; center column: GTA night and its night-to-day transformation results using UNIT (learning from GTA) and AugGAN (learning from GTA), respectively; right column: ITRI-Night and its night-to-day transformation results using UNIT (learning from GTA) and AugGAN (learning from GTA), respectively.

TABLE VI

Average Precision Comparison for Night-Time Vehicle Detectors (YOLO) Trained With Real Nighttime Images and Day-to-Night Transformed Image Generated by GANs Learning From SYNTHIA: ITRIN: ITRI-Night Dataset; ITRIN1: ITRI-Night-1

Training	Testing	AP
ITRIN1 random 2k	ITRIN	22.6
ITRIN1 random 4k	ITRIN	33.8
ITRIN1 random 4.5k	ITRIN	35.9
ITRIN1 random 5k	ITRIN	38.7
ITRIN1 random 6k	ITRIN	41.2
ITRIN1 random 8k	ITRIN	43.3
ITRIN1 9k	ITRIN	45.6
CycleGAN (ITRID-D2N-S: random 9k)	ITRIN	23.0
CycleGAN (ITRID-D2N-S: all)	ITRIN	35.5
AugGAN (ITRID-D2N-S: random 9k)	ITRIN	33.9
AugGAN (ITRID-D2N-S: all)	ITRIN	47.2

A. Weight-Sharing Strategy Comparison

Our network design is based on the assumption that extracted semantic segmentation features of individual layers, through proper weight-sharing, can serve as auxiliary regularization for image-to-image translation. Thus finding the proper weight-sharing policy came to be the most important factor in our design. Weight-sharing mechanism in neural networks can be roughly categorized into soft weight-sharing [36] and hard weight-sharing. The former was originally proposed for regularization and could be applied to network compression [37]. The latter is the most commonly used approach to multi-task learning in neural networks and goes back to [38]. Recently, UNIT [31] has successfully applied hard weight-sharing in their model for generating images with similar high-level semantics.

Fig. 9. More image-translation results: 1st row: GTA-day to SYNTHIA; 2nd row: GTA-day to GTA-sunset; 3rd row: GTA-day to GTA-rain; 4th row: SYNTHIA-day to ITRI-Night; 5th row: Cityscape to SYNTHIA; 6th row: Cityscape to ITRI-Night.

Fig. 10. Faster R-CNN detection result comparison: 1st row: ITRI-Night detection results using detector trained by KITTI-D2N-S (transformed by CycleGAN) training data; 2nd row: ITRI-Night detection results using detector trained by KITTI-D2N-S (transformed by UNIT) training data; 3rd row: ITRI-Night detection results using detector trained by KITTI-D2N-S (transformed by AugGAN) training data.

TABLE VII

FASTER R-CNN DETECTION ACCURACY COMPARISON OF AUGGAN TRAINED WITH SD2N AND TESTED WITH SN USING DIFFERENT WEIGHT-SHARING STRATEGIES: λ_w DENOTES THE COSINE SIMILARITY LOSS MULTIPLIER WHEN DECONVOLUTION LAYERS ARE SOFT-SHARED, AND THE BEST RESULT IS YIELDED WHEN $\lambda_w = 0.02$

Weight-sharing strategy	AP
Encoder: hard	39.9
Encoder: hard; Decoder: hard	57.2
Encoder: hard; RESBLK: hard; DCONV: soft (λ_w=0.02)	68.7

In our initial experiments, hard weight-sharing is applied on the encoders of our multi-tasking network, but both tasks failed. Then, we tried to apply hard weight-sharing on both encoders and decoders but the results showed that both networks could not be optimized at the same time. As shown in Table VII and Fig. 11, the strategy leading us to the best results is two-folded: (1) hard-sharing encoders and the residual blocks of the generator-parsing net pairs, (2) soft-sharing deconvolution layers in the same net pairs. This setting is decided based on extensive experiments, and during the process we realized that both policies are integral to the optimization of our network.

B. Loss Function Analysis

In our initial model, AugGAN-1, the parsing network was solely utilized and jointly optimized in the (day-to-night) image translation phase of the forward cycle. i.e., only the segmentation annotation in the source domain is required. Therefore, only equation 1 among segmentation losses is used and the SYNTHIA-day to ITRI-Night and Cityscape to ITRI-Night cases in Fig. 9 were done by this version.

Since segmentation annotation is available in both day and night scenarios in SYNTHIA and GTA, the segmentation subtask is also possible to be applied to the image translation phase (i.e., night-to-day) of the backward cycle. i.e., both equation 1 and 2 are involved. We denote this version as AugGAN-2, which is proposed with AugGAN-1 in our previous work [17].

Then, we noticed that the image structure is well-preserved in the image translation and the image reconstruction phases in both cycles so segmentation subtask could be further applied to the reconstruction phases. In other words, four segmentation losses introduced in section-III.A are all involved during training. Also, the discriminators could be applied on the reconstructed images to tell apart the difference between reconstructed day/night images and the real day/night ones. i.e., two additional adversarial loss $\mathcal{L}_{GAN_1}(E_x, G_x, D_x, \bar{X}, Y)$

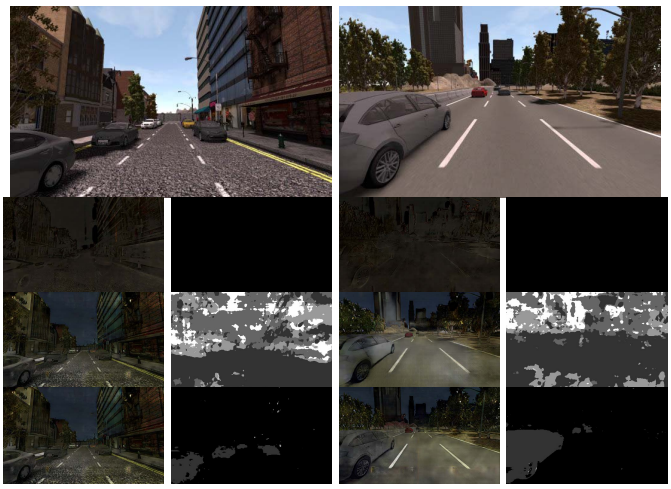

Fig. 11. Style transfer and segmentation results for different weight-sharing strategies: 1st row: input images; 2nd row: results of hard weight-sharing in encoders only; 3rd row: results of hard weight-sharing in both the encoders and decoders; 4th row: results of hard weight-sharing in encoders & the residual blocks of the generator-parsing net pairs and soft weight-sharing in the deconvolution layers ($\lambda_w = 0.02$) in the same net pairs.

and $\mathcal{L}_{GAN_2}(E_y, G_y, D_y, \bar{Y}, X)$ in equation 9 are involved in training. This version is denoted as AugGAN-3.

As can be seen in Table VIII, it is quite obvious that AugGAN-1, which only adds segmentation subtask to guide image translation (with the proposed weight-sharing strategy) in forward cycle, already leads to results better than CycleGAN and UNIT. AugGAN-2 brings further accuracy improvement by introducing segmentation in regularizing the image translation phases in both cycles. Finally, the best quantitative result is achieved by AugGAN-3 which performs segmentation subtask to regularize image translation and reconstruction phases in both cycles and applies adversarial loss to the translated and the reconstructed images. As can be seen in Fig. 12, it is visually obvious that AugGAN-2 is better than AugGAN-1 and AugGAN-3 is the best because the transformed image is clearer, sharper and looks more real. It is worth mentioning that all the quantitative analysis in the experimental section is done by using AugGAN-3. Finally, the impacts of the semantic segmentation subtasks and the discriminators in the reconstruction phases are also analyzed. Without the former, the detection accuracy is 39.1 & 72.4 in YOLO and Faster R-CNN, respectively. Without the latter, only 39.3 & 72.9 are achieved. In short, both losses are critical to the optimization of our network and using both of them would lead us to better results.

C. Semantic Segmentation Analysis across Domains

Our model has been proven effective in boosting the night-time vehicle detection accuracy. In order to quantitatively evaluate the quality of the entire transformed image, we also adopt FCN8s (VGG16-based) [39] to report FCN score as Pix2Pix and CycleGAN did. The intuition is that if the day-to-night transformed images are realistic, then FCN8s could be trained by them to achieve better segmentation results on real nighttime images. This analysis is done in SYNTHIA dataset.

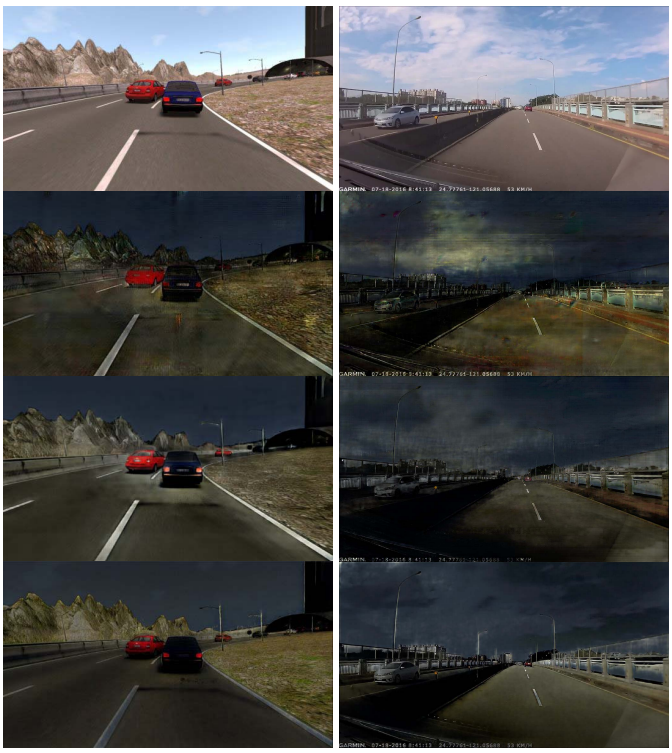

Fig. 12. SYNTHIA and ITRI-Day transformed results using AugGAN variations learning from SYNTHIA: left column: SYNTHIA input image and the transformed results using AugGAN-1, AugGAN-2, and AugGAN-3, respectively; right column: ITRI-Day input image and the transformed results using AugGAN-1, AugGAN-2, and AugGAN-3 learning from STNTHIA, respectively.

TABLE VIII

DETECTION ACCURACY COMPARISON (AP) ON SN USING DETECTORS TRAINED WITH SD2N GENERATED BY DIFFERENT GANS INCLUDING AUGGAN VARIATIONS AND OTHERS)

CycleGAN	UNIT	AugGAN-1	AugGAN-2	AugGAN-3	Detector
36.1	35.2	38.1	39.0	39.5	YOLO
65.9	57.2	68.7	72.2	73.1	FRCN

As shown in Table IX, AugGAN outperforms CycleGAN and UNIT in terms of per-class accuracy, mIoU and fwIoU. In our experiments, the images are all re-sized to 600×600 and the FCN8s are trained for 100k iterations using SGD with learning rate 1e-10 and momentum 0.9.

D. Subjective Evaluation

In order to know if visually-better day-to-night transformation results are positively-related to higher night-time detection accuracy, we conducted a subjective evaluation to provide a visual rating for our method and other competing ones. With 47 random non-expert observers involved, the questions are designed to demonstrate two factors directly related to the subsequent detection performance. The first one is the level of object preservation. The second one is the style-transfer quality. The former question is designed to know if objects are successfully preserved. Otherwise, the capability of

TABLE IX

FCN-SCORES FROM FCN8S TRAINED WITH SD2N: SYNTHIA-SEQ-1-SPRING DAY-TO-NIGHT-TRANSFORMED BY GANS (TRAINED WITH SYNTHIA SPRING & NIGHT SEQUENCES OTHER THAN SEQ-1), AND TESTED WITH SN: SYNTHIA-SEQ-1-NIGHT SEQUENCE EXCLUDED IN GAN TRAINING

Model	Training	Testing	Per-class acc.	mIoU	fwIoU
CycleGAN	SD2N	SN	62.0	53.5	82.5
UNIT	SD2N	SN	65.8	55.8	85.6
AugGAN	SD2N	SN	68.4	60.4	89.0

TABLE X

DEGREE OF OBJECT PRESERVATION AND STYLE-TRANSFER QUALITY AFTER DAY-TO-NIGHT TRANSFORMATION IN SYNTHIA, KITTI AND ITRI-DAY DATASETS USING GANS LEARNING FROM SYNTHIA

Data	GAN model	MOS
SD2N	CycleGAN	2.81
SD2N	UNIT	2.79
SD2N	AugGAN	3.83
KITTI-D2N-S	CycleGAN	0.92
KITTI-D2N-S	UNIT	1.84
KITTI-D2N-S	AugGAN	2.85
ITRID-D2N-S	CycleGAN	1.60
ITRID-D2N-S	UNIT	2.21
ITRID-D2N-S	AugGAN	3.87

cross-domain adaptation of the corresponding GAN model is therefore questionable. The latter one is for making sure that the daytime image is style-transferred to nighttime-looking without noticeable artifacts because it is theoretically possible that objects are well-preserved but the images are not transformed towards the expected nighttime style.

In both the SYNTHIA case and the GTA one, day-to-night transformed result video clips from CycleGAN, UNIT, AugGAN are displayed with the original daytime one at the same time. The observer is expected to score each video clip on a scale ranging from one to five (very low, relatively low, medium, relatively high, and very high) according to the aforementioned two factors.

As to the KITTI case, images instead of video clips are provided because KITTI images are sampled from real driving video. In our experimental design, five of them are randomly selected and each one is processed by CycleGAN, UNIT and AugGAN, respectively. Besides, there are two versions of each GAN model. Each KITTI image is processed by one of the GAN models learning day-to-night transformation from SYNTHIA and GTA datasets, separately.

For the ITRI-Day dataset, it is again processed by each GAN model learning from two synthetic datasets. Similarly, every processed image as well as their daytime counterpart is shown to the observers.

In each assessment, we calculate the mean opinion score (MOS) for each comparison. As shown in Table X, Our AugGAN consistently outperforms UNIT and CycleGAN in that the objects are preserved in the transformation because the parsing network would guide the (image-translation) generator not to alter the image structure. UNIT could provide visually better results than CycleGAN because it tries to preserve high-level semantic in the transformation. However, they sometimes fail to keep the fine texture of large objects and thus trailed the AP in the subsequent vehicle detector analysis with AugGAN.

TABLE XI

DEGREE OF OBJECT PRESERVATION AND STYLE-TRANSFER QUALITY AFTER DAY-TO-NIGHT TRANSFORMATION IN GTA, KITTI AND ITRI-DAY DATASETS USING GANS LEARNING FROM GTA

Dataset	GAN model	MOS
GTA-val-D2N	CycleGAN	2.72
GTA-val-D2N	UNIT	2.64
GTA-val-D2N	AugGAN	3.91
KITTI-D2N-G	CycleGAN	1.83
KITTI-D2N-G	UNIT	2.10
KITTI-D2N-G	AugGAN	2.62
ITRID-D2N-G	CycleGAN	1.96
ITRID-D2N-G	UNIT	2.17
ITRID-D2N-G	AugGAN	3.38

GTA dataset is way more realistic than SYNTHIA. This explains why in the detector training, transforming datasets using GANs learning from GTA could achieve better detector training results in both one-stage and two-stage detectors. As can be seen in Table XI, most observers still favor transformation results made by AugGAN although the night scene of GTA is darker than the one of SYNTHIA so they sometimes encounter difficulties in telling if the transformation is good in the detail of object appearance.

VI. CONCLUSION AND FUTURE WORK

In this work, we proposed AugGAN, an unpaired image-to-image translation network for realizing domain adaptation in vehicle detection. Our method quantitatively surpasses competing methods for achieving higher nighttime vehicle detection accuracy because of better image-object preservation. Therefore, most daytime vehicle datasets in public domain become valuable in nighttime vehicle detector development. AugGAN is general in that it could also deal with synthetic-to-synthetic, synthetic-to-real, real-to-real, and real-to-synthetic transformations across different domains ranging from day, night, sunset, rain, etc. Currently, the major limitation of AugGAN is its uni-modality. In the future, we will try to explicitly encode random noise vector to our structure-aware latent vector in order to gain multi-modality in performing unpaired image-to-image translation, such as day-to-night, while image-objects are still well-preserved. This way, a nighttime vehicle detector could learn to better detect vehicles under different degrees of ambient light in the same domain.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for their constructive and insightful comments to this paper.

REFERENCES

[1] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 694–711, May 2006.

[2] X. Wen, L. Shao, W. Fang, and Y. Xue, "Efficient feature selection and classification for vehicle detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 508–517, Mar. 2015.

[3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[4] A. Takeuchi, S. Mita, and D. McAllester, "On-road vehicle tracking using deformable object model and particle filter with integrated likelihoods," in *Proc. IEEE IV Symp.*, Jun. 2010, pp. 1014–1021.

[5] H. T. Niknejad, T. Kawano, M. Shimizu, and S. Mita, "Vehicle detection using discriminatively trained part templates with variable size," in *Proc. IEEE IV Symp*, Jun. 2012, pp. 766–771.

[6] H. T. Niknejad, T. Kawano, Y. Oishi, and S. Mita, "Occlusion handling using discriminative model of trained part templates and conditional random field," in *Proc. IEEE IV Symp*, Jun. 2013, pp. 750–755.

[7] S. Sivaraman and M. M. Trivedi, "Vehicle detection by independent parts for urban driver assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1597–1608, Dec. 2013.

[8] H. T. Niknejad, S. Mita, D. McAllester, and T. Naito, "Vision-based vehicle detection for nighttime with discriminately trained mixture of weighted deformable part models," in *Proc. IEEE ITSC*, Oct. 2011, pp. 1560–1565.

[9] H. Tehrani, T. Kawano, and S. Mita, "Car detection at night using latent filters," in *Proc. IEEE IV Symp.*, Jun. 2014, pp. 839–844.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[12] M. Braun, S. Krebs, F. Flohr, and D. M. Gavrila, "The EuroCity persons dataset: A novel benchmark for object detection," 2018, *arXiv:1805.07193*. [Online]. Available: https://arxiv.org/abs/1805.07193

[13] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2014, pp. 2672–2680.

[14] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3234–3243.

[15] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 102–118.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. Int. Conf. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.

[17] S.-W. Huang, C.-T. Lin, S.-P. Chen, Y.-Y. Wu, P.-H. Hsu, and S.-H. Lai, "Auggan: Cross domain adaptation with gan-based data augmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 718–731.

[18] M. Cordts *et al.*, "The cityscapes dataset," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. Workshops*, vol. 1, no. 2, Jun. 2015, p. 3.

[19] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 91–99.

[21] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2016, pp. 379–387.

[22] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 354–370.

[23] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 21–37.

[24] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525.

[25] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.

[26] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2014, pp. 740–755.

[27] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.

[28] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[29] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," 2017, *arXiv:1703.05192*. [Online]. Available: https://arxiv.org/abs/1703.05192

[30] Z. Yi, H. Zhang, T. Gong, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Jul. 2017, pp. 2868–2876.

[31] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2016, pp. 469–477.

[32] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2017, pp. 700–708.

[33] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 267–276, Jun. 2010.

[34] Y. Zhou, L. Liu, L. Shao, and M. Mellor, "DAVE: A unified framework for fast vehicle detection and annotation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 278–293.

[35] L. Yang, P. Luo, C. Change Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3973–3981.

[36] S. J. Nowlan and G. E. Hinton, "Simplifying neural networks by soft weight-sharing," *Neural Comput.*, vol. 4, no. 4, pp. 473–493, 1992.

[37] K. Ullrich, E. Meeds, and M. Welling, "Soft weight-sharing for neural network compression," 2017, *arXiv:1702.04008*. [Online]. Available: https://arxiv.org/abs/1702.04008

[38] R. Caruna, "Multitask learning: A knowledge-based source of inductive bias," in *Proc. ICML*, Jun. 1993, pp. 41–48.

[39] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

Che-Tsung Lin received the B.S. degree in mechanical engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 2003, and the M.S. degree from the Institute of Applied Mechanics, National Taiwan University, Taipei, in 2005. He is currently pursuing the Ph.D. degree with the Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan. He was a Visiting Scholar with the Department of Computer Science, University of California, Santa Barbara, CA, USA, in 2013. He has been working with the Mechanical and Mechatronics System Research Laboratory, Intelligent Vehicle Division, Safety Sensing and Control Department, Industrial Technology Research Institute, Hsinchu, since 2006, where he is currently a Researcher in the field of intelligent transportation system, intelligent vehicle, and advanced driver assistance system. His research interests include computer vision, machine learning, deep learning, and generative adversarial networks and their applications in on-road object detection.

Sheng-Wei Huang received the B.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2017. He was with Prof. Shang-Hong Lai as a Research Assistant at the Department of Computer Science, National Tsing Hua University. His research interests in computer vision include semantic segmentation and generative adversarial networks.

Yen-Yi Wu received the B.S. degree in computer science from the University of Taipei, Taipei, Taiwan, in 2017, and the M.S. degree in computer science from National Tsing Hua University, Hsinchu, Taiwan, in 2019. Her research interests in computer vision include generative adversarial networks, domain adaption, and object detection.

Shang-Hong Lai received the Ph.D. degree in electrical and computer engineering from the University of Florida, Gainesville, in 1995. Then, he joined the Siemens Corporate Research, Princeton, New Jersey, as a member of technical staff. Since 1999, he became a faculty member with the Department of Computer Science, National Tsing Hua University (NTHU), Taiwan, where he is currently a Professor. Since 2018, he has been on leave from NTHU to join the Microsoft AI R&D Center, Taipei, as a Principal Researcher. He has authored more than 200 articles published in the related international journals and conferences. His research interests include computer vision, image processing, and machine learning. He has been a member of program committee of several international conferences, including CVPR, ICCV, ECCV, ACCV, ICPR, PSIVT, and ICME. He has been an Associate Editor or a Guest Editor for several international journals, including *Pattern Recognition, Journal of Signal Processing Systems*, *Journal of Visual Communication and Image Representation*, and *IPSJ Transactions on Computer Vision and Applications*.