Privacy-Preserving Auto-Driving: a GAN-based Approach to Protect Vehicular Camera Data

Zuobin Xiong*, Wei Li*, Qilong Han† and Zhipeng Cai*
*Department of Computer Science, Georgia State University, Atlanta, USA
†College of Computer Science and Technology, Harbin Engineering University, Harbin, China
* *zxiong2@student.gsu.edu, {wli28, zcai}@gsu.edu*; † *hanqilong@hrbeu.edu.cn*

***Abstract*—The autonomous driving (auto-driving) technology has been promoted significantly by the rapid advances in computer vision and deep neural networks. Auto-driving vehicles, nowadays, are fully equipped with numerous sensors such as cameras, geo-sensors, and radar sensors, to capture real-time data inside the vehicles and outside surroundings. Meanwhile, the captured data contains lots of private information about vehicles, drivers and passengers and thus faces a high risk of privacy breaches. Especially, side-channel information can be mined from camera data to identify vehicles' locations and even trajectories, raising serious privacy issues. Unfortunately, the issue, how to resist location-inference attack for camera data in auto-driving, has never been addressed in literature. In this paper, we intend to fill this blank by developing a GAN-based image-to-image translation method named Auto-Driving GAN (ADGAN). Through performance comparisons between ADGAN and the state-of-the-art, the superiority of ADGAN can be validated – offering an effective tradeoff between recognition utility and privacy protection for camera data.**

***Index Terms*—Autonomous driving; Location privacy; Generative Adversarial Networks; Image generation.**

I. Introduction

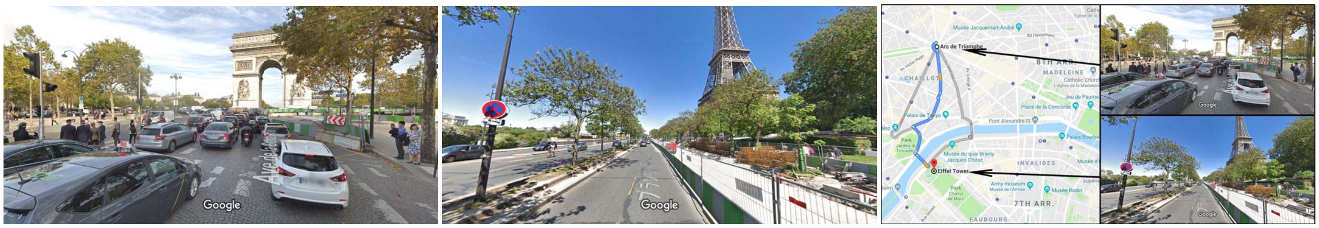
Over the last decade, the autonomous driving (auto-driving) technology, combined with computer vision and deep learning, has flourished in both industry and academia. This significantly promotes some leading manufactures, including Tesla, Ford, BMW and even Google, to produce their own auto-driving vehicles [1]. These auto-driving vehicles have become a robust and efficient stuff and already driven millions of miles without human intervention [2], [3]. Such an incredible success is inseparable from two core elements: perception and decision-making, which are in desire of numerous data for performance improvement. In other words, the auto-driving vehicles are nothing but a car driven by an amount of data. The data can be collected from a variety of sensors embedded in the autonomous vehicles, *e.g.*, GPS for navigation; a wheel encoder for monitoring movement; behavior-relevant sensors for capturing passengers' behaviors; radar on the front and rear bumpers for identifying traffic; and camera near the rear-view mirror for color identification, lane departure, read collision, and pedestrian alerts [4], [5]. Besides driving guidance, such valuable data can benefit individuals and the society in various ways, including traffic analysis, accident investigation, auto insurance assessment, vehicular communication, and "smart city function", *etc.*

Yet, despite these attractive benefits, the volume-rich data inevitably exposes the privacy of vehicles and drivers/passengers to an extremely dangerous situation. The auto-driving vehicles are very likely to become the targets of malicious attackers no matter with what purposes. Once attackers access to the collected data, personal privacy behind the data will be leaked. For examples, by analyzing GPS data, the attackers know passengers' home addresses and moving patterns; and by analyzing behavior-relevant data, the attackers can infer the information of sex, age, hobby, *etc.*

As an indivisible part of the autonomous vehicles, camera data definitely suffers from severe privacy threats. By capturing real-time images, the cameras work as the eyes to help monitor road conditions (*e.g.,* recognizing pedestrians) and guide driving behaviors (*e.g.*, stopping and braking). The captured images can be also collected for use in real applications, such as building 3D street view, training detection model, and arbitrating disputes in traffic accidents. However, the cameras not only have the power to record images and videos of ambient environment view for their host vehicles, but also can collect other "over-needed information", such as street view background, faces of pedestrians by streets, license plate and model of surrounding vehicles, and others. This "over-needed information" becomes a breakthrough for attackers to steal privacy. Fig. 1 illustrates an attack scenario: an attacker gets an image captured by a victim's camera as shown in Fig. 1(a), and can learn that the victim was on a street at the front of "Triumphal Arch". In this scenario, the victim's location privacy is totally leaked via side-channel information in the image without GPS data. Moreover, if the attacker can obtain a set of victim's images with time correlation (*e.g.*, Fig. 1(a) and Fig. 1(b)), he can infer the victim's possible trajectory and driving speed as presented in Fig. 1(c).

Even worse is that with the developments of computer vision and deep learning, attackers can strengthen their attack ability by means of object recognition and image geo-localization. Early before decade, some vocabulary tree-based matching and feature-based matching methods have been proposed to detect location in images, which can reach high recognition accuracy above 70% [6], [7]. That is, attackers are strong enough to easily recover real trajectory with large probability by recognizing camera images.

2374-8486/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDM.2019.00077



(a) camera data from victim near Triumphal arch (b) camera data from victim near Eiffel tower (c) leaked location and trajectory information

Fig. 1. An example to illustrate how victim's location and trajectory privacy is inferred by Geo-localization attack (pictures source from Google Map).

Thus, **in auto-driving, preventing camera data[1] from being attacked by location inference has become an urgent problem to be solved**. To protect camera data, three unprecedented challenges are ahead of us. (i) As no existing work addresses this problem for auto-driving, the study of problem formulation and technique design is an unknown exploration. (ii) It is desired that the privacy-preserving camera images are still usable in real applications. For this purpose, balancing the tradeoff between image recognition and privacy protection is critical to technique design, which is not a trivial issue. (iii) Although some existing methods aim to protect privacy of small objects (such as face, number, and license plate) by blurring or removing the objects from images, these methods will lose the context structure of images and damage the usability of images if the object is large (*e.g.*, buildings). Since the street view images have more complex context structures containing a variety of objects, the aforementioned methods are not suitable for protecting vehicular camera data.

To overcome these three challenges, in this paper, we novelly propose a Generative Adversarial Networks (GAN)-based approach named **ADGAN**. Our basic idea is that reducing the risk of privacy breaches by removing location-relevant information (*e.g.*, background buildings) from the camera images before being used in real applications. To be concrete, we utilize image-to-image translation to eliminate private objects in images while maintaining the utility of valuable objects, so that the processed images can still be used for real applications. Then, to effectively balance the tradeoff between privacy and utility, we design a min-max loss function to control image synthesis in ADGAN. Particularly, we develop an innovative multi-discriminator setting in our ADGAN for performance enhancement: (i) the loss of context structure is reduced, improving image recognition; and (ii) the guarantee of privacy protection is reinforced. Finally, our multi-fold contributions are summarized below:

- To the best of our knowledge, this paper is the first work to investigate the privacy issue of camera data for auto-driving.
- A GAN-based image-to-image translation method (ADGAN) is designed to generate privacy-preserving images, which can resist location-inference attack towards side-channel information of camera data.
- The results of real-dataset experiments validate that our ADGAN can achieve privacy protection while simultaneously maintaining accuracy of image recognition, which provides a more effective tradeoff between privacy and utility compared with the state-of-the-art.

The rest of this paper is organized as follows. The preliminaries and related works are introduced in Section II and Section III, respectively. The details of our ADGAN are presented in Section IV. After evaluating ADGAN in Section V, this paper is concluded in Section VI.

II. Preliminaries

In this section, the background and fundamentals of Generative Adversarial Network (GAN), Auto-Encoder (AE), and image-to-image translation are briefly introduced.

A. Generative Adversarial Network

As the most creative idea of Deep Learning in recent years, Generative Adversarial Network (GAN) has been widely applied in the field of computer vision since it was proposed in 2014 [8]. GAN consists of two "adversarial" models: *a generator* G and *a discriminator* D. The two adversarial models play with each other to complete in a min-max game, where G intentionally generates samples from a real data distribution to fool D while D judges whether its input is the fake data generated by G or the real data. Mathematically speaking, G could be any form but a simple differentiable function, and $G(z)$ is the output sample drawn from p_g where z is a low dimensional vector sampled from a prior distribution p_z. Thus, the aim of D is to classify the data from $G(z)$ as fake and the data from training set p_{data} as real. Formally, GAN is expressed as a structured probabilistic model to optimize the following loss function:

$$\begin{aligned}\min_G \max_D \mathcal{L}_{GAN}(G,D) =& \mathbb{E}_{x\sim p_{data}(x)}[logD(x)]+\\ & \mathbb{E}_{z\sim p_z(z)}[1-logD(G(z))],\end{aligned} \tag{1}$$

where G aims to minimize $\mathcal{L}_{GAN}(G,D)$ while D aims to maximize it.

Furthermore, GAN can be extended to a conditional version with an additional input y that could be any kind of auxiliary information (*e.g.*, the class labels or data from other domains).

[1] In this paper, we focus on the camera images in auto-driving, and thus camera data and camera image are exchangeable.

The corresponding objective function of such min-max game is formulated as:

$$\min_G \max_D \mathcal{L}_{GAN}(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x|y)] + \mathbb{E}_{z\sim p_z(z)}[1-logD(G(z|y))]. \quad (2)$$

B. Auto-Encoder and U-Net

The Auto-Encoder (AE) that is an unsupervised neural network learns a mapping function from input data x to output $\tilde{x} = h(x)$. The goal of learned mapping function is to get the minimum distance between x and $\tilde{x}$. AE consists of two parts: Encoder and Decoder [9]. The Encoder samples data x from a real distribution and then encodes it into a latent representation z, *i.e.*, $z \sim Encoder(x) = q(z|x)$. The Decoder reconstructs the real data x from the low-dimension representation z, *i.e.*, $\tilde{x} \sim Decoder(z) = p(x|z)$. Then the loss function of AE can be defined as:

$$\min_{Enc,Dec} \mathcal{L}_{AE}(Enc, Dec) = Dis(x, Dec \circ Enc(x)), \quad (3)$$

where Enc and Dec represent Encoder and Decoder, respectively; Dis could be any distance metrics (such as L_1 distance [9] and Kullback-Leibler divergence [10]); and $\circ$ is the composite function of Encoder and Decoder.

In this paper, we adopt a variation of AE, called "U-Net". Besides using the idea of traditional AE, "U-Net" also adds some skip links between layers in Encoder and Decoder. To improve the performance of data reconstruction, the skip links are used to concatenate the i-th layer of Encoder and the $(n-i)$-th layer of Decoder. The similarity and connection between layers are enhanced by such a concatenation, thus promoting U-Net to generate more similar result $\tilde{x}$. A comparison of the structure between AE and U-Net is shown in Fig. 2. In the traditional AE, latent z is obtained by passing input x through the Encoder and then is recovered to $\tilde{x}$ by Decoder. While in U-Net, each layer of Encoder produces an intermediate result after convolution and pooling, and every intermediate result is sent to the corresponding layer of Decoder where the result is concatenated with additional recovered data and goes through the rest of neural network.

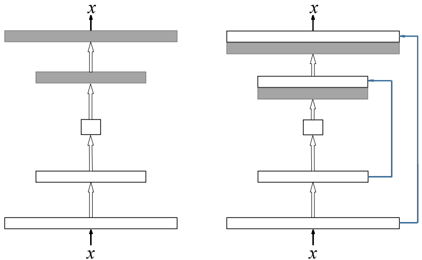


Fig. 2. A comparison between Auto-Encoder and U-Net.

C. Image-to-Image Translation

Image-to-Image translation, which is a widely investigated problem in image processing and computer vision, tries to "translate" images from one domain to another corresponding domain. For examples, translating RGB image into grayscale image, and translating nighttime photos into daytime photos *etc*. The first light of image-to-image translation [11] hired a non-parametric model to implement translation on a paired dataset. In recent research, as the deep learning technology emerges, parametric models have made impressive progress in computer vision. By using CNN, a semantic segmentation method, called SegNet, was proposed to translate original images into semantic segmented images [12]. The "Domain Transfer Network" (DTN) in [13] defined an automatic image-to-image translation framework. DTN is a universal translation model covering many common domains, such as mapping photographs to edges, segments, or semantic labels, and mapping labels and sketch inputs to realistic images.

III. Related Work

The most relevant works are summarized along two directions: *privacy protection in auto-driving* and *application of GAN for privacy protection.*

A. Privacy Protection in Auto-Driving

In the traditional Vehicular Ad hoc Networks (VANETs), a number of methods have been proposed to protect privacy for the vehicles and drivers; especially, location and trajectory privacy are the major focus as (i) most of the vehicular applications are based on location information and (ii) the location information is tightly related to driving safety. In [14], a Social-based PRivacy-preserving packet forwardING (SPRING) protocol was designed based on symmetric cryptography and public key infrastructure. An efficient Social spot-based Packet Forwarding (SPF) protocol is proposed by [15], where the social spots are referred to as the locations in a city environment that many vehicles often visit. By using differential privacy, a spatial division based method was developed in [16] to protect location and trajectory privacy. In [17], the authors presented an efficient packet forwarding protocol, named Social-Tier-Assisted Packet (STAP), for vehicular networks. Particularly, STAP is effective not only in packet dissemination, but also in protection of location privacy of receiver. Notably, for location privacy preservation, almost all of the current works focused on location-based services (LBS), ignoring the leakage of side-channel information in location-independent services. Therefore, these works cannot prevent location-inference attack towards side-channel information.

On the other hand, in the autonomous vehicles, camera data is an indispensable part to help monitor road conditions (*e.g.,* recognizing pedestrians) and guide driving behaviors (*e.g.*, stopping and braking). Although these road conditions and driving behaviors are not determined by locations (*e.g.* in both New York and San Francisco, the Stop sign has the same meaning.), the location of a vehicle can be easily identified through recognizing camera images. However, to the best of our knowledge, no work has been proposed to address the issue how to resist location-inference attack for camera data in auto-driving. Such a blank will be filled by this paper.

B. Application of GAN for Privacy Protection

By exploring the "adversarial" property of GAN, the generator G and the discriminator D can be modeled as a defender and an attacker, respectively. The training process reflects interactions between the defender and the attacker in a zero-sum game, and terminates when a Nash equilibrium is reached such that the defender can win the game.

In [18], [19], generative full body and face de-identification methods were respectively proposed to avoid the recognition of human ID or other biometrics identifiers while preserving data utility and naturalness. Also, GAN-based visual secrets protection methods were introduced by [20], [21], in which the authors used GAN as obfuscation to decrease the probability of successfully detecting secret pixels. To protect text privacy, a GAN-based privacy-preserving method was developed to prevent attackers from inferring age and sex of text author as well as to remain most utility for NLP [22]. The prior work [23] designed a VGAN-based image representation learning for privacy-preserving facial expression recognition, which can protect human ID and maintain expression recognition accuracy. In [24], a method is proposed using adversarial regularization to protect the membership privacy of the training dataset. Additionally, GAN is combined with differential privacy to generate a private dataset to keep enough utility while preserving user privacy by adding designated noise upon training parameters [25]–[27]. In addition to protection, a distributed GAN model was used to recover individual victim's private data even though the data of victim is protected by using distributed differential privacy [28].

For the privacy of image data, the existing GAN-based methods mainly focused on small objects, *e.g.*, face and number. These small objects are easy to be detected and modified because they hold fixed features, and their modifications do not destroy the entire context structure of the images. But, this situation changes when it involves the street view images. Since the street view images have more complex context structures containing a variety of objects, the aforementioned GAN-based methods cannot be applied directly to protect the vehicular camera data. More importantly, the main challenge of our work is that in the camera images, the location-relevant information should be protected to avoid inference attack while preserving data utility.

IV. Methodology of ADGAN

To preserve location privacy while maintaining the utility of the vehicular camera data, we propose an innovative mechanism termed Auto-Driving GAN (ADGAN).

A. Framework & Problem Formulation

As shown in Fig. 3, our ADGAN contains a generator denoted by G and two discriminators respectively denoted by D_1 and D_2.

The generator G is built based on "U-Net" [29] structure. Let I and I' be the set of raw image from real camera data and the set of synthesized image, respectively. For each real camera image $x \in I$, we aim to train G to produce $x' = G(x)$. Specifically, for each captured camera image, we feed it into G and then perform the pixel-to-pixel transformation to output a corresponding synthesized image, which is what a generator does in the conditional GAN. The multi-fold objectives of G include: (i) generate synthesized image as realistic as possible; (ii) maintain the recognition accuracy of non-sensitive information; and (iii) reduce the recognition accuracy of sensitive information.

With the inputs coming from I and I', the goal of D_1 and D_2 is to judge whether their inputs are the real images or the synthesized images. Instead of using one discriminator as the traditional GAN does, we deploy two discriminators in ADGAN. Such a multi-discriminator setting stems from the following consideration: (i) a single discriminator with the fixed receptive field only reads a certain part of an image and thus is easily fooled by the generator; and (ii) combining two discriminators can enhance the ability to distinguish image, which provides privacy protection guarantee when facing a more powerful attacker. To achieve good performance of image recognition with this multi-discriminator setting, D_1 has a small receptive field to perceive the details of small part in image, while D_2 has a large receptive field to obtain a relatively global view of the whole image structure.

By integrating the generator and the two discriminators, we have the following loss function $\mathcal{L}_{ADGAN}$:

$$\begin{aligned}\mathcal{L}_{ADGAN}(G,D) =& \mathcal{L}_{cGAN}(G,D) + \lambda_1 \mathcal{L}_{sim}(G) \\ &+ \lambda_2 \mathcal{L}_{pri}(G),\end{aligned} \tag{4}$$

where $\mathcal{L}_{cGAN}$ is the loss function of the two discriminators, $\mathcal{L}_{sim}(G)$ is the similarity loss indicating the similarity between x and x', $\mathcal{L}_{pri}(G)$ is the privacy loss implying the distance of predefined sensitive objects between all real and synthesized data, and λ_1 and λ_2 are system parameters. Accordingly, when the training process terminates, we can obtain the optimal result G^* as

$$\begin{aligned}G^* = \arg\min_G \max_D [&\mathcal{L}_{cGAN}(G,D) + \lambda_1 \mathcal{L}_{sim}(G) \\ &+ \lambda_2 \mathcal{L}_{pri}(G)].\end{aligned} \tag{5}$$

In Eq. (5), "min-max" means G is expected to beat D_1 and D_2 even though the capability of D_1 and D_2 is maximum. After getting G^*, the auto-driving vehicles could use $x' = G^*(x)$ for their own purposes without leaking location privacy.

B. Image Synthesis

In ADGAN, the generator and discriminators are utilized to fulfill the task of image synthesis, *i.e.*, obtaining $x' = G(x)$. The design of G, D_1 and D_2 are described as follows.

Generator. Image-to-image translation is basically a function that takes an image of a certain domain as input and outputs the image of another domain pixel by pixel. To keep the desired similarity and recognition, the output should be close to the original input. For the problem we consider, except for the sensitive objects (*e.g.*, the background buildings), the other information of synthesized images should be similar to those of real images. This requires the output and input are different in appearance at background pixel location, but

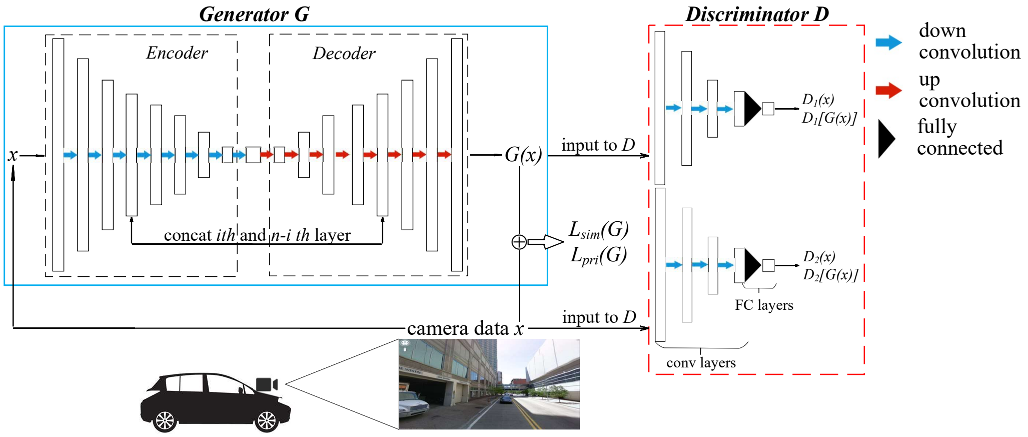


Fig. 3. System framework of ADGAN, which contains a "U-Net" based generator G and a multi-discriminator structure $\mathbf{D} = \{D_1, D_2\}$. The notation $\oplus$ between $G(x)$ and x represents our adopted measurements for optimizing $\mathcal{L}_{sim}(G)$ and $\mathcal{L}_{pri}(G)$ rather than XOR operation.

have exactly same underlying structure. For this purpose, the encoder-decoder based "U-Net", which can reconstruct core stuff in images, is adopted.

The generator contains an encoder to compress input images as well as a decoder to recover output images from middle hidden tensor. The hidden tensor layer, like a bottleneck in neural networks, saves common underlying structure shared by input and output. Also, there exist many effective skip links between i-th layer of the encoder and $(n-i)$-th layer of the decoder, performing copy and corp operations to compel the output images to preserve more boundary in the input.

During the process of image synthesis, G has three objectives: (i) it should be able to mimic the real distribution of the input image set, so that the generated images are indistinguishable from real images; (ii) the generated images should be as realistic as possible and look very similar to the input images; and (iii) to protect the side-channel information of privacy, this information in the generated images should be dissimilar to that in the input images and unrecognizable by either human or detection mechanism.

The objective (i), also called "adversarial loss", can be achieved by adjusting network weight to minimize the loss function $\mathcal{L}_{cGAN}$ in Eq. (6):

$$\mathcal{L}_{cGAN} = \mathbb{E}_{x\sim I}[logD(x)] + \mathbb{E}_{x'\sim I'}[1 - logD(x')]. \qquad (6)$$

The remaining two objectives, including "similarity loss" (denoted by $\mathcal{L}_{sim}(G)$) and "privacy loss" (denoted by $\mathcal{L}_{pri}(G)$), will be addressed in Section IV-C.

Discriminator. The discriminator in ADGAN performs a classification task, *i.e.*, differentiating whether its input images are from real captured camera dataset I or the generated dataset I'. In the scenario of auto-driving, the captured camera images are usually complex street view scene, which has not been well investigated in the previous GAN-based privacy protection methods. As mentioned in [30], one fixed discriminator has difficulty in differentiating real or fake on high-resolution images, because it is hard for a discriminator to get a comprehensive understanding about every part in the high-resolution images.

Thus, to differentiate these complex high-resolution real images and generated images, more than one discriminator is needed to deal with images with different receptive fields in an effective manner. In ADGAN, we creatively exploit a multi-discriminator setting $\mathbf{D} = \{D_1, D_2\}$, which is the most significant difference from the existing GAN. The two discriminators in $\mathbf{D} = \{D_1, D_2\}$ perform the binary classification with two separate convolutional neural networks. For a generative task, on one hand, we need to make the synthesized image under the target distribution at a global aspect; and on the other hand, we intend to make the generated images more realistic in details. More specifically, to have delicate particulars in the generated images, D_1 is a CNN discriminator with a smaller receptive field to capture details; and to have a better global view in the generated images, D_2 is a CNN discriminator with a quite large receptive field to scan entire input. The output of D_1 and D_2 is a scalar representing a probability of real data, and the summation of both output is the loss function of discriminator $\mathbf{D} = \{D_1, D_2\}$. Accordingly, we need to maximize the following "extended adversarial loss":

$$\mathcal{L}_{cGAN} = \sum_{D_1, D_2 \in \mathbf{D}} [\mathbb{E}_{x\sim I}[logD_i(x)] + \mathbb{E}_{x'\sim I'}[1 - logD_i(x')]]. \qquad (7)$$

C. Utility & Privacy

In this subsection, the "similarity loss" and the "privacy loss" are introduced. These two types of loss are used to improve G in GAN so that the synthesized images are similar to the input images while preserving visual location privacy in images.

Utility. Our multi-discriminator in $\mathcal{L}_{cGAN}$ can ensure the generated images are subject to the distribution of real images, but can not guarantee that the generated image x' and input image x are similar in image space when images are complex

TABLE I
NETWORK ARCHITECTURE

Layer	Encoder	Decoder	Discriminator
1	$5 \times 5 \times 64$ conv, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$D_1/D_2 : 2 \times 2 \times 64/5 \times 5 \times 64$ conv, Leaky ReLU
2	$5 \times 5 \times 128$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$D_1/D_2 : 2 \times 2 \times 128/5 \times 5 \times 128$ conv, B_N, Leaky ReLU
3	$5 \times 5 \times 256$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$D_1/D_2 : 2 \times 2 \times 256/5 \times 5 \times 256$ conv, B_N, Leaky ReLU
4	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$D_1/D_2 : 2 \times 2 \times 512/5 \times 5 \times 512$ conv, B_N, Leaky ReLU
5	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 256$ deconv, B_N, ReLU	Fully Connected, Sigmoid
6	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 128$ deconv, B_N, ReLU	
7	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 64$ deconv, B_N, ReLU	
8	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 3$ deconv, tanh	

due to the model collapse property of GAN. This is the weakness that some images generated by GAN-based models may lost perceptual accuracy. Therefore, the similarity loss, $\mathcal{L}_{sim}(G)$, can work as a kind of constraint to restrict perceptual loss. There exist a few methods to adjust the generator, such as mean square error (MES) and the related quantity of peak signal-to-noise ratio (PSNR), but these methods are not suitable for human perceived quality. So we use the Structure Similarity Index Measurement (SSIM) to access high perceptual accuracy.

SSIM contains three parts: luminance similarity $l(x, x')$, contrast similarity $c(x, x')$ and structural similarity $s(x, x')$:

$$SSIM(x, x') = l(x, x')^{\alpha} \cdot c(x, x')^{\beta} \cdot s(x, x')^{\gamma}. \tag{8}$$

More details about SSIM could be referred to [31]. The output of SSIM is a numerical number in range $[0, 1]$ and represents the similarity between x and x', where 0 means totally different and 1 means exactly same. In order to minimize the entire loss function $\mathcal{L}_{ADGAN}$, $\mathcal{L}_{sim}(G)$ is defined to be

$$\mathcal{L}_{sim}(G) = \mathbb{E}_{x \sim I, x' \sim I'}[1 - SSIM(x, x')]. \tag{9}$$

Privacy. For privacy protection, we aim to change private information in the input images to irrelevant information in the generated images. To achieve this, we need to locate private information in the real images first. In semi-supervised learning process, we use the paired data $\{image, label\}$ in the dataset to train a FCN8s model [32], which can tell us correct classification of each pixel in images. Then, we can get the label of each pixel for more new coming images through this specific pre-trained model as long as the images are from the same data source, *e.g.* street view. Thus, we can obtain the location of private information in the images. In ADGAN, we adopt L_1 distance to measure the difference of selected information between the input and generated images. With respect to privacy protection, a larger L_1 distance indicates more privacy is preserved, and a smaller L_1 distance means more privacy is exposed. Therefore, the loss function, $\mathcal{L}_{pri}(G)$, can be formulated as:

$$\mathcal{L}_{pri}(G) = \frac{1}{\mathbb{E}_{x \sim I, x' \sim I'}[\| x^{pri} - x'^{pri} \|_1]}, \tag{10}$$

where x^{pri} represents private information defined in the real image x, and x'^{pri} is the corresponding private information in the generated image x'.

Remarks. In this paper, we define background buildings as the private/sensitive objects, because background building is a kind of important side-channel information for location-inference attack. Nevertheless, we can protect any kind of information in images by locating and modifying them according to the requirements of privacy protection.

In conclusion, the loss function of entire ADGAN, $\mathcal{L}_{ADGAN}$, can be expressed as the following explicit function:

$$\begin{aligned} &\mathcal{L}_{ADGAN} = \\ &\sum_{D_1, D_2 \in \mathbf{D}} [\mathbb{E}_{x \sim I}[logD_i(x)] + \mathbb{E}_{x' \sim I'}[1 - logD_i(x')]] \\ &+ \lambda_1 \mathbb{E}_{x \sim I, x' \sim I'}[1 - SSIM(x, x')] \\ &+ \lambda_2 \frac{1}{\mathbb{E}_{x \sim I, x' \sim I'}[\| x^{pri} - x'^{pri} \|_1]}, \end{aligned} \tag{11}$$

where λ_1 and λ_2 are hyper-parameters that indicate the weight of last two terms and also act as regularization terms when their values are reduced to the scale of $\mathcal{L}_{ADGAN}$ with proper adjudication.

D. Network Architecture

Our generator and discriminators are constructed based on the architecture in [33]. Each layer of the networks contains convolution/deconvolution, Batch Normalization, and a Leak ReLU/ReLU activation function. Particularly, the Batch Normalization is employed to normalize the input to zero mean and unit standard deviation. In the encoder of generator, there are 8 fully convolutional layers with filter size 5×5, each of which has a Leaky ReLU and Batch Normalization except the first layer. The structure of decoder is the opposite of that of the encoder except for ReLU and the $tanh$ activation of the 8th layer. For the two discriminators, the receptive field of D_1 is 5×5, the receptive field of D_2 is 61×61, and both of two are traditional CNNs holding 4 convolutional layers and end with fully connected layer. The details of the network architecture are presented in TABLE I.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our ADGAN model via intensive real-data experiments.

A. Experiment Setting

1) Datasets: To explore the effectiveness and robustness of our ADGAN under different scenarios, two different vehicular camera datasets are adopted. **(i) Cityscapes Dataset [34].** It contains 2975 street view images from 18 cities for training and 500 street view images from 3 cities for evaluating. Besides, in Cityscapes, the labels for supervised learning are provided. **(ii) Google Street View [35].** There are 62,058 high-quality street view images and 10,343 related placemark collected from 4 cities in USA. Every 6 of the images share a placemark that works as an identifier to perform geo-localization detection.

2) Adversary Model: Suppose an attacker can get raw camera data from other vehicles or open source *e.g.*, Google Map API as the training reference dataset. Then the attacker acquires camera data from target victim and performs feature extraction and object detection to infer the victim's location using Multi-KNN [35]. To avoid such attack, the most important thing is to prevent feature extraction from location-relevant side information. In our experiments, background buildings are concealed as the sensitive location-relevant information while other useful information is reserved in the generated images.

3) Comparisons: In the experiments, a comprehensive comparison is set up between our ADGAN and other three baseline models.

(i) pix2pix [30]**.** It uses a pure supervised method on labeled pair data. Since it does not consider privacy protection, it can be adopted as the benchmark for performance comparison.

(ii) pix2pix+pri. It is a combination of pix2pix and our privacy-preserving mechanism. The major difference is loss function and structure of discriminator. The loss function of pix2pix+pri is $\mathcal{L}(G,D) = \mathcal{L}_{cGAN}(G,D) + \lambda_1 \mathbb{E}[||y - G(x)||_1]$ $+\lambda_2 \mathcal{L}_{pri}(G)$, where $\mathcal{L}_{pri}(G)$ is defined in our ADGAN.

(iii) UNIT+pri. UNIT [36] is an unsupervised image-to-image translation applied on street view, which does not need any labeled data. For the purpose of comparison, we modify the part of GAN in UNIT to a privacy-preserving version "UNIT+privacy". The modified loss function is $\mathcal{L}_{GAN_1}(G_1, D_1) = \lambda_0 \mathbb{E}[logD_1(x_1)] + \lambda_0 \mathbb{E}[1 - log(D_1(G_1(z))] - \lambda_1 \mathbb{E}[\| x_1 - G_1(z)]$, where z is the shared latent vector between the input and output domains.

4) Training Method: In all experiments, we set the epoch is 200. Specially, in our ADGAN model, we set $\lambda_1 = 200$ and $\lambda_2 = 10$ to balance the entire loss function. In ADGAN, each layer with a Leaky ReLU activation has a drop-out rate of 50% and a slope of 0.2.

B. Analysis of Utility and Privacy

FCN-scores, which are used in image-to-image translation methods [30], [37], are employed to quantify *privacy preservation* and *recognition utility* in the synthesized images. Particularly, in FCN, pixel accuracy and interaction over union (IoU) are two key indices to indicate the correct rate of object detection in images. We first perform semantic segmentation on the generated images, and then compare the predicted segmentation of generated images and the ground truth of real images. It is expected that for the predefined sensitive information (*i.e.*, background buildings in this paper), the pixel accuracy and IoU of should be lower than ground truth value, which implies that an attacker can not recognize the object on certain pixels and thus are not able to detect real location from the generated images. On the other hand, for the non-sensitive information, the predicted results should be the same as the ground truth with a high probability.

TABLE II
FCN-SCORES COMPARISON OF 4 MODELS ON CITYSCAPES

Model	**pix2pix**	**pix2pix+pri**	**UNIT+pri**	**ADGAN**
global accuracy	85.04%	50.92%	47.67%	**76.52**%
sensitive accuracy	84.91%	46.80%	**35.33**%	64.65%
non-sens accuracy	84.93%	59.56%	55.54%	**81.91**%
global IoU	36.45%	10.16%	8.46%	**22.36**%
sensitive IoU	36.31%	7.45%	**6.34**%	11.75%
non-sens IoU	36.65%	14.21%	10.53%	**29.93**%

The results of pixel accuracy and IoU of the four models are compared in TABLE II and TABLE III, where "global accuracy" is the average accuracy of each pixel in an image being classified into correct class for all objects (*i.e.*, true positive), "sensitive accuracy" represents the average accuracy of each pixel being classified into correct class for pre-defined private objects, and "non-sens accuracy" is the average accuracy of each pixel being classified into correct class for non-sensitive objects. For "global IoU", "sensitive IoU", and "non-sens IoU", their definitions are similar to those of pixel accuracy for all, sensitive and non-sensitive objects, respectively. **The pixel accuracy and IoU for all and non-sensitive objects are used to estimate recognition utility, and those for sensitive objects are used to evaluate privacy protection.**

From TABLE II that shows the results on Cityscapes, we can obtain the following observations.

(i) Since pix2pix does not consider privacy protection, it gets the highest recognition utility. Meanwhile, an attacker can recognize the sensitive objects from the synthesized images with the highest accuracy, resulting in serious privacy leakage.

(ii) For privacy preservation, all of pix2pix+pri, UNIT+pri, and ADGAN perform better than pix2pix. But, the recognition utilities of pix2pix+pri, UNIT+pri, and ADGAN are reduced, because hiding sensitive objects has negative impact on recognizing non-sensitive objects. Among these three models, our ADGAN achieves the best recognition utility. What's more, the recognition utility of ADGAN is comparable to that of pix2pix. In TABLE II, the pixel accuracy of ADGAN and pix2pix for non-sensitive objects is 81.91% and 84.93%, respectively; and IoU of ADGAN and pix2pix for non-sensitive objects are 29.93% and 36.65%, respectively.

(iii) Although both pix2pix+pri and UNIT+pri can preserve more private information than our ADGAN does, they nearly lose all useful information and have worse recognition utility. As shown in TABLE II, for non-sensitive objects, the pixel accuracy of pix2pix+pri and UNIT+pri are dramatically reduced to 59.56% and 55.54%, respectively, while our ADGAN

TABLE III
FCN-SCORES COMPARISON OF 4 MODELS ON GOOGLE STREET VIEW

Model	**pix2pix**	**pix2pix+pri**	**UNIT+pri**	**ADGAN**
global accuracy	82.75%	59.87%	31.70%	**72.31%**
sensitive-accuracy	83.07%	58.25%	**25.05%**	62.82%
non-sens accuracy	82.73%	66.69%	34.69%	**78.97%**
global IoU	29.65%	11.01%	5.52%	**18.37%**
sensitive IoU	30.13%	8.71%	**3.27%**	13.40%
non-sens IoU	29.74%	13.81%	6.54%	**21.89%**

TABLE IV
FCN-SCORES COMPARISON FOR DIFFERENT DISCRIMINATOR

Metrics	single-D	**multi-D**
global accuracy	75.19%	**76.52%**
sensitive-accuracy	64.03%	**64.65%**
non-sens accuracy	80.11%	**81.91%**
global IoU	19.87%	**22.36%**
sensitive IoU	10.46%	**11.75%**
non-sens IoU	26.75%	**29.93%**

keeps this accuracy at 81.91%. Such low pixel accuracy of pix2pix+pri and UNIT+pri is very likely to cause the synthesized images to be useless in real applications.

The reasons for the performance of pix2pix+pri and UNIT+pri are analyzed below. Our ADGAN has a framework similar to pix2pix+pri, where the most significant difference lies in the discriminator structure and SSIM loss function. In our ADGAN, the multi-discriminator deployment greatly contributes to the improvement of recognition utility. UNIT+pri works as a symmetric cycGAN structure, which trains the final model by minimizing the distance between hidden vector of two domains. Even we grab the private information in input images, it is still hard to control what the hidden vector is after being processed by the encoder. So, the generated images of UNIT+pri has a bad recognition utility.

We also implement the four models on Google Street View dataset. Recall that pix2pix is a supervised method, UNIT is an unsupervised method, and our ADGAN utilizes the semi-supervised idea to handle data augmentation. We first grab a small part of images with labels to train a fully convolutional image segmentation model, and then use the pre-trained model to generate large amount of labeled data as an indicator for the targeted private information. The results on Google Street View dataset are given via TABLE III, of which the observations are the same as those of TABLE II.

Summary of Analysis. For the synthesized images, recognition utility and privacy protection conflict with each other. Enhancing recognition utility makes the images suitable for real applications (*e.g.*, object detection and data mining) but causes the images to suffer from severe privacy threats. On the other hand, the improvement of privacy protection is achieved at the cost of recognition utility, leading to the useless synthesized images. Thus, balancing the tradeoff between recognition utility and privacy protection is the challenging issue for protecting camera data. Notably, the above experiment results well validate that our ADGAN can provide an effective tradeoff.

C. Analysis of Discriminator

Different from the previous works, we build a multi-discriminator structure to constrain the generated images by using different receptive fields. With this novel structure, the discriminators together can provide more effective feedback to the generator in ADGAN, because they are able to have better understanding on what is real or fake. We evaluate the advantage of multi-discriminator over single-discriminator by fixing the generator and loss function structure. The FCN-scores in TABLE IV clearly demonstrate that our multi-discriminator performs better than single-discriminator in terms of pixel accuracy and IoU.

D. Perception Comparison

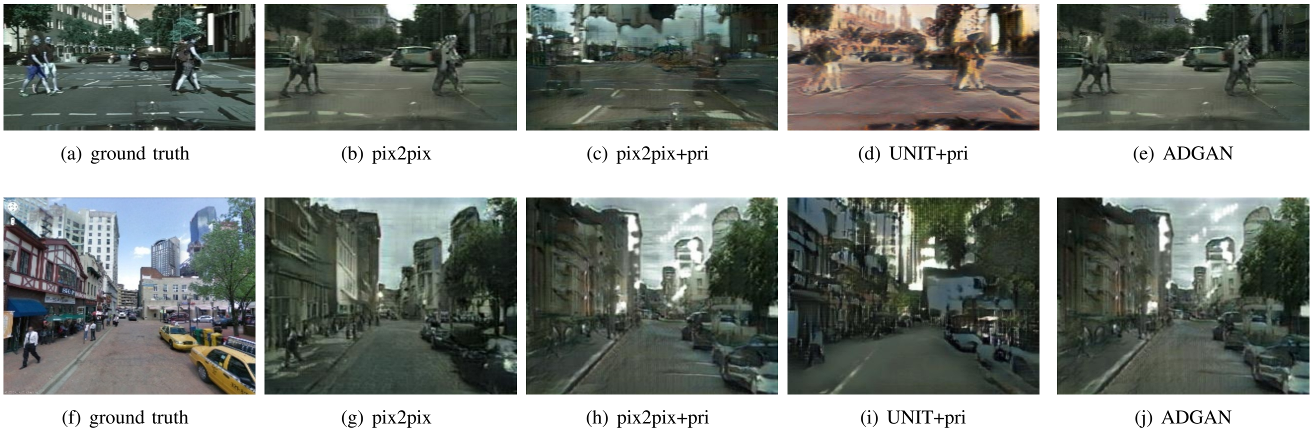
TABLE V
AMAZON MECHANICAL TURK (AMT) TEST "REAL *v.s.* FAKE" TEST ON CITYSCAPES AND GOOGLE STREET VIEW

Model	**Cityscapes**	**Google Street View**
	% Turkers labeled $real$	% Turkers labeled $real$
pix2pix	18.9%±2.5%	11.2%±1.3%
pix2pix+pri	9.9%±0.4%	6.1%±0.5%
UNIT+pri	2.6%±1.1%	2.1%±1.0%
ADGAN	**10.1%±0.3%**	**8.7%±0.6%**

To evaluate the perceptual effectiveness of ADGAN, we perform the Amazon Mechanical Turk (AMT) test, image quality comparison and semantic segmentation comparison.

AMT Evaluation. For the AMT experiments, the protocol in [38] states: Turkers were presented with a series of trials that need to be labeled with "real" if facing raw data and "fake" if it is generated data by image-to-image model. During each trial, each image appears for 1 second, after which the images disappear and Turkers are given limited time (random time from 1/8 second to 8 second [39]) to respond which one was fake. The first 10 images of each session are practice and then Turkers are given feedback. No feedback is provided on the 40 trials of the main experiment. Each session tests just one method at one time, and Turkers are not allowed to complete more than one session. There are 50 Turkers to evaluate each model. At last, we can get a percentage for each model which represents how much percentages of Turkers is fooled to give a wrong answer. TABLE V shows the AMT test results. It can be seen that our ADGAN can fool 10.1% of participants on Cityscapes and 8.7% on Google Street View, which is the best AMT recognition performance among three models that consider privacy. Compared with pix2pix that does not consider privacy, the degradation of AMT recognition performance of ADGAN is not too much.

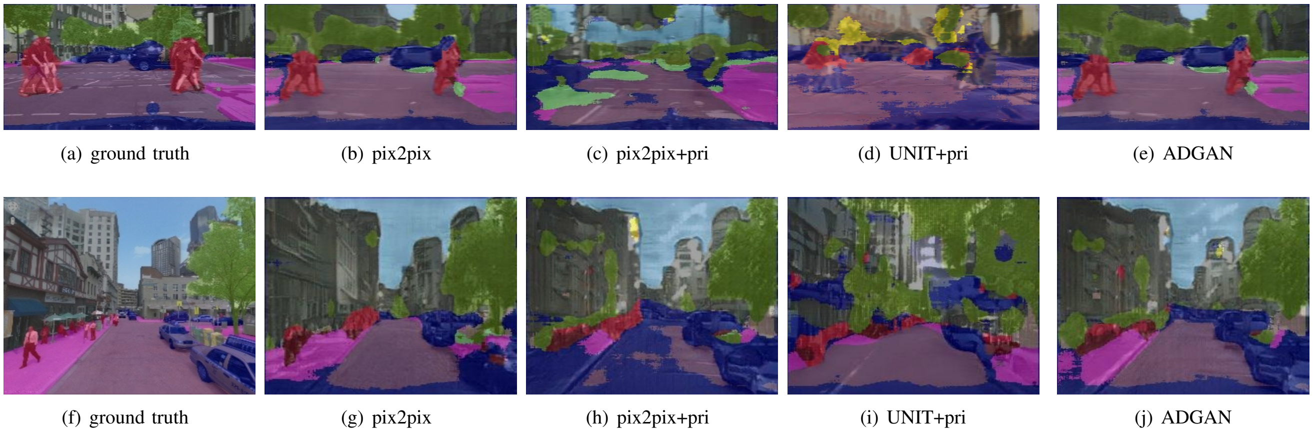
Image Quality Comparison. In Fig. 4, we show the random sampled images from our model $G(x)$ and the three baselines. From these images, we observe that: (i) the output of ADGAN are very similar to the original input images; (ii) our results are closer to those of pix2pix with clear boundary and details;



(a) ground truth (b) pix2pix (c) pix2pix+pri (d) UNIT+pri (e) ADGAN

(f) ground truth (g) pix2pix (h) pix2pix+pri (i) UNIT+pri (j) ADGAN

Fig. 4. Visual quality comparison of generated images for Cityscapes dataset and Google Street View dataset. The first row is Cityscapes, and second row is Google Street View. Figures from left to right column is ground truth of input, pix2pix result, pix2pix+pri result, UNIT+pri result and ADGAN result respectively.



(a) ground truth (b) pix2pix (c) pix2pix+pri (d) UNIT+pri (e) ADGAN

(f) ground truth (g) pix2pix (h) pix2pix+pri (i) UNIT+pri (j) ADGAN

Fig. 5. Semantic segmentation comparison of generated images for Cityscapes dataset and Google Street View dataset. The first row is Cityscapes, and second row is Google Street View. Figures from left to right column is ground truth of input, pix2pix result, pix2pix+pri result, UNIT+pri result and ADGAN result respectively.

and (iii) the quality of images from ADGAN are better than that from both pix2pix+pri and UNIT+pri.

Semantic Segmentation. Semantic segmentation is an important research topic in computer vision and auto-driving. In the prior work, conditional GANs have been proved workable in semantic segmentation [40]. To explore what extent our ADGAN can achieve after adding privacy consideration, we train a FCN8s model to segment sample images generated from the four models. The segmentation results in Fig. 5 show that our ADGAN outperforms the other three models in a qualitative view. As the pre-defined sensitive objects, the background buildings are hard to be recognized as buildings no matter in computer vision or even in human vision, which is consistent with the observations of TABLE II and TABLE III.

VI. Conclusion and Future Work

In this paper, we design a novel method called ADGAN, which seamlessly integrates GAN and image-to-image translation, to generate privacy-preserving camera images for protecting location privacy in auto-driving. We use two real datasets to evaluate the performance of ADGAN, and comprehensive comparisons between ADGAN and the baselines well confirm the advantages of our ADGAN. With limited experimental environment and hardware, in this paper, we implement our model on 256×512 and 400×512 scale. In our future work, we will further investigate the performance improvement of our model for higher resolution camera data.

Acknowledgment

This work is partly supported by the National Science Foundation (NSF) under grant NOs. 1704287, 1912753 and 1829674, and the National Natural Science Foundation of China under Grant NO. 61872105.

References

[1] Y. Tian, K. Pei, S. Jana, and B. Ray, "Deeptest: Automated testing of deep-neural-network-driven autonomous cars," in *Proceedings of the 40th international conference on software engineering*. ACM, 2018, pp. 303–314.

[2] A. C. MADRIGAL, "Inside waymo's secret world for training self-driving cars," The Atlantic, Tech. Rep., 2017. [Online]. Available: https://www.theatlantic.com/technology/archive/2017/08/inside-waymos-secret-testing-and-simulation-facilities/537648/

[3] R. Aoki, "Autonomous vehicle disengagement reports," California DMV, Tech. Rep., 2016. [Online]. Available: https://www.dmv.ca.gov/portal/dmv/detail/vr/autonomous/disengagement/report/2016

[4] E. Guizzo, "How google's self-driving car works," IEEE Spectrum, Tech. Rep., 2011.

[5] J. Petit, "Self-driving and connected cars: Fooling sensors and tracking drivers," University of Twente, Tech. Rep., 2015.

[6] G. Schindler, M. Brown, and R. Szeliski, "City-scale location recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. Citeseer, 2007, pp. 1–7.

[7] A. Feryanto and I. Supriana, "Location recognition using detected objects in an image," in *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics*. IEEE, 2011, pp. 1–4.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[9] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 37–49.

[10] A. Boesen Lindbo Larsen, S. Kaae Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *arXiv preprint arXiv:1512.09300*, 2015.

[11] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 327–340.

[12] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

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

[14] L. Rongxing, L. Xiaodong, and S. Xuemin, "Spring: A social-based privacy-preserving packet forwarding protocol for vehicular delay tolerant networks," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[15] R. Lu, X. Lin, X. Liang, and X. Shen, "Sacrificing the plum tree for the peach tree: A socialspot tactic for protecting receiver-location privacy in vanet," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*. IEEE, 2010, pp. 1–5.

[16] Q. Han, Z. Xiong, and K. Zhang, "Research on trajectory data releasing method via differential privacy based on spatial partition," *Security and Communication Networks*, vol. 2018, 2018.

[17] X. Lin, R. Lu, X. Liang, and X. Shen, "Stap: A social-tier-assisted packet forwarding protocol for achieving receiver-location privacy preservation in vanets," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 2147–2155.

[18] K. Brkic, I. Sikiric, T. Hrkac, and Z. Kalafatic, "I know that person: Generative full body and face de-identification of people in images," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017, pp. 1319–1328.

[19] Y. Wu, F. Yang, Y. Xu, and H. Ling, "Privacy-protective-gan for privacy preserving face de-identification," *Journal of Computer Science and Technology*, vol. 34, no. 1, pp. 47–60, 2019.

[20] N. Raval, A. Machanavajjhala, and L. P. Cox, "Protecting visual secrets using adversarial nets," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017, pp. 1329–1332.

[21] F. Pittaluga, S. Koppal, and A. Chakrabarti, "Learning privacy preserving encodings through adversarial training," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 791–799.

[22] Y. Li, T. Baldwin, and T. Cohn, "Towards robust and privacy-preserving text representations," *arXiv preprint arXiv:1805.06093*, 2018.

[23] J. Chen, J. Konrad, and P. Ishwar, "Vgan-based image representation learning for privacy-preserving facial expression recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1570–1579.

[24] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 634–646.

[25] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 308–318.

[26] C. Huang, P. Kairouz, X. Chen, L. Sankar, and R. Rajagopal, "Context-aware generative adversarial privacy," *Entropy*, vol. 19, no. 12, p. 656, 2017.

[27] G. Acs, L. Melis, C. Castelluccia, and E. De Cristofaro, "Differentially private mixture of generative neural networks," *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[28] B. Hitaj, G. Ateniese, and F. Pérez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 603–618.

[29] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[30] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[31] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[32] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

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

[34] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[35] A. R. Zamir and M. Shah, "Image geo-localization based on multiplenearest neighbor feature matching usinggeneralized graphs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 8, pp. 1546–1558, 2014.

[36] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.

[37] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[39] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1511–1520.

[40] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," in *NIPS Workshop on Adversarial Training*, 2016.