Fooling Semantic Segmentation in One Step via Manipulating Nuisance Factors

Guangyu Shen[1], Chengzhi Mao[2], Junfeng Yang[2], and Baishakhi Ray[2]

[1] Purdue University, West Lafayette, IN, USA
[2] Columbia University, New York, NY, USA
shen447@purdue.edu,mcz,junfeng,rayb@cs.columbia.edu

Abstract. Due to the inherent robustness of segmentation, traditional methods fail to attack it in a few steps. We demonstrate a simple and effective method that changes the nuisance factors to fool semantic segmentation models in a single step, even if the models have been adversarially trained for defense. By training conditional generative models with a designed adversarial loss, we can generate realistic adversarial attack by changing nuisance factors. Our method does not require neural network back-propagation, which is even faster than fast gradient sign method (FGSM). We validate our approach on the popular Cityscapes and ADE20K datasets, and demonstrate better attack success rate compared to the existed adversarial attacks for semantic segmentation including PGD, Houdini and DAG with over 100x speed-up, which is the first method that can attack semantic segmentation models online.

Keywords: Adversarial Attack, Segmentation, Generative Adversarial Network

1 Introduction

Generating successful adversarial attacks often requires a large number of optimization steps [2]. Past work assumed that the attacker has unlimited time and computational budget. However, this is not always true in practice. For example, if the attacker is trying to fool an autonomous-driving car online under limited resources, the attacker will have little time to optimize the adversarial noise before the car going away. Existing successful attack requires more than 30 seconds on a single NVIDIA TITAN Xp GPU, which is slow given the frame rate [12, 21] for current self-driving cars.

The real-world deep learning models are getting increasingly complex. The complexity of the tasks not only strengthens models' inherently robustness themselves [1, 14], but also slow down the optimization speed for the attacker, preventing successful online adversarial attack. Semantic segmentation is a key task for autonomous cars, which are shown to be inherently robust [1, 3] due to the high dimensional output [14]. Existing methods require hundreds of steps to attack semantic segmentation models [3, 26], which is slow considering the computational cost of back-propagating through the large semantic segmentation models.

A large amount of efforts have been made to defend neural networks against these adversarial attacks, which further rising the need for more attack steps [13, 27, 15]. Robust models are trained such that they are not easily evaded by adversarial examples

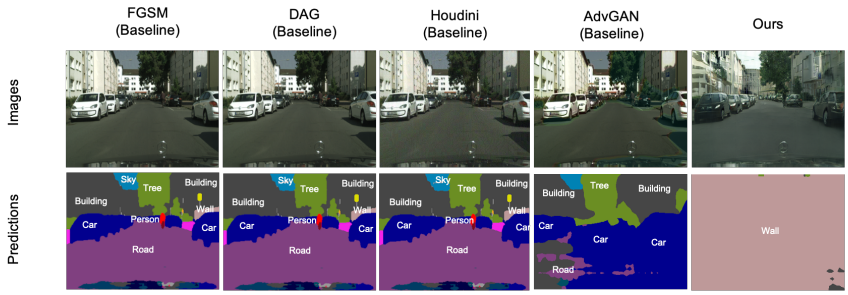

Fig. 1. Our one step attack Vs. existed segmentation attacks under limited time budget (1s).The 1^{st} column shows the FGSM adversarial example and attack result. The 2^{nd} and 3^{rd} columns show the DAG and Houdini generated adversarial examples and corresponding attack results. The 4^{th} column shows advGAN generated attack image and prediction. The 5^{th} column shows our unrestricted adversarial image and result. By changing the nuisance factors (color, illumination, texture) of each object, our attack is successful while the existing methods fail to fool the semantic segmentation model in real time.

[5, 17, 13]. Although these defense methods improve the models' robustness, they are mostly limited to addressing norm-bounded attacks such as PGD [13]. Recently, unrestricted adversarial attacks are proposed [19], where the attack images modify the nuisance factors. The generated images still contain consistent semantic information with the original images from the human perspective. The unrestricted adversarial attacks are powerful to attack existing models given it can also change the texture, lighting, color, etc [22], which can expose more vulnerabilities of a given machine learning model. However, given each query, existing method requires to optimize the latent space with back-propagation for many steps [19].

In this paper, we propose to generate effective and realistic unrestricted attack in a single forward inference step (Figure 1). By training the state-of-the-art conditional generative networks with an adversarial loss, we steer the generator to generate realistic attack images that fool the semantic segmentation model. To ensure the semantic consistency between our generated images and the groundtruth images, we conditional on the true semantic labels during the training phase.

While previous methods focus on the attack success rate, we are the first one to consider adversarial attack in an online fashion under limited time. We believe considering the attack time and computational budget is part of the key for practical adversarial attack in the real world. Empirical results on two real-world datasets, Cityscapes and ADE20K, show that our attack methods is up to 100 times faster than existing attacks for segmentation tasks, while keeping semantic consistent with the original image and improving the attack success rate up to 41%.

2 Related Work

Adversarial Attacks. Adversarial examples are carefully crafted to mislead the models' predictions, while still perceived identical or similar semantics with original images to the human. There are a few studies focused on the adversarial attack on modern semantic segmentation networks. [1] conducted the first systematic analysis about the

effect of multiple adversarial attack methods on different modern semantic segmentation network architectures across two large-scale datasets. [26] proposed a new attack method called Dense Adversary Generation(DAG), which generates a group of adversarial examples for a bunch of state-of-the-art segmentation and detection deep networks. [3] further advanced the restricted attacks using surrogate Houdini loss functions for semantic segmentation to make non-differential IOU loss to be approximate differential, however, due to the expensive computationally, traditional norm-bounded attack methods, such as PGD [13], are more widely used in practise.

Generative Adversarial Network (GAN), popular for image manipulation [28, 11, 7, 30], image-to-image translation [8, 23, 18], etc., was also leveraged to generate adversarial examples. Xiao et al. [24] proposed AdvGAN to generate norm-bounded perturbation for the classification task. Song et al. [19] used GAN to generate *unrestricted adversarial attacks* for image classification. However, none of the methods are effective for online attack setting for segmentation tasks.

3 Method

We introduce our methodology for generating unrestricted adversarial examples for semantic segmentation.We leverage a conditional Generative Adversarial Networks. While existing conditional GAN aims to synthesize realistic images that fool the discriminator, our method also requires to fool the segmentation model. We add an additional loss function to fool both the discriminator and the segmentation model. The rest of the section describes the steps in details.

Conditional Generative Adversarial Networks. A Conditional Generative Adversarial Network [16] consists of a generator G and a discriminator D and they are both conditioned on auxiliary information y. Combining random noise z and extra information y as input, G is able to map it to a realistic image. The discriminator aims to distinguish the real images and synthetic images from the Generator. G and D correspond to a minimax two-player game and can be formalized as $\min_G \max_D V(G, D) = \mathbb{E}_{\mathrm{x}\sim P_{data}(x)}[\log D(x|y)] + \mathbb{E}_{\mathrm{z}\sim P_z(z)}[\log(1 - D(G(x|y, z)))]$. We use the ground-truth prediction as the condition y to ensure the semantic consistency between the generated image and the ground-truth image.

Variational Inference. The nuisance factors of the generated image are encoded in the latent noise z. We leverage a Variational Auto-Encoder [9] to learn the posterior of the hidden noise z that generate successful adversarial attack, where an encoder E encodes the input image into a noise space which generates image with style that can attack successfully.

Unrestricted Adversarial Loss. In practice, we use the SPADE architecture [18], where the SPADE generator is trained to mislead the prediction of target segmentation network. We introduce the target segmentation network into the training phase and aim to maximize the loss of the segmentation model while keeping the quality and semantic meaning of the synthetic images. We denote the target segmentation network by S, SPADE generator by G, input semantic label by y, and the hidden state vector by z. We select Dice Loss [20] as the objective function f. We define the untargeted version of Unrestricted Adversarial Loss as follows:

$$L_{ATK} = -\mathbb{E}_{\mathrm{z}\sim P_z(z)} \log f(S(G(x|y, z)), y) \tag{1}$$

Fig. 2. Visual Results on Cityscapes: The 1^{st} row is the real images in Cityscapes validation set, which is never seen during our training phase. The 2^{nd} row is our unrestricted adversarial examples that only change nuisance factors. The 3^rd row is the corresponding segmentation results. Given a new street scene, our method can fool the segmentation model online.

The complete objective function of ours then can be written as:

$$\min_{G,E}((\max_{D_1,D_2,D_3}\sum_{k=1,2,3}\mathcal{L}_{GAN}(G,D_k))+\lambda_0\sum_{k=1,2,3}\mathcal{L}_{FM}(G,D_k))+\lambda_1\mathcal{L}_{VGG}(G)$$
$$+\lambda_2\mathcal{L}_{KLD}(E)+\lambda_3\mathcal{L}_{ATK}(S,G)) \quad (2)$$

We follow the definition of feature matching loss $\mathcal{L}_{FM}$ and perceptual loss $\mathcal{L}_{VGG}$ in [23]. The feature matching loss, $\mathcal{L}_{FM}(G,D_k) = \mathbb{E}_{(y,x)}\sum_{i=1}^{T}\frac{1}{N_i}[||D_k^{(i)}(y,x) - D_k^{(i)}(s,G(y))||_1]$, is able to stabilizes the GAN training by minimizing the distance between features of synthesis and real images from multiple layers of the discriminator. The VGG perceptual loss ($\mathcal{L}_{VGG}(G) = \sum_{(i=1)}^{N}\frac{1}{M_i}[||F^{(i)}(x) - F^{(i)}(G(y))||_1]$) plays a similar role as $\mathcal{L}_{FM}$ by introducing a pretrained VGG network.

Why it is fast. We train the above generator on the training set offline. During the inference time, we forward the query image to the encoder E and generate the latent noise which can produce attack, and use the generator to decode that image into the real-world attack image. The whole process only contains the forward computation, without any backward optimization. Our method can be further accelerated using tools such as TensorRT for fast forward.

4 Experimental Set-up

Datasets. We evaluate our method on two large image segmentation datasets: Citsycapes [4] and ADE20K [31].

Baseline Models. We compare our attacks with traditional *norm-bounded* attacks in two settings: (i) real images with perturbation and (ii) GAN-generated clean images with perturbation. For (ii), we generate clean images with vanilla SPADE and then add norm-bounded perturbation over the synthetic images. For a better comparison, we choose the same segmentation networks as target networks for each dataset as [18] mentioned: DRN-D-105 [29] for Cityscapes, Upernet-101 for ADE20K [25].

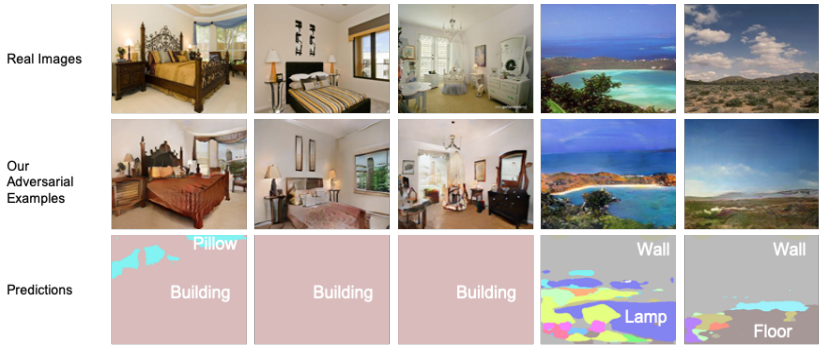

Fig. 3. Visual Results on ADE20K. By changing nuisance factors of the real images, such as the color of the bed, our method fools the semantic segmentation model.

We also compare our attack with GAN-based attack [24] on Cityscapes: AdvGAN. We adapt the GAN structure for the segmentation task. We reproduce AdvGAN with two different hyper-parameter settings ($\lambda_{adv} = -1, -100$). For Song et al.'s [19] method, we set the dimension of input random noise z to 256. We run SPADE with 50 epochs and then run another 50 epochs to optimize the adversarial loss on Cityscapes.(see Table 3).

We also consider DAG [26] and Houdini [3], two state-of-the-art adversarial attacks for semantic segmentation as our baseline methods on Cityscapes. For DAG, we set hyper-parameter γ = 0.5 and number of iteration N = 200 as author suggests in [26]. For Houdini, we replace the loss function to surrogate Houdini loss in PGD attack with setting l_∞ norm bound size ϵ = 8, number of iteration N = 300. Since Houdini and DAG are both white-box attack, we also choose DRN-D-105 [29] for Cityscapes as our target segmentation network for a fair comparison.

Evaluation metric. We evaluate the effectiveness of our attack by following metrics:
(1) *Attack Success Rate(ASR)*: Due to the dense output property of the semantic segmentation task, the evaluation of the attack success rate is different from that of the classification [19]. In this paper, we consider an attack is successful if over 95% pixels in an image are mis-segmented.
(2) *Mean Intersection-over-Union (mIoU)*: For measuring the effect of different attack methods on the target networks, we measure the drop in recognition accuracy using mIoU score which is widely used in semantic segmentation tasks [4, 31]—lower mIoU score indicates better adversarial example.
(3) *Fréchet Inception Distance (FID)*: We use FID [6] to compute the distance between the distribution of our adversarial examples and the distribution of the real images; small FID stands for the high quality of generated images.
(4)*Amazon Mechanical Turk (AMT)*: AMT is used to verify the success of our unrestricted adversarial attack. Here, we randomly select 250 generated adversarial images under two experimental settings from each dataset to generate AMT assignments. Each assignment is answered by 3 different workers and each worker has 3 minutes to make decision. We use the result of a majority vote as each assignment's final answer.

Table 1. Comparison with norm-bounded attacks Compared to norm-bounded attacks (FGSM,PGD) with bound size 8, our method achieves significantly higher attack success rates(ASR), larger mIoU drop and low FID on Cityscapes(DRN-105) and ADE20K(Upernet-101) dataset.

DRN-105(Cityscapes)					Upernet-101(ADE20K)				
Method	ASR	mIoU	FID	Time	Method	ASR	mIoU	FID	Time
FGSM+Real Image	0%	0.196	-	0.5s/Image	FGSM+Real Image	2.6%	0.178	-	0.5s/Image
FGSM+Vanilla SPADE	0%	0.152	82.144	1s/Image	FGSM+Vanilla SPADE	2.6%	0.152	60.563	1s/Image
PGD+Real Image	22.2%	0.036	-	30s/Image	PGD+Real Image	11.5%	0.070	-	34s/Image
PGD+Vanilla SPADE	43.4%	0.022	69.162	31s/Image	PGD+Vanilla SPADE	24.2%	0.020	62.289	33s/Image
Ours	**84.4%**	**0.01**	**67.302**	**0.25s/Image**	**Ours**	**57.7%**	**0.011**	**53.49**	**0.32s/Image**

Table 2. FID Comparison between our method and state-of-art semantic image synthesis models.Our method outperforms Pix2PixHD and CRN and achieve comparable FID with vanilla SPADE on Cityscapes and ADE20K.

Dataset \ Model	Vanilla SPADE	Pix2PixHD	CRN	**Ours**
Cityscapes	62.939	95.0	104.7	**67.302**
ADE20K	33.9	81.8	73.3	**53.49**

(5)*Running Time*: We measure the running time of different attack methods for generating a single adversarial example.

5 Experiment Result

Evaluating Generated Adversarial Images. Here, we compare our attack with norm bounded adversarial attacks, including FGSM and PGD [5, 13], for two datasets. We set the l_∞ norm bound size ϵ to 8 for both FGSM and PGD. We set the number of step for PGD as 12 [10, 1] . We apply FGSM and PGD on both real images and synthetic images by vanilla SPADE, and compare their attack success rate, mIoU scores FID and running time with ours. The results (see Table 1) show that PGD and FGSM attacks can barely attack target networks even with large time budge. In detail, PGD takes about 30s to generate perturbation for each real and synthetic image and achieves 22.2% and 43.4% ASR on DRN-105 on Cityscapes. In contrast, our method achieves 84.4% ASR with 0.25s/Image on DRN-105, 57.7% ASR with 0.32s/Image on Upernet-101.

Compare to vanilla SPADE, the FID of our adversarial examples increases slightly (62.939 to 67.302 on Cityscapes,see Table 2) indicating our samples have comparable quality and variety. Note that we only train our method half epochs as reported in [18] and achieve 53.49 FID on ADE20K, still smaller than other leading semantic image synthetic models. Figure 2, 3 show qualitative results.

GAN-based and Segmentation Adversarial Attack. We compare the attack success rate, mIoU drop and performance between AdvGAN [24],Houdini [3],DAG [26], Song's attack [19] and ours (see Table 3). For AdvGAN, we also set the same bound size(8) as in the traditional norm-bound attack. With suggested hyper-parameter setting in [24], although AdvGAN has comparable time cost($0.2s$/Image) with our method, AdvGAN generated examples only show a 0% attack success rate and 0.264 mIoU score. After adjusting the hyper-parameter, generated adversarial examples can achieve 7.8% attack success rate, but still worse than ours. Similar to AdvGAN, Song's attack also shows limited effectiveness on segmentation tasks.(0% ASR and 0.461 mIoU). From our experimental results, it turns out that the effectiveness of current GAN-based attacks are

Method	Bound Size	ASR	mIoU	Performance
AdvGAN	8	0%	0.264	0.2s/Image
AdvGAN (λ_{adv} = -100)	8	7.8%	0.055	0.2s/Image
Houdini	8	80.4%	0.013	40s/Image
DAG	-	73.8%	0.014	30s/Image
Song's Attack	-	0%	0.461	35s/Image
Ours	-	**84.4%**	**0.01**	**0.25s/Image**

Table 3. Comparison with GAN-based and segmentation attacks on DRN-105. We present the mIoU score , attack success rate and performance of Houdini [3],DAG [26],Song's unrestricted attack [19] and AdvGAN [24] on Cityscapes.

Table 4. Results of AMT study: We present the details of AMT evaluation results on Cityscapes and ADE20K datasets.

	Semantic Consistence Test					Fidelity AB Test	
Datasets	True Positive	True Negative	False Positive	False Negative	Accuracy	Ours Vs. Vanilla SPADE	Ours Vs. Real Images
Cityscapes	124	112	1	13	94.4%	68.4%	49.2%
ADE20K	125	120	0	5	98.0%	75.2%	46.6%

even worse than the traditional norm-bound attacks, showing that its applicability to only classification tasks. Compared to Houdini and DAG, our attack also shows better attack result (4% higher than Houdini, 10.6% higher than DAG for attack success rate). Meanwhile, our method is over 100 times faster than both DAG and Houdini while generating adversarial examples. In conclusion, ours can generate adversarial examples successfully in an effective manner.

Human Evaluation. Using Amazon Mechanical Turk (AMT), we evaluate how a human perceives generated adversarial images (see Table 4). A detailed result is presented in the Supplementary part. This is done in two settings:

(1) *Semantic Consistency Test*: We give AMT workers a pair of images: a generated adversarial image and a semantic label and ask them if the semantic meaning of given synthetic image is consistent with the given semantic label. We notice that users can identify the semantic meaning of our adversarial examples precisely (94.4% for Cityscapes, 98.0% for ADE20K). This shows that our attack does not change the semantic meaning of the original input.

(2) *Fidelity AB Test*: We compare the visual fidelity of ours with real images. We give workers the semantic ground truth label, our generated adversarial image and real image and ask them to select the more appropriate image corresponding to the ground truth label. 49.2% and 46.6% users favor our examples over real images for Cityscapes and ADE20K dataset which shows that our generated adversarial examples are indistinguishable for human.

Attack Under Defense. We show that our unrestricted adversarial examples can attack the models adversarial trained with norm-bounded adversarial images successfully [5]. We follow the setting introduced by [13] during the adversarial training. After the training phase, we use PGD with the same setting to generate norm-bounded perturbation and add it on both real and synthetic images by vanilla SPADE. We find that real and synthesized images with perturbation make mIoU decrease to 0.325 and 0.225 on robust DRN-105, 0.239 and 0.197 on robust Upernet-101, respectively. In contrast, our adversarial examples can achieve 0.033 mIoU score on robust DRN-105, and 0.024 on robust Upernet-101 indicating that our examples can successfully surpass the robust models trained with norm-bounded adversarial examples. More detail settings can be found in the supplementary materials.

References

1. Arnab, A., Miksik, O., Torr, P.H.: On the robustness of semantic segmentation models to adversarial attacks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (Jun 2018). https://doi.org/10.1109/cvpr.2018.00099, http://dx.doi.org/10.1109/cvpr.2018.00099
2. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples (2018)
3. Cisse, M., Adi, Y., Neverova, N., Keshet, J.: Houdini: Fooling deep structured prediction models. arXiv preprint arXiv:1707.05373 (2017)
4. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
5. Goodfellow, I., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Learning Representations (2015), http://arxiv.org/abs/1412.6572
6. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium (2017)
7. Hong, S., Yan, X., Huang, T., Lee, H.: Learning hierarchical semantic image manipulation through structured representations (2018)
8. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Jul 2017). https://doi.org/10.1109/cvpr.2017.632, http://dx.doi.org/10.1109/CVPR.2017.632
9. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2013), http://arxiv.org/abs/1312.6114, cite arxiv:1312.6114
10. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial machine learning at scale (2016)
11. Lee, D., Liu, S., Gu, J., Liu, M.Y., Yang, M.H., Kautz, J.: Context-aware synthesis and placement of object instances. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31, pp. 10393–10403. Curran Associates, Inc. (2018), http://papers.nips.cc/paper/8240-context-aware-synthesis-and-placement-of-object-instances.pdf
12. Liang, M., Yang, B., Chen, Y., Hu, R., Urtasun, R.: Multi-task multi-sensor fusion for 3d object detection. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 7337–7345 (2019)
13. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
14. Mao, C., Gupta, A., Nitin, V., Ray, B., Song, S., Yang, J., Vondrick, C.: Multitask learning strengthens adversarial robustness (2020)
15. Mao, C., Zhong, Z., Yang, J., Vondrick, C., Ray, B.: Metric learning for adversarial robustness (2019)
16. Mirza, M., Osindero, S.: Conditional generative adversarial nets (2014)
17. Papernot, N., McDaniel, P., Wu, X., Jha, S., Swami, A.: Distillation as a defense to adversarial perturbations against deep neural networks. In: 2016 IEEE Symposium on Security and Privacy (SP). pp. 582–597. IEEE (2016)
18. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization (2019)
19. Song, Y., Shu, R., Kushman, N., Ermon, S.: Constructing unrestricted adversarial examples with generative models (2018)

20. Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Cardoso, M.J.: Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Deep learning in medical image analysis and multimodal learning for clinical decision support, pp. 240–248. Springer (2017)
21. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhao, S., Cheng, S., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in perception for autonomous driving: Waymo open dataset (2019)
22. Tian, Y., Pei, K., Jana, S., Ray, B.: Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In: Proceedings of the 40th international conference on software engineering. pp. 303–314. ACM (2018)
23. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (Jun 2018). https://doi.org/10.1109/cvpr.2018.00917, http://dx.doi.org/10.1109/cvpr.2018.00917
24. Xiao, C., Li, B., Zhu, J.y., He, W., Liu, M., Song, D.: Generating adversarial examples with adversarial networks. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (Jul 2018). https://doi.org/10.24963/ijcai.2018/543, http://dx.doi.org/10.24963/ijcai.2018/543
25. Xiao, T., Liu, Y., Zhou, B., Jiang, Y., Sun, J.: Unified perceptual parsing for scene understanding. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 418–434 (2018)
26. Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., Yuille, A.: Adversarial examples for semantic segmentation and object detection. 2017 IEEE International Conference on Computer Vision (ICCV) (Oct 2017). https://doi.org/10.1109/iccv.2017.153, http://dx.doi.org/10.1109/ICCV.2017.153
27. Xie, C., Wu, Y., Maaten, L.v.d., Yuille, A.L., He, K.: Feature denoising for improving adversarial robustness. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (Jun 2019). https://doi.org/10.1109/cvpr.2019.00059, http://dx.doi.org/10.1109/CVPR.2019.00059
28. Yao, S., Hsu, T.M.H., Zhu, J.Y., Wu, J., Torralba, A., Freeman, W.T., Tenenbaum, J.B.: 3d-aware scene manipulation via inverse graphics (2018)
29. Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 472–480 (2017)
30. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4471–4480 (2019)
31. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. arXiv preprint arXiv:1608.05442 (2016)