Unrestricted Adversarial Attacks for Semantic Segmentation

Anonymous authors
Paper under double-blind review

Abstract

Despite the rapid development of adversarial attacks for machine learning models, many types of new adversarial examples still remain unknown. Uncovered types of adversarial attacks pose serious concern for the safety of the models, which raise the question about the effectiveness of current adversarial robustness evaluation. Semantic segmentation is one of the most impactful applications of machine learning; however, their robustness under adversarial attack is not well studied. In this paper, we focus on generating unrestricted adversarial examples for semantic segmentation models. We demonstrate a simple yet effective method to generate unrestricted adversarial examples using conditional generative adversarial networks (CGAN) without any hand-crafted metric. The naïve implementation of CGAN, however, yields inferior image quality and low attack success rate. Instead, we leverage the SPADE (Spatially-adaptive denormalization) structure with an additional loss item, which is able to generate effective adversarial attacks in a single step. We validate our approach on the well studied Cityscapes and ADE20K datasets, and demonstrate that our synthetic adversarial examples are not only realistic, but also improves the attack success rate by up to 41.0% compared with the state of the art adversarial attack methods including PGD attack.

1 Introduction

Despite their impressive accuracy and wide adaption, machine learning models remain fragile to adversarial attacks (Szegedy et al. (2013); Carlini & Wagner (2017); Papernot et al. (2016a)), which raises serious concerns for deploying them into real-world applications, especially in safety and security-critical systems. Extensive efforts have been made to combat these adversarial attacks: robust models are trained such that they are not easily evaded by adversarial examples (Goodfellow et al. (2015); Papernot et al. (2016b); Madry et al. (2018)). Although these defense methods improve the models' robustness, they are mostly limited to addressing norm bounded attacks such as PGD (Madry et al. (2018)). Realistic adversarial attacks beyond norm bound thus remain a major concern to those robust models, which spur extensive efforts to explore stronger and realistic adversarial attacks, e.g., using Wasserstein bound measurement (Wong et al. (2019)), realistic image transformations (Engstrom et al. (2017)) etc. In particular, Song et al. (2018) propose unrestricted adversarial attacks using conditional GAN for the image classification models, which is a big step toward realistic attacks beyond human crafted constrains. However, due to their model design, they are mostly restricted to low-resolution images—for high resolution, the generated images are not very realistic.

The problem of achieving realistic adversarial attacks and defenses aggravate further for more difficult visual recognition tasks such as semantic segmentation, where one needs to attack order of magnitude more pixels while achieving a consistent perception by human. It is essential to make the segmentation models robust against adversarial attacks, especially due to their applicabilities in autonomous driving (Ess et al. (2009)), medical imaging Ronneberger et al. (2015); Shen et al. (2018), and computer-aided diagnose system (Milletari et al. (2016)). Unfortunately, we show that existing attack methods that are primarily designed for simple classification tasks do not generalize well to semantic segmentation. For instance, following the work of Arnab et al. (2018), we show that the norm-bound perturbation becomes human visible since larger bounds are required for launching a successful attack. The unrestricted adversarial attack, on the other hand, is not constrained by the norm bounded budget, which can expose more vulnerabilities of a given machine learning model.

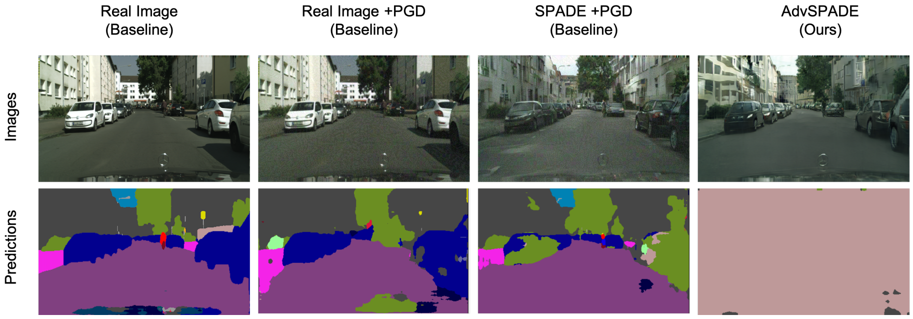


Figure 1: **Illustrating the effectiveness of AdvSPADE generated unrestricted adversarial example compared to norm bounded attacks**. The first column shows a real image from Cityscapes dataset and its prediction result of DRN-105 segmentation network. The 2nd column shows the result of applying PGD to real images. The 3rd column shows the result of applying PGD to a synthesized image generated by SPADE, while the 4th column shows an unrestricted adversarial image created by AdvSPADE which completely fools DRN-105. Note that, while the image in the 4th column is clean, there are conspicuous noises in the 2nd and 3rd column images. Also, the segmentation results are still good for the first three columns. Notice that the objects critical for a driving system like street lamps, cars, buildings are all well preserved in our generated image (fourth column); the only differences are in their colors and textures, which does not affect the semantics for a human driver. However, the prediction of the segmentation model is totally wrong in the fourth column, which demonstrates the effectiveness of our unrestricted adversarial attacks for segmentation models.

However, the quality and resolution of the unrestricted adversarial images generated by those methods are low and limited to simple images like the handwritten digits.

In this paper, we present the first realistic unrestricted adversarial attack, AdvSPADE, for semantic segmentation models. Figure 1 illustrates the effectiveness of our proposed method. To generate realistic images for the semantic segmentation models, we use SPADE (Park et al. (2019)), a state-of-the-art conditional generative model to generate high-resolution images (up to 1 million pixels). We further add an additional adversarial loss term on the original SPADE architecture to fool the target model. Such a simple yet effective method, AdvSPADE, helps us to create a wide variety of adversarial examples from a single image in a single step. Empirical results show that successful adversarial attacks vary in different styles (e.g., different lighting condition, texture, etc.), suggesting a large volume of vulnerabilities exist for semantic segmentation models that are beyond hand-crafted perturbations. We further show that augmenting the training data with such realistic adversarial examples have the potential to improve the models' robustness.

To this end, our main contribution is: (1) We propose a new realistic attack for semantic segmentation which surpasses the existing state-of-the-art robust models. We demonstrate the existence of rich varieties of unrestricted adversarial examples besides the previously known ones. (2) We demonstrate that augmenting the training dataset with our new adversarial examples have the potential of improving the robustness of existing models. (3) We present an empirical evaluation of our approach using two popular semantic segmentation dataset. First, we evaluate the quality of our generated adversarial examples using Amazon Mturk and prove that our samples are indistinguishable to natural images for humans. Using these adversarial images, we further show that the attack success rate can be improved by up to 41%.

2 Related Work

Semantic Segmentation. It is one of the most critical tasks in the computer vision field, which can be considered as a multi-output classification task that provides more fine-granular information in the prediction (Barrow & Tenenbaum (1981)). Plenty of network architectures have been proposed to address semantic segmentation task efficiently (Ronneberger et al. (2015); Long et al. (2015); Chen et al. (2017a); Badrinarayanan et al. (2017)). Generally, a segmentation network contains two parts: an encoder E and a decoder D. Encoder is for the feature extraction and decoder is for the dimension restoration. Although the structure design of segmentation networks have been well-studied, very few studies (Arnab et al. (2018); Xie et al. (2017)) look into the robustness of this class of networks against adversarial examples.

Adversarial Attacks. The attacks are examples that carefully crafted to mislead the prediction of machine learning models, while still perceived the same by the human. In recent years, researchers have proposed multiple methods for generating adversarial examples for the image classification task (Goodfellow et al. (2015); Kurakin et al. (2016); Carlini & Wagner (2017); Madry et al. (2018)), where the target model is fooled by the adversarial images. Hand-crafted metrics, such as L_p norm bound (Madry et al. (2018); Feinman et al. (2017)) and Wesstrasien distance (Wong et al. (2019)), are applied to the generation process to preserve the semantic meaning of the adversarial examples to human. Recently, (Song et al. (2018)) proposes to use generative adversarial network to generate unrestricted adversarial attacks for image classification. By leveraging Auxiliary Classifier Generative Adversarial Network ($AC\text{-}GAN$) (Odena et al. (2016)), the model was able to generate low quality adversarial examples from scratch and beyond any norm bound. (Wang et al. (2019)) proposed a new generative model called ($AT\text{-}GAN$) to learn a transformation between a pre-trained GAN and a adversarial GAN which can generate adversarial examples for the target classifier. The work generating adversarial examples using GAN (Song et al. (2018); Wang et al. (2019)) adopt two step procedures which involves many steps of gradient descent on the second part. In contrast, our adversarial examples are generated in a single step which is more efficient.

A few studies focused on the adversarial attack on modern semantic segmentation networks. Arnab et al. (2018) 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. (Xie et al. (2017)) propose 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. However, all of the attack methods rely on norm-bounded perturbations, which only cover a small fraction of all the feasible adversarial examples.

Defense Methods. Adversarial training is the state-of-the-art method for training robust classifiers. (Goodfellow et al. (2015); Lyu et al. (2015); Shaham et al. (2018); Szegedy et al. (2013); Hinton et al. (2015); Papernot & McDaniel (2017); Xu et al. (2018); Madry et al. (2018)). Besides, Other defense methods like the input transformation, including rescaling, JPEG compression, Gaussian blur, HSV jitter, grayscale against adversarial attack on semantic segmentation networks are evaluated by (Arnab et al. (2018)). These input transformation methods, however, were shown to rely on obfuscated gradients and give a false sense of robustness (Athalye et al. (2018)). On the other hand, (Athalye et al. (2018)) endorsed the robustness of the model trained with adversarial training, which is the state-of-the-art adversarial robust model available. In this paper, we demonstrate the robustness of our attack method by bypassing the adversarial training instead of evaluating on the defense method relies on obfuscated gradients.

3 Generating Adversarial Examples

In this section, we introduce our methodology, AdvSPADE, for generating unrestricted adversarial examples for semantic segmentation. For this purpose, we leverage a conditional Generative Adversarial Networks, SPADE. The main goal of a standard conditional GAN is to synthesize realistic images that will fool the discriminator. The generation of adversarial attack, however, also requires to fool the segmentation model under attack. AdvSPADE thus adds an additional loss function to fool both the discriminator and the segmentation model. Figure 2 shows the overall workflow. The rest of the section describes the relevant terminologies and the new loss function in details.

Unrestricted Adversarial Examples. Consider $\mathcal{I}$ as a set of images and $\mathcal{C}$ be the set of all possible categories for $\mathcal{I}$. Suppose $o : \mathcal{O} \subseteq \mathcal{I} \rightarrow \mathcal{C}$ is an oracle that can map any image from its domain $\mathcal{O}$ to $\mathcal{C}$ correctly. A classification model $\mathcal{F} : \mathcal{I} \rightarrow \mathcal{C}$ can also provide a class prediction for any given images in $\mathcal{I}$. Under the assumption that $\mathcal{F} \neq o$, an unrestricted adversarial example x is any image which meets following requirements (Song et al. (2018)): $x \subseteq \mathcal{O}$, $o(x) \neq \mathcal{F}(x)$.

Conditional Generative Adversarial Networks. A Conditional Generative Adversarial Network (Mirza & Osindero (2014)) 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 $\min_G \max_D V(G, D) = \mathbf{E}_{\text{x}\sim P_{data}(x)}[\log D(x|y)] + \mathbf{E}_{\text{z}\sim P_z(z)}[\log(1 - D(G(z|y)))]$

Unrestricted Adversarial Loss. We design an adversarial loss term for the unrestricted adversarial examples generation. We mainly focus on the untargeted attack in this paper though our approach is general and can be simply applied on targeted attack. Intuitively, the SPADE generator is trained to mislead the prediction of target segmentation network. The synthetic images are not only required

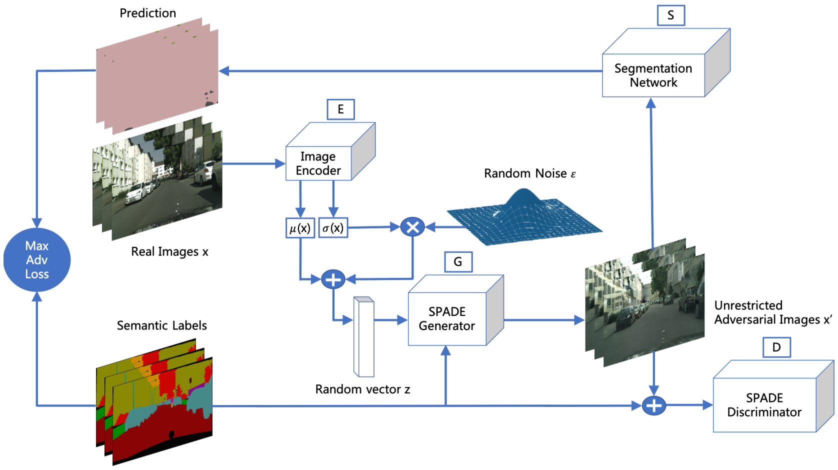


Figure 2: **Illustration of proposed AdvSPADE Architecture for generating unrestricted adversarial examples.** Image encoder E takes real images x as input to compute mean and variance vectors $(\mu(x), \sigma(x))$ and apply reparameterization trick to generate random noise z. SPADE generator G considers z and semantic labels s and generates synthetic images x'. Next, x's are fed into a fixed pre-trained target segmentation network S and encouraged to mislead S's predictions by maximizing adversarial loss between predictions and semantic labels. Meanwhile, x', as SPADE discriminator D's input, also aims to fool D. D is trained to reliably distinguish between generated (x') and real images (x). Random sampled ϵ brings randomness into the model so that G can generate various adversarial examples. Notice that, due to the adversarial loss item, prediction results at the top left corner of the figure are completely mis-segmented.

to fool the discriminator for the conditional GAN but also need to be mis-segmented by the target segmentation network. To achieve this goal, 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 input random vector by z. We define the untargeted version of Unrestricted Adversarial Loss as follows:

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

We select Dice Loss Sudre et al. (2017) as the objective function f. An image encoder E processes a real image I and generates a mean vector $\mu_{E(I)}$ and a variance vector $\sigma_{E(I)}$ and then compute the noise input z according to reparameterization trick Kingma & Welling (2013).

$$\boldsymbol{z} = \mu_{E(I)} + \sigma_{E(I)} \cdot \epsilon, \epsilon \sim \mathcal{N}(0, I) \tag{2}$$

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

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

More details about remaining loss terms in Eq 3 can be found in (Park et al. (2019); Wang et al. (2018)). To speed up the generation process as well as the quality of synthesized images, we follow Spatially-adaptive denormalization, as proposed by Park et al. (2019).

Spatially-adaptive denormalization. Our model uses SPADE architecture (Park et al. (2019)) as the conditional GAN model, where the Batch Normalization (Ioffe & Szegedy (2015)) is replaced with Spatially-adaptive denormalization. This method is proved to maintain the semantic segmentation information which will get lost during the subsampling. Please refer to Appendix A for details.

4 Experimental Set-up

Datasets. We evaluate our method on two large-scale image segmentation datasets: Citsycapes (Cordts et al. (2016)) and ADE20K (Zhou et al. (2016)). Cityscapes contains street view images from 50 German cities and 19 semantic classes, and it consists of 3000 training and 500 validation images. ADE20K covers 150 semantic classes in multiple real world scenes, where the training and validation set contains 20100 and 2000 images respectively.

Training Details. Following Park et al. (2019), we apply the Spectral Norm (Miyato et al. (2018)) in all layers for both generator and discriminator. We train our model with 50 epochs on Cityscapes. However, due to the large size of ADE20K and computation limits, we only run 100 epochs on ADE20K rather than 200 epochs reported in (Park et al. (2019)). We set the learning rate of the generator and discriminator both equal to 0.0002 and start to decay learning rate linearly from 50-th epoch when trained on ADE20K. We employ the ADAM (Kingma & Ba (2014)) with $\beta_1 = 0.5$, $\beta_2 = 0.999$. In Equation 3, we set $\lambda_0 = 10$, $\lambda_1 = 10$, $\lambda_2 = 0.05$ for both Cityscapes and ADE20K and $\lambda_3 = 10$ for Cityscapes, $\lambda_3 = 70$ for ADE20K respectively. All experiments are done on a single NVIDIA TITAN Xp GPU.

Baseline Models. We compare AdvSPADE generated attacks with traditional *norm-bounded* attacks in two settings: real images with perturbation and generated clean images with perturbation. For the second setting, we use vanilla SPADE to generate clean images first, 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 (Park et al. (2019)) mentioned: DRN-D-105 (Yu et al. (2017)) for Cityscapes, Uppernet-101 for ADE20K (Xiao et al. (2018)). Besides, we also select several state-of-the-art open source segmentation networks to evaluate the transferability of our unrestricted adversarial examples as a black box setting: DRN-38, DRN-22 (Yu et al. (2017); Yu & Koltun (2016)), DeepLab-V3 (Chen et al. (2017b)), PSPNet-34-8s (Zhao et al. (2017)) for Cityscapes, PPM-18, MobilenetV2, Uppernet-50, PPM-101 (Zhou et al. (2018)) for ADE20K.

Evaluation metric. 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 (Song et al. (2018)). Let $\mathcal{I}^{H \times W \times C}$ be a set of RGB images with height H and width W and channel C. Let $\mathcal{L}^{H \times W}$ be the set of semantic labels for the corresponding images from $\mathcal{I}$. Suppose $o : \mathcal{O} \subseteq \mathcal{I} \to \mathcal{L}$ is an oracle that can map any images from its domain $\mathcal{O}$ which presents all images that look realistic to humans to $\mathcal{L}$ correctly. A segmentation model $\mathcal{S} : \mathcal{I} \to \mathcal{L}$ can provide pixel-wise predictions for any given images in $\mathcal{I}$. With this setting, we evaluate the following two categories of adversarial examples: 1) Given a constant $\epsilon > 0$ and a hand-craft norm $\|\cdot\|$, a *restricted adversarial example* x is an image that meets the following conditions: $x \subseteq \mathcal{O}$, $\exists x' \subseteq \mathcal{O} \; \|x - x'\| < \epsilon$, $\frac{\sum_{i,j}(o_{i,j}(x) \neq \mathcal{S}_{i,j}(x))}{H \cdot W} > \theta$. (2) An *unrestricted adversarial example* x is an image that meets following requirement: $x \subseteq \mathcal{O}$, $\frac{\sum_{i,j}(o_{i,j}(x) \neq \mathcal{S}_{i,j}(x))}{H \cdot W} > \theta$. Here, $o_{i,j}(x), \mathcal{S}_{i,j}(x)$ stand for the prediction given by oracle o and segmentation network $\mathcal{S}$ at pixel $x(i,j)$ respectively. $\theta \in [0,1]$ is a hyperparameter. In this paper, we set $\theta = 0.95$.

Given the nature of semantic segmentation task, misclassifying a single pixel does not lead an image fall into the class of adversarial examples. A legitimate adversarial example should have the property that the majority of pixels in it are misclassified (measured by mIoU score), and the adversarial image still looks realistic to humans (measured by FID score) with the same semantic meaning as the original images (measured by Amazon Turk). In particular, we use following three measures: 1. **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 (Cordts et al. (2016); Zhou et al. (2016))—lower mIoU score means better adversarial example. 2. **Fréchet Inception Distance (FID).** We use FID (Heusel et al. (2017)) 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. 3. **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.

5 Experiment Result

5.1 Evaluating Generated Adversarial Images

Here, we compare the adversarial images generated by AdvSPADE with the original real images and the clean synthetic images created by vanilla SPADE using mIoU and FID scores. Table 1 shows that compared to vanilla SPADE, AdvSPADE generated images under whitebox attack can lead to a giant decline on mIoU score (from 0.62 to 0.01 for DRN-105, from 0.403 to 0.011 for Uppernet-

Table 1: **The effectiveness of our proposed attack on Cityscapes and ADE20K dataset under mIoU metric under both whitebox and transfer based blackbox attacks.** We show the mIoU score of real images, standard SPADE synthesis images, and our generated attack images under different segmentation models. The **bold** shows results under whitebox attack and the non-bold numbers are mIoU achieved by transferring the adversarial examples generated with the bold network architecture to the non-bold ones. As we can see, our proposed method successfully mislead the segmentation models, while both the real and synthesis images are predicted correctly by the models.

Datasets	Seg Model	Real Images	SPADE	**AdvSPADE (Ours)**	Datasets	Seg Model	Real Images	SPADE	**AdvSPADE (Ours)**
Cityscapes	**DRN-105**	**0.756**	**0.620**	**0.010**	ADE20K	**Uppernet-101**	**0.420**	**0.403**	**0.011**
	DRN-38	0.714	0.551	0.407		MobilenetV2	0.348	0.317	0.110
	DRN-22	0.68	0.526	0.387		PPM-18	0.340	0.362	0.102
	DeepLab-V3	0.68	0.54	0.425		Uppernet-50	0.404	0.395	0.096
	PSPNet-34-8s	0.691	0.529	0.441		PPM-101	0.422	0.409	0.078

Table 2: **FID Comparison between AdvSPADE and state-of-art semantic image synthesis models**. The results show that AdvSPADE outperforms Pix2PixHD and CRN and achieve comparable FID with vanilla SPADE on Cityscapes.

Model / Dataset	Vanilla SPADE	Pix2PixHD	CRN	**AdvSPADE (Ours)**
Cityscapes	62.939	95.0	104.7	**67.302**
ADE20K	33.9	81.8	73.3	**53.49**

101). On different network architectures, our adversarial examples can also decrease of mIoU to a certain extent (around 20% on Cityscapes, 30% on ADE20K) showing strong transferability of our examples across models.

Compare to vanilla SPADE, the FID of our adversarial examples increases slightly (62.939 to 67.302 on Cityscapes, 33.9 to 53.49 on ADE20K, as shown in Table 2) which means our samples have comparable quality and variety . Note that we only train AdvSPADE half epochs as reported in Park et al. (2019) and achieve 53.49 FID on ADE20K, which is still smaller than other leading semantic image synthesis models such as Pix2PixHD (Wang et al. (2018)), SIMS (Qi et al. (2018)), CRN (Chen & Koltun (2017)). Qualitative results are shown in Figure 3. Moreover, by introducing an image encoder and KL Divergence loss, we can generate multi-modal stylized adversarial examples which are shown in the appendix.

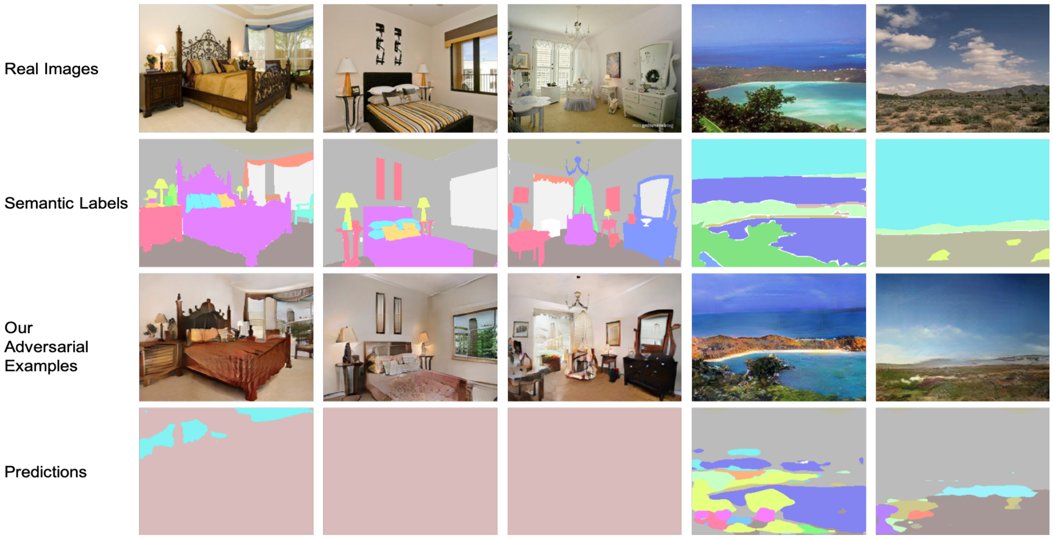


Figure 3: **Visual Results on ADE20K.** As we can see, the semantic meaning of the generated adversarial examples are well aligned to the original image, but are different in the style and mis-predicted by the segmentation model. For example, the color and strip on bed in the first two columns are changed, but human still perceive them as bed while the segmentation model predict the wrong label. The results demonstrate the effectiveness of our method for generating realistic adversarial examples that mislead the target model.

5.2 Norm-bounded Adversarial attacks

We compare the attack success rate of AdvSPADE with the state-of-the-art norm bounded adversarial attacks, including FGSM and PGD (Goodfellow et al. (2015); Madry et al. (2018)), for two

Table 3: **Attack Success Rate under white-box setting**. We present AdvSPADE and traditional norm-bounded attacks (FGSM,PGD) with different bound sizes $(0.25, 1, 8, 32)$. The results show that PGD and FGSM attacks can barely attack target networks with small bound size ($\epsilon = 0.25, 1, 8$). For instance, FGSM attack with bound size $\epsilon = 1$ on real and vanilla SPADE generated images achieve 0% attack success rate on DRN-105 network on Cityscapes. In contrast, AdvSPADE achieves high attack success rate (84.4% and 57.7% on DRN-105 and Uppernet-101, respectively).

DRN-105				Uppernet-101			
Method	Bound Size (ϵ)	Real Images +Perturbation	Vanilla SPADE +Perturbation	Method	Bound Size (ϵ)	Real Images +Perturbation	Vanilla SPADE +Perturbation
FGSM	0.25	0%	0%	FGSM	0.25	0.4%	0.9%
	1	0%	0%		1	0.9%	1.8%
	8	0%	0%		8	2.6%	2.6%
	32	15.6%	16.4%		32	6.1%	8.0%
PGD	0.25	0%	0%	PGD	0.25	0.4%	0.9%
	1	0%	0%		1	0.8%	2.8%
	8	22.2%	43.4%		8	11.5%	24.2%
	32	33.8%	47.2%		32	39.0%	44.1%
AdvSPADE (Ours)		**84.4%**		**AdvSPADE (Ours)**		**57.7%**	

Table 4: **Quantitative comparison of adversarial images generated by different attack methods under white-box setting.** We show the mIoU score of AdvSPADE generated examples and norm-bounded attacks on both real and standard SPADE generated images with multiple bound sizes $(0.25, 1, 8, 32)$ for DRN-105 (Cityscapes) and Uppernet-101 (ADE20K). Digits in the parentheses show the corresponding FID scores. The results indicate that to achieve similar mIoU as AdvSPADE, traditional norm-bounded attacks need large size perturbation which is easily detectable by the human. However, our unrestricted adversarial examples remain invisible to human, which reveals the effectiveness of our proposed method.

DRN-105				Uppernet-101			
Method	Bound Size (ϵ)	Real Images +Perturbation	Vanilla SPADE +Perturbation	Method	Bound Size (ϵ)	Real Images +Perturbation	Vanilla SPADE +Perturbation
FGSM	0.25	0.557	0.431(63.354)	FGSM	0.25	0.346	0.286(33.821)
	1	0.408	0.355(64.455)		1	0.278	0.221(35.254)
	8	0.196	0.152(82.144)		8	0.178	0.152(60.563)
	32	0.009	0.009(248.175)		32	0.070	0.048(166.724)
PGD	0.25	0.557	0.431(63.354)	PGD	0.25	0.346	0.286((33.821)
	1	0.339	0.287(63.971)		1	0.276	0.181(34.876)
	8	0.036	0.022(69.162)		8	0.070	0.022(62.289)
	32	0.013	0.009(89.998)		32	0.013	0.007(113.553)
AdvSPADE (Ours)		**0.01(67.302)**		**AdvSPADE (Ours)**		**0.011(53.49)**	

datasets. We set the l_∞ norm bound size ϵ to $\{0.25, 1, 8, 32\}$ for both FGSM and PGD. For PGD, we follow the (Kurakin et al. (2016); Arnab et al. (2018)) and set number of attack iterations to $min\{\epsilon + 4, \lceil 1.25\epsilon \rceil\}$. We apply FGSM and PGD on both real images and synthetic images by vanilla SPADE, and compare their mIoU scores and FID with ours. Overall, AdvSPADE achieves higher attack success rates on both datasets (84.4% on Cityscapes, 57.7% on ADE20K) than traditional norm-bounded attack approaches, as shown in Table 3.

Table 4 further reveals that for both FGSM and PGD attack, to decrease the mIoU to the same level as AdvSPADE (mIoU = 0.01), the generated perturbation becomes conspicuous ($\epsilon = 32$) so that human can easily distinguish adversarial examples from clean images. FID also reflects the decline of adversarial images' quality. Secondly, adversarial examples generated by FGSM and PGD attack can not make mIoU drop down to the same level as AdvSPADE if it is required to maintain the quality of the samples. Consider the adversarial samples on Cityscapes generated by vanilla SPADE and add perturbation with $\epsilon = 1$, their FID (64.455) is comparable with our samples, but mIoU (0.355) is much larger than ours (0.01). Figure 4 illustrates the difference between AdvSPADE samples and norm-bounded samples on the same level of mIoU score. We can easily see the noise pattern in norm-bounded samples rather than in our examples.

5.3 Human Evaluation

Using Amazon Mechanical Turk (AMT), we evaluate how a human perceives AdvSPADE generated adversarial images. A detailed result is presented in the Appendix. This is done in two settings:

(1) Semantic Consistency Test: Here, we aim to validate that semantic meanings of our adversarial examples are consistent with their respective ground truth labels. If it is true, humans will segment our adversarial examples correctly. However, asking workers to segment every pixel in an image is time-consuming and inefficient. Instead, we give AMT workers a pair of images: a generated adversarial image and a semantic label (half of the images pairs are matched, and rest are mismatched) 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

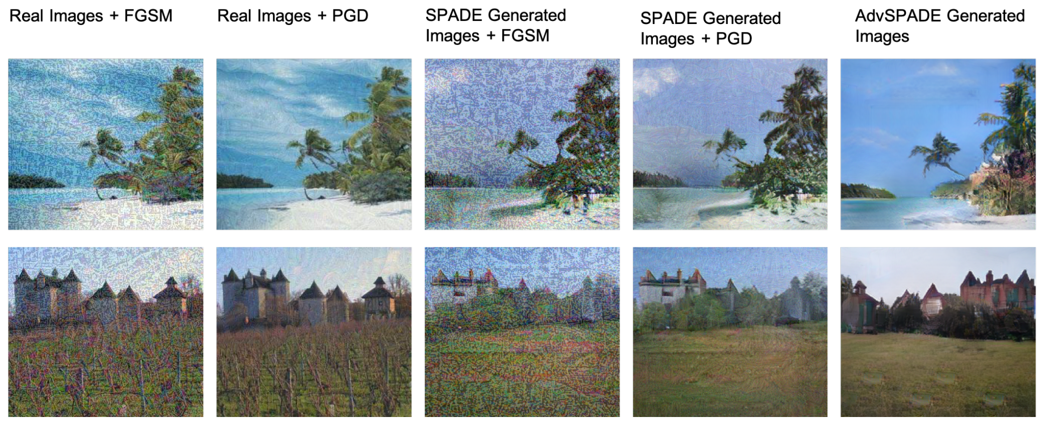

Real Images + FGSM
Real Images + PGD
SPADE Generated Images + FGSM
SPADE Generated Images + PGD
AdvSPADE Generated Images

Figure 4: **Comparison of norm-bounded samples and AdvSPADE generated unrestricted adversarial examples at the same mIoU level on ADE20K**. We Apply different attack methods to descend mIoU score to the same level (around 0.01) and show the visual comparison. First and second columns are adversarial examples generated by FGSM and PGD with $\epsilon = 32$ on real images, and third and fourth columns are the same on vanilla SPADE synthesis images. The last column is examples generated by AdvSPADE. We can clearly see the noise pattern in norm-bounded adversarial images rather than in our examples showing that our examples can attack target networks successfully, yet keep undetectable to human.

(94.4% for Cityscapes, 98.0% for ADE20K). In other words, the segmentation network's reaction toward our adversarial examples is inconsistent with human, which proves the success of our attack.

(2) Fidelity AB Test: We compare the visual fidelity of AdvSPADE with vanilla SPADE. Here we give workers the semantic ground truth label and two generated images by AdvSPADE and vanilla SPADE respectively and ask them to select the more appropriate image corresponding to the ground truth label. 68.4% and 75.2% users favor our examples over vanilla SPADE for Cityscapes and ADE20K dataset respectively, which indicates competitive visual fidelity of our adversarial images.

5.4 Robustness Evaluation

In this section, we first show that robust training with norm-bounded adversarial images can defend restricted adversarial attacks successfully where perturbation can be added either on real images or synthesis images. Then, we show that our unrestricted adversarial examples can still attack these robust models successfully (Goodfellow et al. (2015)). Finally, we present the results of our experiment to build a more robust segmentation model based on our unrestricted examples. We follow the training setting introduced by (Madry et al. (2018)): we select PGD as the attack method and set the adversarial training epoch = 100 on Cityscapes, 50 on ADE20K, norm-bound size $\epsilon = 8$, attack iteration = 10, step size = 1. After the training phase, we use PGD with the same setting to generate norm-bounded perturbation and add it on both real images and synthesis images by vanilla SPADE.

It turns out that real and synthesized images with perturbation can only make mIoU decrease to $\mathbf{0.325}$ and $\mathbf{0.225}$ on robust DRN-105, $\mathbf{0.239}$ and $\mathbf{0.197}$ on robust Uppernet-101, respectively. In contrast, our adversarial examples can achieve $\mathbf{0.033}$ mIoU score on robust DRN-105, and $\mathbf{0.024}$ on robust Uppernet-101 which shows that our unrestricted adversarial examples can successfully surpass the robust models trained with norm-bounded adversarial examples. Next, we train a model with our unrestricted adversarial examples on the Cityscapes dataset and then apply PGD to attack. The result shows that PGD attack can only achieve 1.8% attack success rate on DRN-105. Since l_∞ norm-bound examples are unknown for the robust model defended by our samples, the low success rate reflects models gain stronger robustness from adversarial training with AdvSPADE examples.

6 Conclusion

In this paper, we explore the existence of adversarial examples beyond norm-bounded metric on the state-of-the-art semantic segmentation neural networks. By modifying the loss function of SPADE architecture, we are able to generate high quality unrestricted realistic adversarial examples beyond any l_p norms, which mislead segmentation networks' behavior. We demonstrate the effectiveness and robustness of our method by comparing ours with traditional norm-bounded attack methods. We also show that our generated adversarial examples can easily surpass the state-of-the-art defense method, which raises new concerns about the security of segmentation neural networks.

REFERENCES

Anurag Arnab, Ondrej Miksik, and Philip H.S. Torr. On the robustness of semantic segmentation models to adversarial attacks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun 2018. doi: 10.1109/cvpr.2018.00099. URL http://dx.doi.org/10.1109/cvpr.2018.00099.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, 2018.

Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

H. G. Barrow and J. M. Tenenbaum. Interpreting line drawings as three-dimensional surfaces. Artif. Intell., 17(1-3):75–116, August 1981. ISSN 0004-3702. doi: 10.1016/0004-3702(81)90021-7. URL http://dx.doi.org/10.1016/0004-3702(81)90021-7.

Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017, pp. 39–57, 2017.

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4):834–848, 2017a.

Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017b.

Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In Proceedings of the IEEE International Conference on Computer Vision, pp. 1511–1520, 2017.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling cnns with simple transformations, 2017.

Andreas Ess, Tobias Mueller, Helmut Grabner, and Luc J Van Gool. Segmentation-based urban traffic scene understanding. In BMVC, volume 1, pp. 2. Citeseer, 2009.

Reuben Feinman, Ryan R. Curtin, Saurabh Shintre, and Andrew B. Gardner. Detecting adversarial samples from artifacts. ArXiv, abs/1703.00410, 2017.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representations, 2015. URL http://arxiv.org/abs/1412.6572.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2017.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In NIPS Deep Learning and Representation Learning Workshop, 2015. URL http://arxiv.org/abs/1503.02531.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013. URL http://arxiv.org/abs/1312.6114. cite arxiv:1312.6114.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp. 1097–1105, 2012.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale, 2016.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440, 2015.

Chunchuan Lyu, Kaizhu Huang, and Hai-Ning Liang. A unified gradient regularization family for adversarial examples. 2015 IEEE International Conference on Data Mining, Nov 2015. doi: 10.1109/icdm.2015.84. URL http://dx.doi.org/10.1109/ICDM.2015.84.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE, 2016.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957, 2018.

Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In ICML, 2016.

Nicolas Papernot and Patrick McDaniel. Extending defensive distillation, 2017.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016a.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. 2016 IEEE Symposium on Security and Privacy (SP), May 2016b. doi: 10.1109/sp.2016.41. URL http://dx.doi.org/10.1109/sp.2016.41.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization, 2019.

Xiaojuan Qi, Qifeng Chen, Jiaya Jia, and Vladlen Koltun. Semi-parametric image synthesis. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun 2018. doi: 10.1109/cvpr.2018.00918. URL http://dx.doi.org/10.1109/cvpr.2018.00918.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention MICCAI 2015, pp. 234241, 2015. ISSN 1611-3349. doi: 10.1007/978-3-319-24574-4_28. URL http://dx.doi.org/10.1007/978-3-319-24574-4_28.

Uri Shaham, Yutaro Yamada, and Sahand Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. Neurocomputing, 307:195–204, 2018.

Guangyu Shen, Yi Ding, Tian Lan, Hao Chen, and Zhiguang Qin. Brain tumor segmentation using concurrent fully convolutional networks and conditional random fields. In Proceedings of the 3rd International Conference on Multimedia and Image Processing, pp. 24–30. ACM, 2018.

Yang Song, Rui Shu, Nate Kushman, and Stefano Ermon. Constructing unrestricted adversarial examples with generative models, 2018.

Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. Lecture Notes in Computer Science, pp. 240248, 2017. ISSN 1611-3349. doi: 10.1007/978-3-319-67558-9_28. URL http://dx.doi.org/10.1007/978-3-319-67558-9_28.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2013.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun 2018. doi: 10.1109/cvpr.2018.00917. URL http://dx.doi.org/10.1109/cvpr.2018.00917.

Xiaosen Wang, Kun He, and John E. Hopcroft. At-gan: A generative attack model for adversarial transferring on generative adversarial nets, 2019.

Eric Wong, Frank R. Schmidt, and J. Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations, 2019.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In Proceedings of the European Conference on Computer Vision (ECCV), pp. 418–434, 2018.

Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial examples for semantic segmentation and object detection. 2017 IEEE International Conference on Computer Vision (ICCV), Oct 2017. doi: 10.1109/iccv.2017.153. URL http://dx.doi.org/10.1109/ICCV.2017.153.

Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. Proceedings 2018 Network and Distributed System Security Symposium, 2018. doi: 10.14722/ndss.2018.23198. URL http://dx.doi.org/10.14722/ndss.2018.23198.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In International Conference on Learning Representations (ICLR), 2016.

Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 472–480, 2017.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jul 2017. doi: 10.1109/cvpr.2017.660. URL http://dx.doi.org/10.1109/cvpr.2017.660.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. arXiv preprint arXiv:1608.05442, 2016.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. International Journal on Computer Vision, 2018.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation, 2017.

A Appendix

A.1 Additional Experiments Details

Background on SPADE Model SPADE normalizes the activation of each layer in a neural network in a channel-wise manner and adjusts it by scale γ and bias β which are learned dynamically from two simple two-layer CNNs respectively. Let m, a denote the input semantic label and adjusted activation map. $h^i_{n,c,y,x}$ stands for the i-th layer's original activation value of n-th sample at location (c, x, y). (c, x, y means the channel, width and height of the activation map respectively)

$$a = \gamma^i_{c,y,x}(m)\frac{h^i_{n,c,y,x} - \mu^i_c}{\sigma^i_c} + \beta^i_{c,y,x}(m) \tag{4}$$

where μ^i_c and σ^i_c are calculated by:

$$\mu^i_c = \frac{1}{NH^iW^i}\sum_{n,y,x} h^i_{n,c,y,x} \tag{5}$$

$$\sigma^i_c = \sqrt{\frac{1}{NH^iW^i}\sigma_{n,y,x}(h^i_{n,c,y,x})^2 - (\mu^i_c)^2} \tag{6}$$

N is the number of samples in a batch. H, W are the height and width of the activation map in the corresponding layer.

SPADE Network Architectures In order to achieve the comparative quality of synthetic images,we basically follow the generator and discriminator architecture settings in (Park et al. (2019)). Due to SPADE module, encoder part is unnecessary for the generator. The simplified lightweight network takes semantic label and random vector as input, after going through alternate SPADE modules and upsampling layers, it can generate high-quality realistic images. For the discriminator structure, we also follow the guideline of SPADE which uses the multi-scale discriminator (Wang et al. (2018)) and loss function with the hinge loss term (Park et al. (2019)). SPADE inherits the property of BicycleGAN (Zhu et al. (2017)) and provides an easier and more straightforward way to synthesize multi-modal realistic images. Our AdvSPADE is based on the multi-modal version SPADE for generating various adversarial examples to increase the coverage of real-world adversarial examples. The implementation details of generator, discriminator and image encoder are shown in Fig 5 and Fig 6 .

AdvSPADE Training Re-normalization Normalization is an essential step in the pre-processing phase. We notice that in SPADE implementation, author leverages z-score normalization with $\mu = [0.5, 0.5, 0.5]$, $\sigma = [0.5, 0.5, 0.5]$, $\alpha = [255, 255, 255]$ to map input RGB images x into the range of $[-1, 1]$:

$$x_{norm} = \frac{(x/\alpha) - \mu}{\sigma} \tag{7}$$

After passing through SPADE generator, the generated image $x\prime$ we gain has the same range with normalized input image x_{norm}: $[-1, 1]$. In our AdvSPADE, we need to feed generated image $x\prime$ into the target segmentation network S. However, there will be a value range shifting. Since main semantic segmentation networks do not use the same normalization parameters (μ, σ) with SPADE. Currently, people use to compute corresponding mean vector ((μ)) and variance vector ((σ)) for a given dataset or directly use mean and variance vectors of Imagenet (Krizhevsky et al. (2012)) dataset: $\mu = [0.485, 0.456, 0.406]$, $\sigma = [0.229, 0.224, 0.225]$. We need to guarantee that generated adversarial example fed into the target network has the same range in training phase and testing phase. Let $\mu\prime$, $\sigma\prime$ be the mean and variance vector for semantic segmentation task. Before feeding generated adversarial examples into target segmentation network S, we need to do a re-normalization:

$$x\prime_{norm} = \frac{\sigma}{\sigma\prime} \cdot x\prime + \frac{\mu - \mu\prime}{\sigma\prime} \tag{8}$$

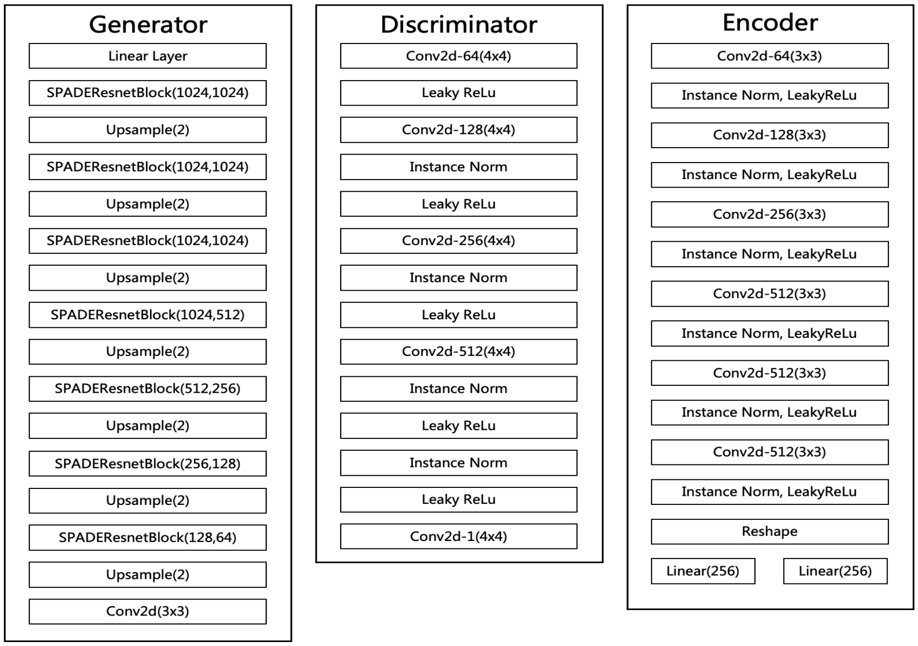


Figure 5: **Generator,discriminator and encoder architectures**

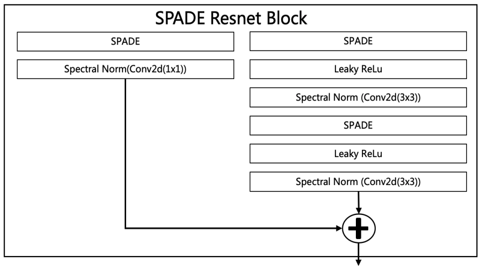

SPADE Resnet Block
SPADE
Spectral Norm(Conv2d(1x1))
SPADE
Leaky ReLu
Spectral Norm (Conv2d(3x3))
SPADE
Leaky ReLu
Spectral Norm (Conv2d(3x3))

Figure 6: **SPADE ResBlock Architecture**

Otherwise, even though generated unrestricted adversarial example can mislead the target network while training, it will still fail in the testing phase due to a large value shifting. We provide a simple quantitative computation to prove our statement.

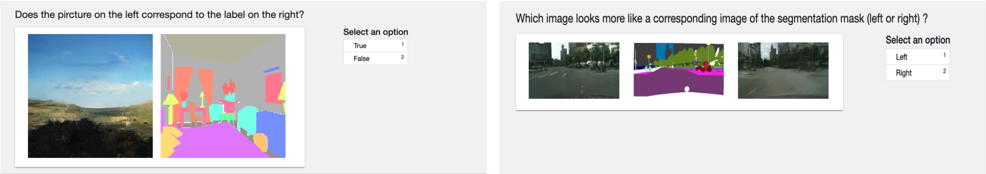
Let $\mu\prime = [0.485, 0.456, 0.406]$, $\sigma\prime = [0.229, 0.224, 0.225]$, $\mu = [0.5, 0.5, 0.5]$, $\sigma = [0.5, 0.5, 0.5]$ and the value at i-th,j-th position in $x\prime$: $x\prime(i, j) = [0.1, 0.1, 0.1]$. After saving $x\prime$ into a jpg file as an adversarial example, it will be map to $x_{RGB}(i, j) = \alpha \cdot [x\prime(i, j) \cdot \sigma + \mu] = [255, 255, 255] \cdot [[0.1, 0.1, 0.1] \cdot [0.5, 0.5, 0.5] + [0.5, 0.5, 0.5]] = [140.25, 140.25, 140.25]$. In the attack phase, $x_{RGB}(i, j)$ will be normalize to $x_{norm}(i, j) = [(x_{RGB}(i, j)/\alpha) - \mu\prime]\ \sigma\prime = [[[140.25, 140.25, 140.25]/[255, 255, 255]] - [0.485, 0.456, 0.406]]/[0.229, 0.224, 0.225] = [0.28, 0.42, 0.64]$. Notice that a valid adversarial pixel $[0.1, 0.1, 0.1]$ is mapped to a complete different value $[0.28, 0.42, 0.64]$ while attacking. There is no guarantee that $x_{norm}(i, j)$ can still mislead the target network which shows that the necessity of re-normalization while training AdvSPADE.

Robust training details We provide more detail settings about the robust training with AdvSPADE examples experiment described in Section 4. Note that the target network S is fixed while training AdvSPADE. In other words, we consider the gradient flow of target segmentation network S but do not update it in the whole training phase. We are able to gain effective unrestricted adversarial example after training model dozens epochs. However, the training time cost is non-negligible (around 48 hours on Cityscapes, 200 hours on ADE20K with single NVIDIA TITAN Xp GPU).

Algorithm 1 Adversarial Training with AdvSPADE

```
**Input:** initialized networks $E$,$G$,$D$, pre-trained network $S$, dataset $d$
**Output:** Updated $S$.
for number of training epochs do
  for number of iterations in each epoch do
    Sample minibatch of $n$ images $\{x^{(1)}, x^{(2)}, ..., x^{(n)}\}$ from $d$.
    Sample minibatch of corresponding $n$ semantic labels $\{l^{(1)}, l^{(2)}, ..., l^{(n)}\}$ from $d$.
    Update the generator by descending its stochastic gradient:
      $\nabla_{\theta_g} \frac{1}{n} \sum_{i=1}^{n} [ \sum_{k=1,2,3} \mathcal{L}_{GAN}(G, D_k)) + \lambda_0 \sum_{k=1,2,3} \mathcal{L}_{FM}(G, D_k) + \lambda_1 \mathcal{L}_{VGG}(G) + \lambda_3 \mathcal{L}_{ATK}(S, G)]$
    Update the discriminator by ascending its stochastic gradient:
      $\nabla_{\theta_d} \frac{1}{n} \sum_{i=1}^{n} [ \sum_{k=1,2,3} \mathcal{L}_{GAN}(G, D_k) + \lambda_0 \sum_{k=1,2,3} \mathcal{L}_{FM}(G, D_k)]$
    if epoch % k == 0 then
      Update the target segmentation network by ascending its stochastic gradient:
        $\nabla_{\theta_s} \frac{1}{n} \sum_{i=1}^{n} [\lambda_3 \mathcal{L}_{ATK}(S, G))]$
    end if
  end for
end for
return S
```

According to the definition from Madry et al. (2018); Goodfellow et al. (2015), in each training epoch, we apply certain attack method (PGD, iFGSM,etc) to generate adversarial examples and augment dataset with above examples, then using augmented dataset to fine-tune the target network. If we directly replace previously attack methods to our attack method, we need to train AdvSPADE dozens epochs in every adversarial training epoch when target network is updated which means the time cost will rise linearly related to the number of adversarial training epoch. In order to decrease the enormous time cost, we adopt a compromise adversarial training strategy with our unrestricted adversarial examples and achieve promising results. Instead of fixing S during training stage, we also optimize parameters of S each k epoch and encourage it to segment generated adversarial examples correctly. We combine robust training into the AdvSPADE training stage and reduce the time cost to a acceptable level (around 50 hours on Cityscapes with single NVIDIA TITAN Xp GPU). Algorithm is shown in 1. In this paper, we set $k = 1$. As we report in the 4, after adversarial training with AdvSPADE generated adversarial examples, the PGD atack can only achieve 1.8% attack success rate on the robust segmentation network.



Does the pircture on the left correspond to the label on the right?

Which image looks more like a corresponding image of the segmentation mask (left or right) ?

(a) Semantic Consistence Test

(b) Fidelity AB Test

Figure 7: **User Interfaces for AMT Workers.**

Table 5: **Results of AMT study**: We present the AMT evaluation results on Cityscapes and ADE20K datasets. Users are asked to answer two questions: (1)Semantic Consistence Test: Given an unrestricted generated adversarial image-label pair, is our example consistent with the semantic label? Note that true positive(TP) indicates the situation that the adversarial example and label is matched and user selects the meaning of this pair is consistent. TN,FP,FN describe the similar situations. respectively. (2)Fidelity AB Test: Given the synthetic images by AdvSPADE and standard SPADE from same input image and its semantic label, which image is more appropriate for the groudtruth label? We can see that in the semantic consistence test, AMT workers get 94.4% and 98% accuracy on test set sampled from Cityscapes and ADE20K. It proves that the semantic meaning of our unrestricted adversarial images are consistent with their labels from human perspective. In fidelity AB test, users prefer our generated images than vanilla SPADE generated images (68.4% on Cityscapes, 75.2% on ADE20k) which shows the fidelity of our generated adversarial examples.

Datasets	True Positive	True Negative	False Positive	False Negative	Accuracy	Ours Vs. Vanilla SPADE
Cityscapes	124	112	1	13	94.4%	68.4%
ADE20K	125	120	0	5	98.0%	75.2%

A.2 Additional Qualitative resutls

Table 5 shows the AMT evaluation results and Figure 7 presents the user interfaces we design for the two experiments. Figure 8 and Figure 9 show the variety of our unrestricted adversarial examples on two datasets. Figure 10 compares l_∞ norm-bounded samples and our unrestricted adversarial examples under the condition that making the mIoU of target network drop to the same level ($\mathbf{0.01}$) on Cityscapes. Figure 11 visualizes the relationship between bound size and mIoU decline on two datasets. The conclusion is that norm-bounded adversarial examples can achieve the same attack effect with our examples only with large bound size. Note that our examples do not contain any bound restriction, drawing it with other norm-bound examples in the same coordinate axis is for the convenience of comparison. Figure 12 and 13 compare the quality and attack effect of norm-bounded examples with multiple bound sizes and our examples. Figure 14 and 15 shows more unrestricted adversarial examples generated by AdvSPADE on two datasets.

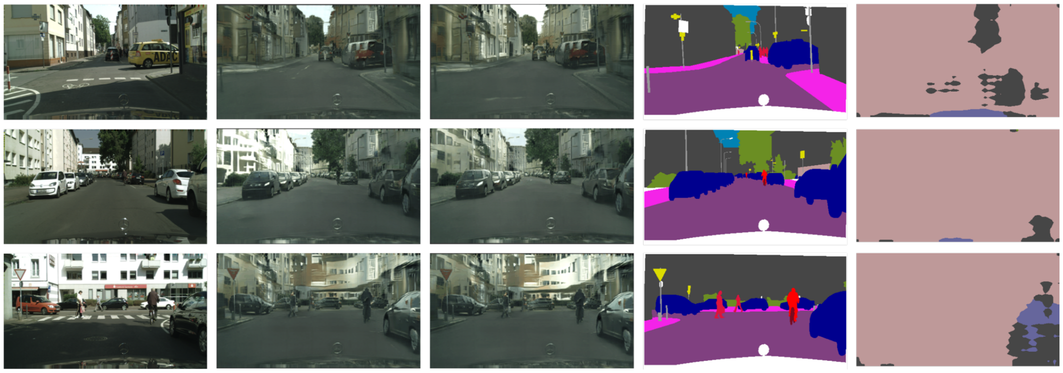


Figure 8: **Variety of Our Unrestricted Adversarial Examples On Cityscapes:** Our model is able to generate various unrestricted adversarial examples which can mislead the state-of-the-art semantic segmentation networks. First column is the real images from Cityscapes dataset. Second and third columns are our stylized unrestricted adversarial examples, the fourth column is the groudtruth label and the final column shows the prediction of our examples on target segmentation network.

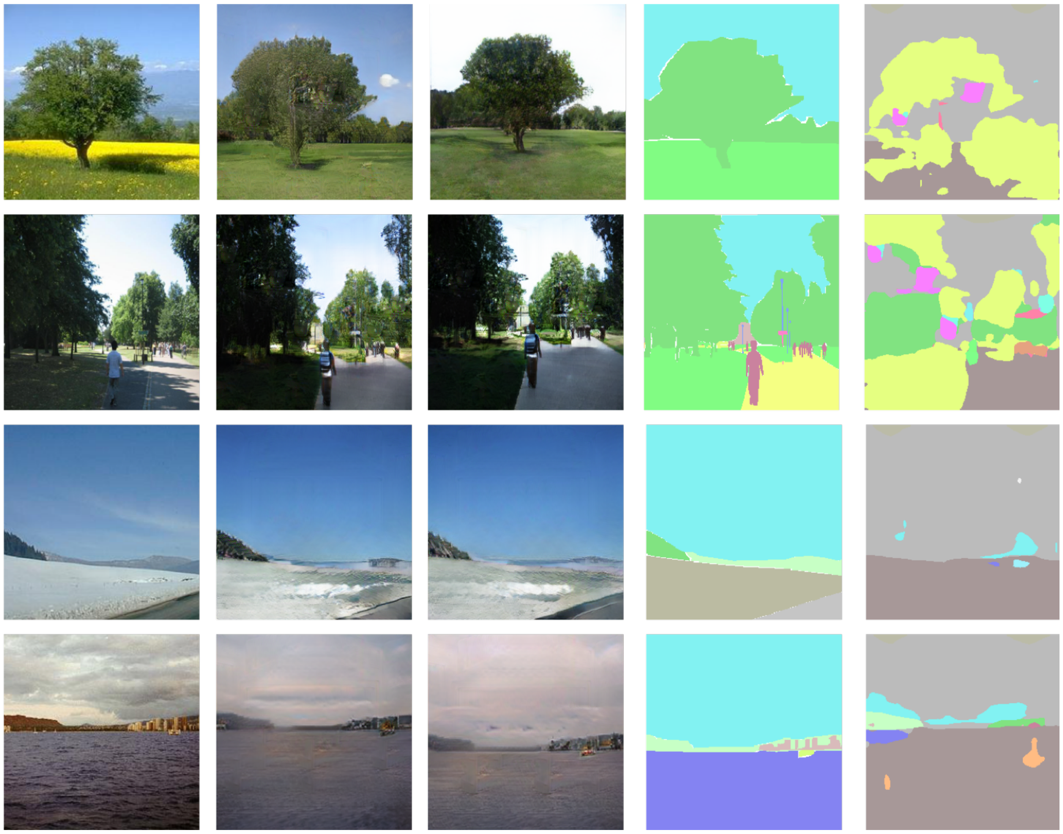


Figure 9: **Variety of Our Unrestricted Adversarial Examples On ADE20K:** Our model is able to generate various unrestricted adversarial examples which can mislead the state-of-the-art semantic segmentation networks. First column is the real images from ADE20K dataset. Second and third columns are our stylized unrestricted adversarial examples, the fourth column is the groudtruth label and the final column shows the prediction of our examples on target segmentation network.

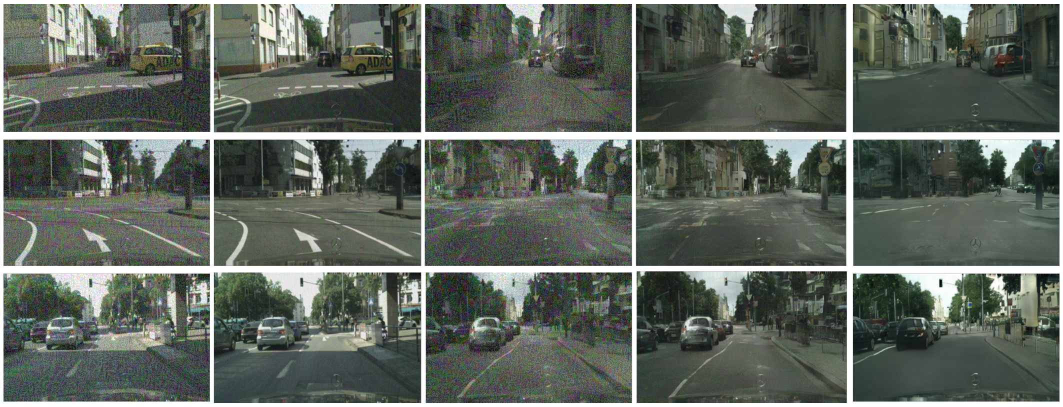
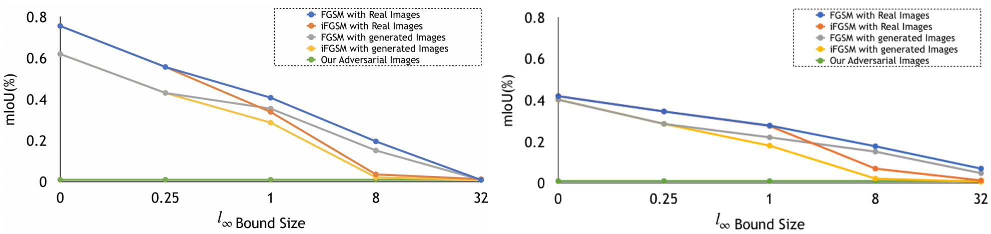


Figure 10: **Comparison of norm-bounded samples and unrestricted adversarial examples on same mIoU level:** First and second columns are adversarial examples generated by FGSM and PGD with $\epsilon = 32$ on real images. Third and fourth columns are adversarial examples generated by FGSM and PGD with $\epsilon = 32$ on vanilla SPADE synthesis images. Last column is unrestricted adversarial examples generated by AdvSPADE.



(a) Cityscapes (b) ADE20K

Figure 11: **Relationship between l_∞ norm bound size and mIoU decline.**

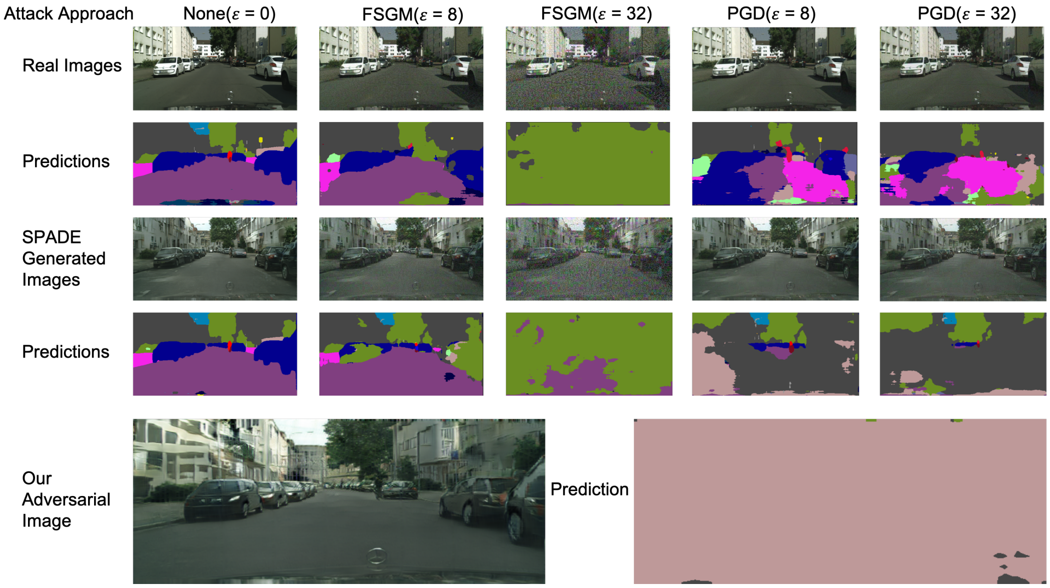


Figure 12: **Visual Comparison on Cityscapes:** We show the visual comparison between our unrestricted adversarial examples and norm-bounded adversarial examples with multiple settings on target segmentation network (DRN-105) on Cityscapes. From left to right, the first and second row show the adversarial examples generated by adding different bound size ($\epsilon = 0, 8, 32$) perturbation on real images with different attack methods (FGSM, PGD) and their corresponding semantic predictions from DRN-105. The third and fourth row follows the same setting except the norm perturbation is added on standard SPADE generated images. The last row shows our unrestricted adversarial example and its prediction.

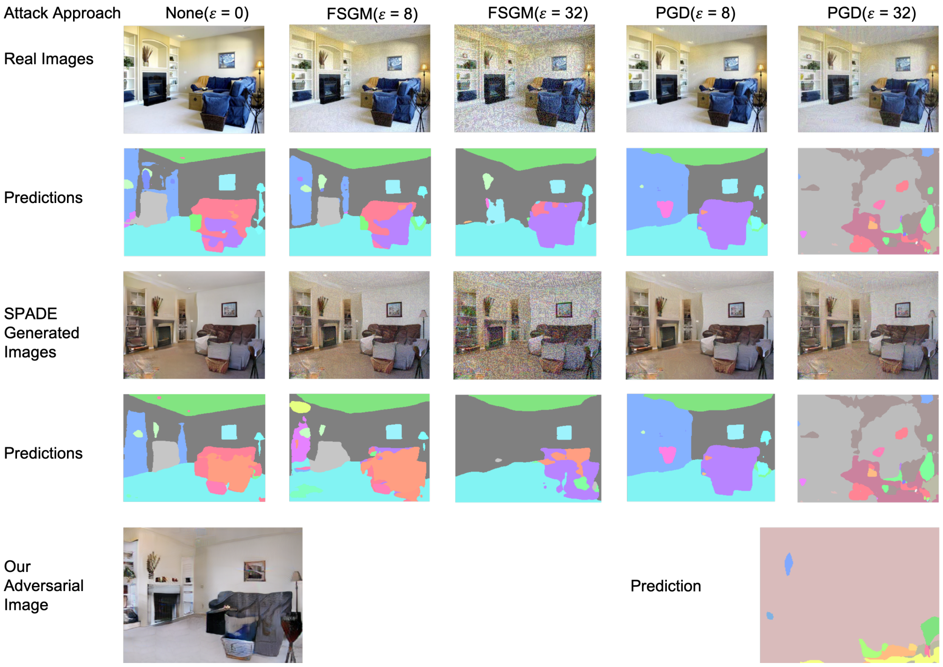


Figure 13: **Visual Comparison between our unrestricted adversarial examples and norm-bounded adversarial examples on ADE20K**

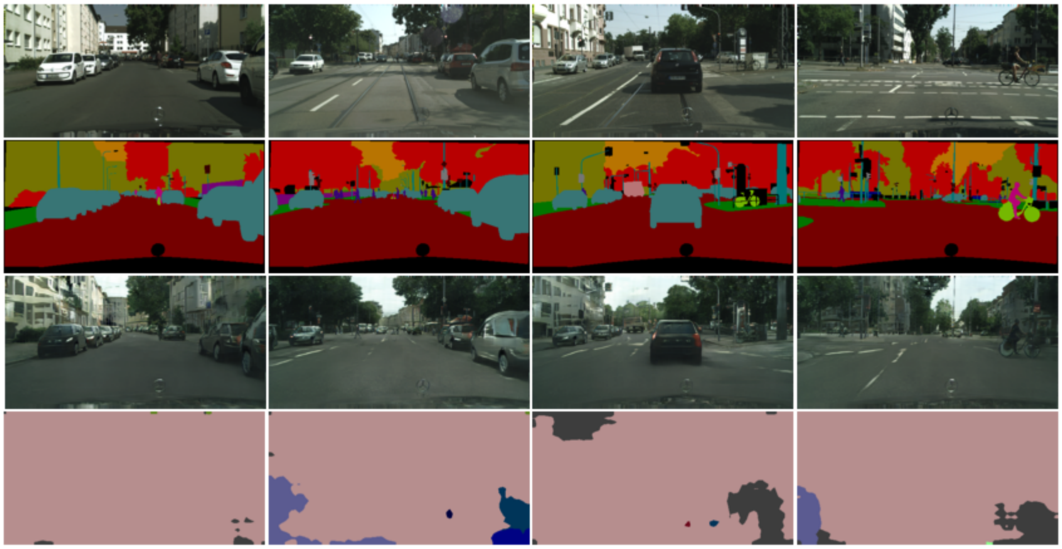


Figure 14: **Visual Results on Cityscapes:** First row is the real images in Cityscapes validation set. Second row is the corresponding groundtruth labels. Third row is our unrestricted adversarial examples and last row is the corresponding segmentation results.

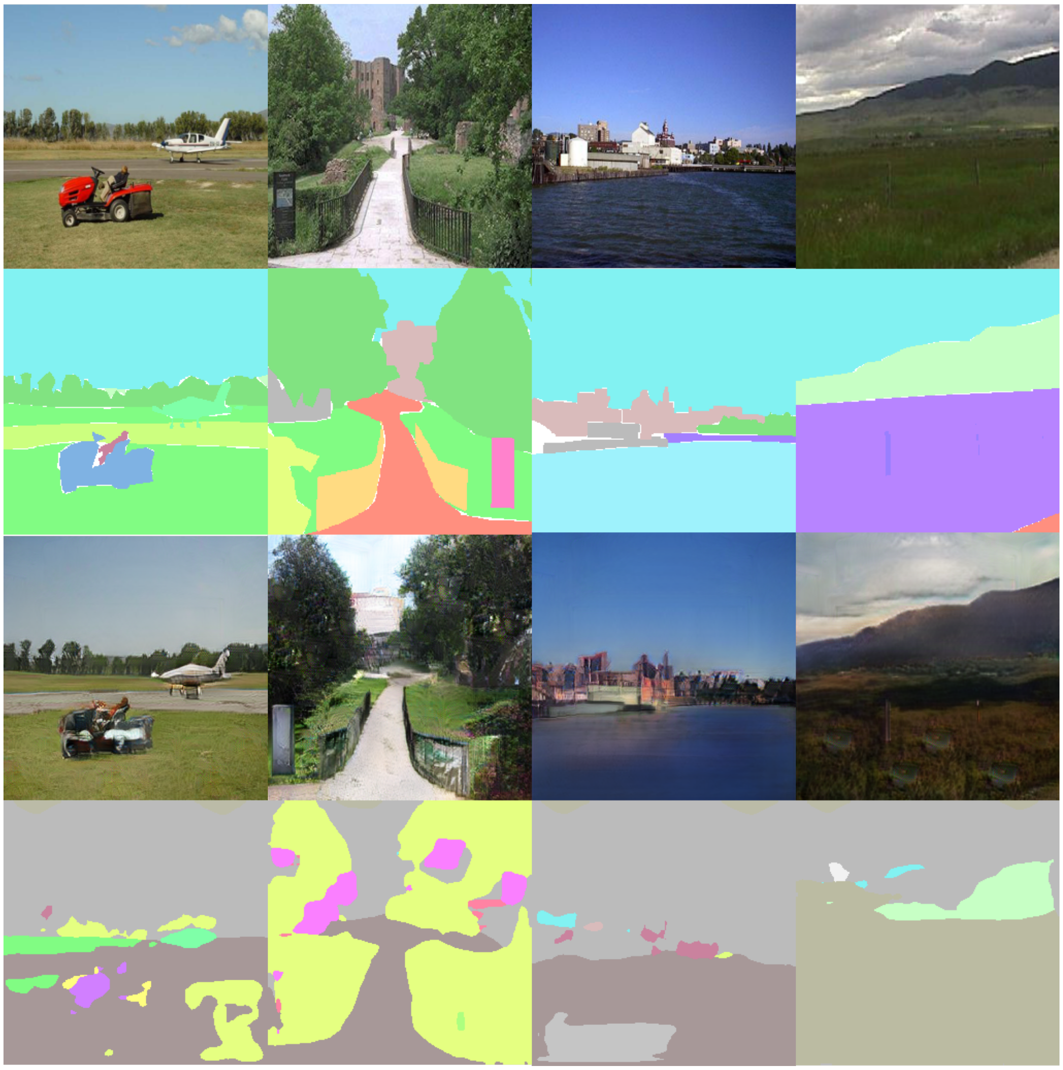


Figure 15: **Visual Results on ADE20K:** First row is the real images in Cityscapes validation set. Second row is the corresponding groundtruth labels. Third row is our unrestricted adversarial examples and last row is the corresponding segmentation results.