UNRESTRICTED CONTROLLABLE ATTACKS FOR SEGMENTATION NEURAL NETWORKS

by

Guangyu Shen

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science

Department of Computer and Information Technology
West Lafayette, Indiana
May 2020

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Baijian Yang, Chair

Department of Computer and Information Technology

Dr. Julia M. Rayz

Department of Computer and Information Technology

Dr. Jin Kocsis

Department of Computer and Information Technology

Approved by:

Dr. Eric T. Matson

Head of the Graduate Program

To my beloved parents and friends for their supports

ACKNOWLEDGMENTS

I sincerely appreciate all members in my thesis committee for their great help and guide. Thank for the encouragement from my parents and all friends in my master careers.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

abbr	abbreviation
GAN	Generative Adversarial Networks
CGAN	Conditional Generative Adversarial Networks
FGSM	Fast Gradient Sign Method
iFGSM	Iterative Fast Gradient Sign Method
PGD	Projected Gradient Descent
mIoU	mean-Intersection of Union
DRIT	Disentangled Representation Image Translation
VAE	Variational Autoencoder

ABSTRACT

Despite the rapid development of adversarial attacks on machine learning models, many types of new adversarial examples remain unknown. Undiscovered types of adversarial attacks pose a serious concern for the safety of the models, which raises the issue about the effectiveness of current adversarial robustness evaluation. Image semantic segmentation is a practical computer vision task. However, segmentation networks' robustness under adversarial attacks receives insufficient attention. Recently, machine learning researchers started to focus on generating adversarial examples beyond the norm-bound restriction for segmentation neural networks. In this thesis, a simple and efficient method: AdvDRIT is proposed to synthesize unconstrained controllable adversarial images leveraging conditional-GAN. Simple CGAN yields poor image quality and low attack effectiveness. Instead, the DRIT (Disentangled Representation Image Translation) structure is leveraged with a well-designed loss function, which can generate valid adversarial images in one step. AdvDRIT is evaluated on two large image datasets: ADE20K and Cityscapes. Experiment results show that AdvDRIT can improve the quality of adversarial examples by decreasing the FID score down to 40% compared to state-of-the-art generative models such as Pix2Pix, and also improve the attack success rate 38% compared to other adversarial attack methods including PGD.

CHAPTER 1. INTRODUCTION

1.1 Scope

Deep Learning has become the core technique in many real-world vision systems, such as autonomous driving (Ess, Mueller, Grabner, & Van Gool, 2009), computer-aided diagnose systems (Milletari, Navab, & Ahmadi, 2016), and facial recognition systems. Due to the inexplicability of the deep learning model, the decision-making process inside the model is opaque to users. The inexplainable decision-making process raises concerns while deploying deep learning models in security-critical systems. Recently, researchers of machine learning security have found that deep learning and machine learning models are fragile to a class of attack methods: Adversarial attack (Szegedy et al., 2013). Adversarial attackers can mislead the model's behavior without disturbing humans by carefully manipulating the model's input.

Machine learning researchers have made strenuous efforts to address the model's security issue under the adversarial attacks. For example, researchers can use adversarial examples to retrain models so that the models can not be easily fooled (Goodfellow et al., 2015). Existed defense techniques can only solve the security vulnerabilities brought from norm-bounded adversarial attacks such as PGD (Madry, Makelov, Schmidt, Tsipras, & Vladu, 2018). Unrestricted attacks still remain a massive concern to current machine learning systems, which inspires researchers to explore more effective and realistic attacks, such as using Wasserstein distance (Wong, Schmidt, & Kolter, 2019)) and realistic image transformations (Engstrom, Tran, Tsipras, Schmidt, & Madry, 2017).

In particular, Song, Shu, Kushman, and Ermon (2018) proposed a conditional-GAN based unrestricted adversarial attack for classification tasks. However, their method can only be applied to low-resolution images because of the limited model capacity. The synthetic images have poor quality. Besides, the generation stage is uncontrollable, which means researchers can not guarantee that the generated example has the features they want. Meanwhile, the image classification tasks are mainly what current studies of adversarial attacks focus on. To solve the above problems, we pay attention to the controllable unrestricted adversarial example generation for image semantic segmentation tasks in this thesis.

1.2 Significance

By constructing robust image segmentation models, deep learning image segmentation techniques can be deployed into fields in a wide range, including security-critical areas. Currently, the most effective defense method is to augment the dataset with adversarial examples. This method is called adversarial training (Madry et al., 2018). Therefore, generating effective and realistic adversarial examples are the prerequisites for constructing robust deep learning models. The significance of this thesis is to discover more powerful and effective adversarial examples for image segmentation models. These adversarial examples can be further used in defense techniques such as adversarial training in the future.

1.3 Research Question

This thesis questions whether the proposed AdvDRIT framework can improve the quality of adversarial examples compared to state-of-the-art attack methods on target semantic segmentation networks. In order to evaluate the quality of generated adversarial examples, we leveraged several valid evaluation metrics including mIoU drop, FID score and attack success rate. To demonstrate the improvement of AdvDRIT, we considered lots of state-of-the-art attack methods as our baseline methods including PGD (Madry et al., 2018), DAG (Xie et al., 2017), Houdini (Cisse, Adi, Neverova, & Keshet, 2017) and AdvGAN (C. Xiao et al., 2018), etc. For the target segmentation network models, we also used the most advanced segmentation models including DRN (Yu, Koltun, & Funkhouser, 2017), Upernet (T. Xiao, Liu, Zhou, Jiang, & Sun, 2018), etc. We introduced the detail of evaluation metrics, baseline models and target models thoroughly in Chapter 4. In conclusion, this thesis questions whether the proposed AdvDRIT framework can improve the quality (mIoU drop, FID and attack success rate) of adversarial examples compared to existed state-of-the-art attack methods(PGD, DAG, Houdini, etc) on target semantic segmentation networks including DRN, Upernet.

1.4 Assumptions

We make the following assumptions in this thesis:

- All images from the dataset are independent and identical distributed.
- All images follow an implicit high dimensional probability distribution.
- All images are generated by several disentangled parts.

1.5 Limitations

The proposed attack method relies on generative models. GAN is used to synthesize realistic adversarial images, which are insensible to humans but vulnerable to deep learning models. The limitation of this research is caused by GAN. First of all, GAN is very difficult to train. Training GAN is a time-consuming and complex process. Over 20 hyper-parameters in the network need to be selected carefully. In the worst case, generated images are artificial, which is useless for adversarial attacks. Secondly, there is a gap in image quality between synthesis images and natural images. Humans can still find some small artifacts in the generated images, which may cause the failure of adversarial attacks. In this thesis, we do not propose a new method to stabilize the GAN training phase.

1.6 Delimitations

We define the delimitation in this thesis. First of all, we only pay attention to the adversarial attacks using generative models. Secondly, we focus on the adversarial attacks on the image semantic segmentation task. Thirdly, we only consider the adversarial attack on the gradient based deep learning neural networks. Fourthly, we only consider the untargeted adversarial attack in this thesis. Last but not least, we only evaluate the proposed method on two image datasets: ADE20K (Zhou et al., 2018) and Cityscapes (Cordts et al., 2016).

1.7 Definitions

Generative Model - A generative model describes the process of generating a data distribution.

GAN - A generative adversarial network is a family of models with certain designed architecture for simulating data distribution.

Adversarial Attack - An adversarial attack consists of subtly modifying a benign image in an adversarial way that the changes are almost undetectable to the human eyes, meanwhile leading to the poor performance of machine learning models.

Semantic Segmentation - Semantic segmentation is a dense visual task for predicting the label information of each pixel in an image.

Disentanglement Representations - Disentanglement representation (Bengio, Courville, & Vincent, 2013) refers to a representation of data in which a change in a single underlying factor of variation leads to a change in a certain feature factor in the learned representation.

White Box Attacks - The attacker knows the inner architectures, gradient and weights of target models and leverages all such information to generate attack samples.

Black Box Attack - The attacker does not know the inner structure and information of target models and leverages other knowledge to attack target models.

Targeted Attack - The attacker generates adversarial examples which are mis-classified as a certain target class.

Untargeted Attack - The attacker generates adversarial examples which are mis-classified as any classes except the correct class.

1.8 Summary

This chapter introduces the research question in this thesis, points out the scope, significance, assumptions, definitions, limitations and delimitations.

CHAPTER 2. REVIEW OF LITERATURE

Controllable unrestricted adversarial examples generation is a comprehensive problem that including several sub-problems in the area of deep learning model security. In order to solve this question properly, a review of the previous work is necessary. This chapter includes the introduction of several core techniques we use to solve the research question in this thesis. Then we present the research gap in the robustness of deep learning models.

As a class of machine/deep learning models, a generative model aims to learn a probability distribution behind real data(image, voice, etc.): P_{data}. Once researchers gain a representative distribution, they can generate new data by sampling from P_{data}. Therefore, we firstly introduce the most popular generative model in the deep learning field: Generative Adversarial Network(GAN) and the variants based on it, such as Conditional-GAN, Pix2Pix, Cycle-GAN, etc. Secondly, we discuss some breakthrough work for the semantic segmentation task. The-state-of-art attacks, defense methods for deep learning models and their flaws are introduced in the third and fourth sections. Last but not least, the authors introduce the definition of disentangled representation.

2.1 Generative Adversarial Nets

Definitions of 5 different Generative Adversarial Nets and their advantages and disadvantages are introduced in this section.

2.1.1 Vanilla GAN

Generative Adversarial Networks have received tremendous success recently. A GAN (Goodfellow et al., 2015) contains two networks: a discriminator network(D) and a generator network(G). G takes a random noise as the input and maps it to natural images. D is required to differentiate real images and G generated images. G and D construct a mini-max two-player game. The loss function of vanilla GAN is as follows:

$$\min_{G}\max_{D} V(G,D) = \mathbb{E}_{x\sim P_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim P_{z}(z)}[\log(1-D(G(z)))] \tag{2.1}$$

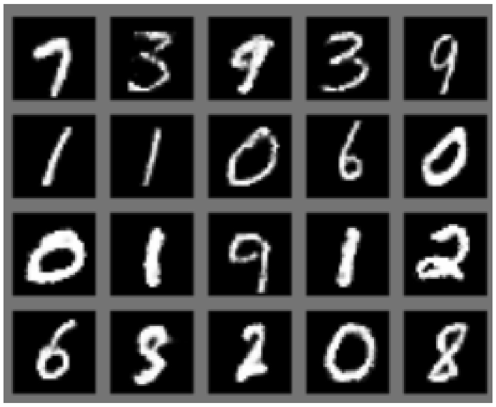

Figure 2.1. Images generated by GAN on MNIST handwritten digit datasets (Goodfellow et al., 2015)

In the Eq 2.1 (Goodfellow et al., 2015), G and D are two neural networks with different architectures. *Z* is a random noise sampled from a probability distribution, such as a Gaussian Distribution. D aims to discriminate whether an input image is from natural or synthesis image distribution. In contrast, G is required to generate data that are able to fool D. By training two networks D and G in turns, researchers can gain a generator G, which can produce realistic images theoretically. The Fig 2.1 (Goodfellow et al., 2015) shows the generated images by vanilla GAN on MNIST dataset (LeCun & Cortes, 2010).

2.1.2 Conditional-GAN

One of the drawbacks of vanilla GAN is that the image generation is random. For instance, from the Fig 2.1, the digits in generated images are random. People can not control the digits in images during the GAN training phase. Mirza and Osindero (2014) proposed Conditional-GAN (CGAN) to address this issue. The main difference between CGAN and vanilla GAN is the generator's input z. In CGAN, the input vector of the G combines random noise and the class label of data researchers want to generate. Therefore, researchers can generate images with different contents by controlling label information in the generator input. The formal definition of CGAN is as follows:

$$\min_{G}\max_{D} V(G,D) = \mathbb{E}_{x\sim P_{data}(x)}[\log D(x|y)] + \mathbb{E}_{z\sim P_{z}(z)}[\log(1-D(G(z|y)))] \tag{2.2}$$

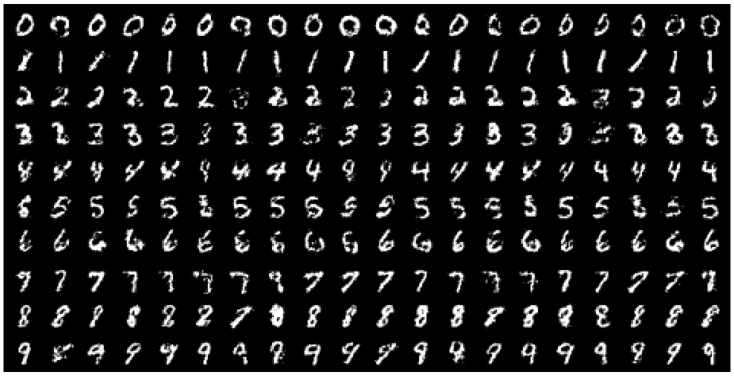

Figure 2.2. Images generated by CGAN on MNIST handwritten digit dataset (Mirza & Osindero, 2014)

In the Eq 2.2 (Mirza & Osindero, 2014), y stands for the label information of corresponding images. In most practical situations, researchers want to control the type of images GAN generates. It is also the reason why Conditional-GAN is more widely used than Vanilla GAN. The Fig 2.2 (Mirza & Osindero, 2014) shows the images generated by CGAN.

2.2 Image-to-Image Translation

With the fast development of GAN, a specific type of vision task emerges: Image-to-Image Translation. I2I translation refers to the task of projecting images between two different domains. I2I Translation has a lot of applications, such as colorization(grey-scale image to RGB image), image segmentation (RGB image to a semantic label) and image generation (semantic label to RGB image). CGAN is widely used in I2I translation tasks. Researchers consider images from domain A as the conditional information in the generator G. The G aims to generate images in domain B conditional on images in domain A. However, both vanilla GAN and Conditional-GAN are only applied to low-resolution image dataset such as MNIST. Image size in MNIST is 28x28. Directly deploying CGAN on a high-resolution real-scene dataset causes a lot of problems. For instance, the generated images' quality is very low. There are lots of noticeable artifacts in the generated images. Besides, the generator is hard to generate legitimate images if the discriminator D can easily distinguish the difference between generated and real images. Therefore, substantial works have been done to propose better CGAN architectures for I2I Translation including Pix2Pix (Isola et al., 2017), Cycle-GAN (Zhu et al., 2017) and SPADE (Park et al., 2019). All of these models are extensions of Conditional-GAN.

2.2.1 Pix2Pix Models

Pix2Pix is one of the earliest models proposed for solving I2I Translation tasks. It learns the domain transfer mapping and a target loss which is applied to train such mapping. Compared to Conditional-GAN, Pix2Pix adds an additional loss item in Generator G objective function. G needs to fool the D. Meanwhile, synthetic images are required to be close to the real images. Besides, both G and D are convolutional neural networks. In order to improve the models' representation ability, authors add more hidden layers in both D and G. The loss function of pix2pix is shown as follows:

$$L_{CGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{z,x}[\log(1-D(x,G(z,x)))] \tag{2.3}$$

$$L_{L1}(G) = \mathbb{E}_{x,y,z}[||y-G(x,z)||_1] \tag{2.4}$$

$$G^* = arg\min_G \max_D L_{CGAN}(G,D) + L_{L1}(G) \tag{2.5}$$

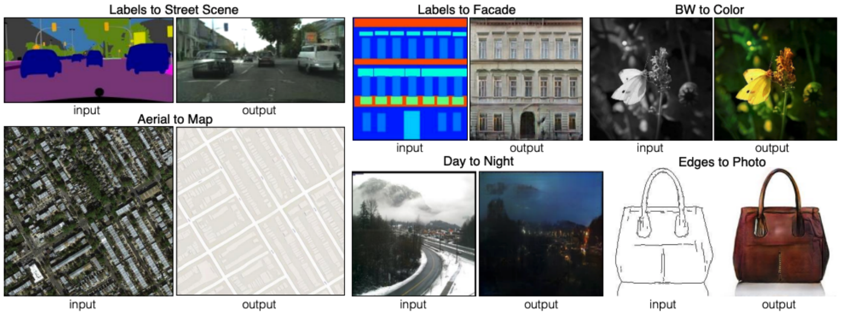

Figure 2.3. Images generated by Pix2Pix (Isola et al., 2017)

In the Eq 2.3 and the Eq 2.4 (Isola et al., 2017), x stands for the real image, y stands for the corresponding semantic label and z means the random noise vector. As shown in the Fig 2.3 (Isola et al., 2017), Pix2Pix can synthesize images with resolution 286x286, which is much larger than the image resolution in MNIST.

2.2.2 Cycle-GAN

Pix2Pix can transfer images between the different domains only if researchers have pair-to-pair images data from two domains. This requirement is unpractical especially when researchers apply such technique in a real-world engineering project. Labeling data is time-consuming. Therefore, researchers only have unaligned images from two domains in real-world scenarios. Cycle-GAN is proposed to address the unaligned images transfer problems in Image-to-Image translation. In this situation, researchers only have two branches of data from two different domains. It is more challenging for a model to learn the translation projection since the input data pair is unaligned. In Cycle-GAN (Zhu et al., 2017), a dual GAN structure is proposed to solve the issue of data unalignment . In CycleGAN, there are two GANs G and F, including two generators and two discriminators. G is tasked to transfer an image from domain X to domain Y and F is inverse. For both GANs, authors add a content-maintain loss item in the objective function of the G to shorten the distance between the original image and the repaired image as shown in the Fig 2.4 (Zhu et al., 2017). Due to such structure, the label information is not required during the training stage. The Fig 2.4 (Zhu et al., 2017) illustrates the structure of CycleGAN. The Fig 2.5 (Zhu et al., 2017) illustrates the images generated by CycleGAN.

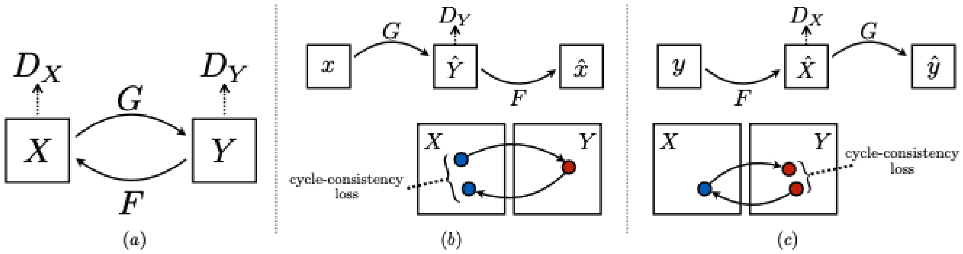

Figure 2.4. CycleGAN Architecture (Zhu et al., 2017)

The target loss function of CycleGAN is as follow:

$$L_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim P_{data}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim P_{data}(x)}[\log(1 - D_Y(G(x)))] \tag{2.6}$$

$$L_{cyc}(G,F) = \mathbb{E}_{x\sim P_{data}(x)}[||F(G(X)) - x||_1] + \mathbb{E}_{y\sim P_{data}(y)}[||G(F(y)) - y||_1] \tag{2.7}$$

$$L_(G,F,D_X,D_Y) = L_{GAN}(G,D_Y,X,Y) + L_{GAN}(F,D_X,Y,X) + \lambda L_{cyc}(G,F) \tag{2.8}$$

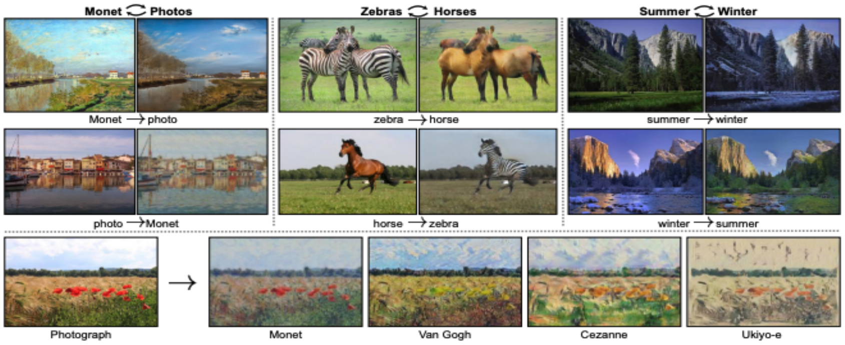

Figure 2.5. Images generated by CycleGAN (Zhu et al., 2017)

2.2.3 SPADE

Spatial Denormalization(SPADE) is the latest I2I translation architecture (Park et al., 2019). It further improves the generated images' quality. In short, SPADE replaces batch normalization layers in G and D networks with a new carefully designed module - SPADE. SPADE regularizes the output of each layer in the 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 stands for the number of channels, x,y represent the width, and the height of the layer output map.)

$$a = \gamma^i_{c,y,x}(m)\frac{h^i_{n,c,y,x} - \mu^i_c}{\sigma^i_c} + \beta^i_{c,y,x}(m) \tag{2.9}$$

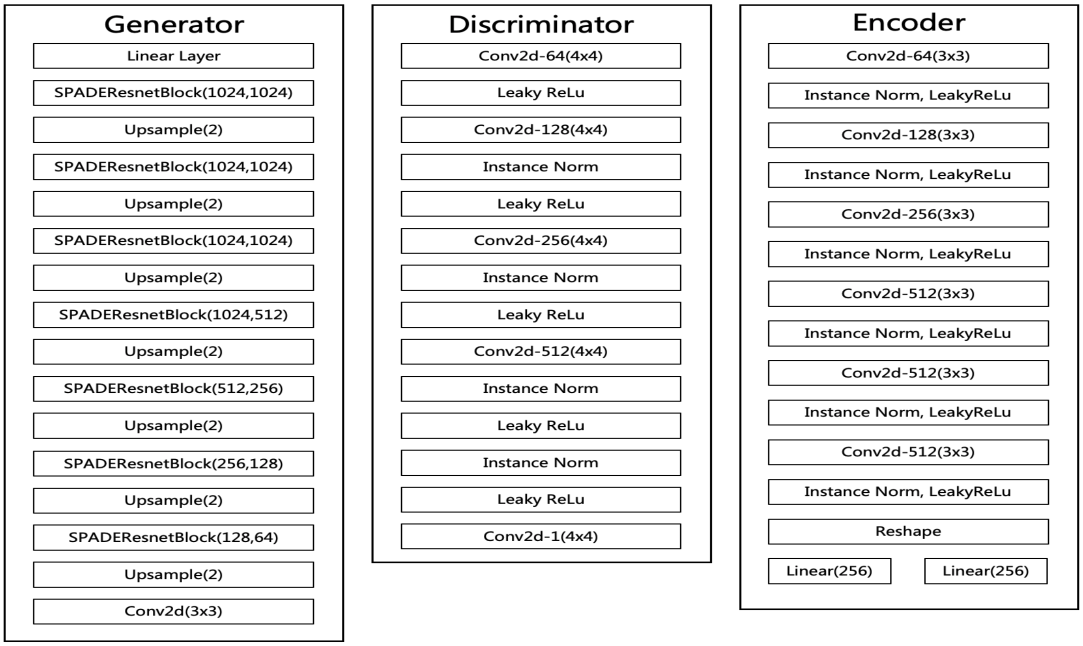

Figure 2.6. Generator,discriminator and encoder architectures (Park et al., 2019)

where μ_c^i and σ_c^i are calculated by:

$$\mu_c^i = \frac{1}{NH^iW^i}\sum_{n,y,x} h_{n,c,y,x}^i \tag{2.10}$$

$$\sigma_c^i = \sqrt{\frac{1}{NH^iW^i}\sigma_{n,y,x}(h_{n,c,y,x}^i)^2 - (\mu_c^i)^2} \tag{2.11}$$

N stands for the batch size. H,W mean the height and the width of the activation map in the corresponding layer respectively.

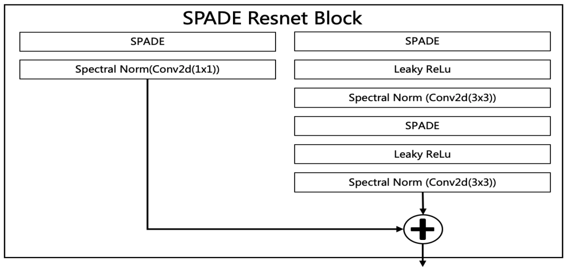
SPADE Resnet Block
SPADE
Spectral Norm(Conv2d(1x1))
SPADE
Leaky ReLu
Spectral Norm (Conv2d(3x3))
SPADE
Leaky ReLu
Spectral Norm (Conv2d(3x3))

Figure 2.7. SPADE ResBlock Architecture (Park et al., 2019)

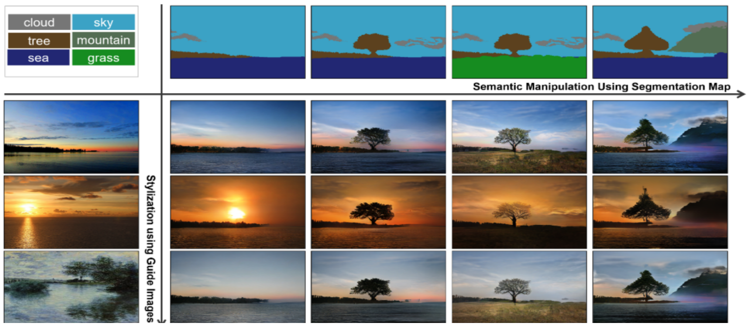
cloud
sky
tree
mountain
sea
grass
Semantic Manipulation Using Segmentation Map
Stylization using Guide Images

Figure 2.8. Generated Images by SPADE (Park et al., 2019)

Encoder part is unnecessary for the generator due to SPADE module. The simplified lightweight network takes the semantic label and a random vector as input. After going through alternate SPADE modules and upsampling layers, SPADE can generate high-quality realistic images. For the discriminator structure, SPADE provides an easier and straightforward way to synthesize multi-modal realistic images. The Fig 2.6 (Park et al., 2019) and the Fig 2.7 (Park et al., 2019) show the implementation details of the generator and the discriminator.

2.3 Image Semantic Segmentation

Image semantic segmentation is a well-applied and critical vision task, which requires detail information from the model output (Barrow & Tenenbaum, 1981). Semantic segmentation is the core technique in multiple vision-related applications, such as autonomous driving (Ess et al., 2009) and computer-aided diagnose system (Ronneberger, Fischer, & Brox, 2015), etc.

Researchers proposed a lot of model architectures to achieve more and more accurate results for the segmentation task (Long, Shelhamer, & Darrell, 2015; Ronneberger et al., 2015; T. Xiao et al., 2018; Yu et al., 2017). Generally, a segmentation model contains two sub-modules: an encoder *E* and a decoder *D*. E aims to extract the features and D aims to restore the output dimension. Although the structural design of segmentation networks have been well-studied, very few studies (Arnab, Miksik, & Torr, 2018; Xie et al., 2017) focus on the robustness of segmentation networks under adversarial attacks.

2.4 Adversarial Attack

Adversarial samples are images that carefully constructed to fool machine learning models, while still perceived the same by the humans. Adversarial perturbations can cause a deep learning model to misbehave when added on a benign image. Most of the researches for adversarial attacks pay attention to the norm bounded attack presented by FGSM (Goodfellow et al., 2015). FGSM constructs adversarial perturbations by maximizing the loss of network w.r.t the model's input. Let L denote the loss function in the neural network, f stands for the network model and θ and y represent all parameters in the network and groud-truth labels respectively. Fast Gradient Sign Method can be denoted as follows:

$$x^{adv} = x + \varepsilon \cdot sign(\nabla_L(f(x;\theta),y)) \tag{2.12}$$

The generated adversarial example is bounded by parameter ε. Note that it is the untargeted version of FGSM. To further increase the possibility that the target network is fooled by adversarial examples, Kurakin, Goodfellow, and Bengio (2016) extended the Fast Gradient Sign Method in an iterative way:

$$x_0^{adv} = x, x_{N+1}^{adv} = Clip_{x,\varepsilon}\{x_N^{adv} + \alpha \cdot sign(\nabla_L(f(x;\theta),y))\}. \tag{2.13}$$

Clip operation ensures that adversarial examples are in the range of $[x+\varepsilon, x-\varepsilon]$ after each iteration. In recent years, researchers have proposed multiple adversarial attack methods for the image classification task (Carlini & Wagner, 2017; Madry et al., 2018). Hand-crafted metrics, such as L_p norm bound (Feinman, Curtin, Shintre, & Gardner, 2017) and Wesstrasien distance (Wong et al., 2019), are leveraged to preserve the adversarial examples' semantic meaning. Several researchers focused on the adversarial attack for segmentation neural networks. Xie et al. (2017) proposed an adaptive attack method on segmentation models: Dense Adver-sary Generation(DAG), which can successfully attack deep segmentation and detection networks. However, DAG still generates small perturbations based on certain norm distances. Cisse et al. (2017) proposed Houdini loss to make loss function of segmentation neural networks to be differential. Song et al. (2018) proposed a GAN-related attack technique to construct unrestricted attack samples. Researchers used a type of GAN: Auxiliary Classifier-Generative Adversarial Network (*AC-GAN*) (Odena, Olah, & Shlens, 2016) to generate adversarial handwritten digits images beyond any norm bound. Existed GAN-based attack methods (Song et al., 2018; Wang, He, & Hopcroft, 2019) adopt two-step procedures that involve many steps of gradient descent on the second part.

2.5 Defense Methods

Adversarial training (Goodfellow et al., 2015; Hinton, Vinyals, & Dean, 2015; Madry et al., 2018; Szegedy et al., 2013) is the most promising method for training robust classifiers currently. Besides, input transformation defense methods on semantic segmentation networks, including rescaling, image compression and Gaussian filtering are evaluated by Arnab et al. (2018). However, all defense methods above rely on obfuscated gradients rather than bringing true robustness to deep learning models (Athalye, Carlini, & Wagner, 2018).

2.6 Disentangled Representations

Disentangled representation targets to model each component in a data distribution. Previous researches leveraged data and label to disentangle features into class-related and class-independent factors (Cheung, Livezey, Bansal, & Olshausen, 2014; Kingma, Rezende, Mohamed, & Welling, 2014; Makhzani, Shlens, Jaitly, Goodfellow, & Frey, 2015; Mathieu, Zhao, Sprechmann, Ramesh, & LeCun, 2016). Recently, unsupervised disentangled representation methods have been explored (X. Chen et al., 2016; Denton et al., 2017). Lee et al. (2018) firstly brought disentangled representation into the I2I translation task. The author introduced DRIT (Disentangled Representation Image Translation) framework to transfer images between different domains in a disentangled way. The author assumed that images from two different domains shared the same content feature space and had independent attribute feature space. The authors gained content features and attribute features from separate encoder networks and fed learned features into a generator to generate images in the target domain.

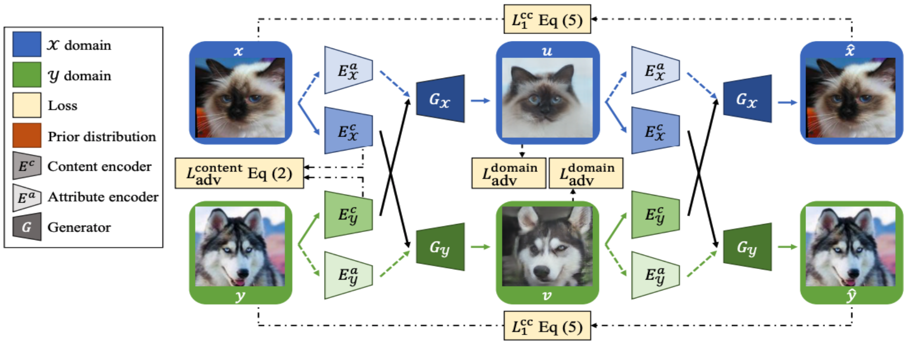

Figure 2.9. DRIT Architecture (Lee et al., 2018)

The Fig 2.9 (Lee et al., 2018) illustrates the framework of DRIT in the Image-to-Image translation task. In this example, researchers want to process the image translation between cat and dog images. In consideration of an one-direction translation(cat to dog), the input cat image x is encoded to cat attribute feature space and content feature space by E_x^a and E_x^c respectively. The input dog image y is encoded to dog attribute feature space and content feature space by E_y^a and E_y^c respectively. Then, the generator G_y takes encoded content feature E_x^c and dog attribute feature E_y^a as the input and generate the transfer dog image v.

CHAPTER 3. RESEARCH METHODOLOGY

In this chapter, AdvDRIT is proposed to construct controllable adversarial examples for semantic segmentation. Firstly, we introduce the data collection in this project. Then, we propose an additional well-designed loss item for vanilla DRIT framework. After that, we show that the modified DRIT framework is able to generate unrestricted adversarial examples for semantic segmentation. Then, we leverage the AdvDRIT framework to learn the disentangled representation to make the generation stage controllable. Finally, we describe the experimental design which can prove the effectiveness of the proposed AdvDRIT. Besides, authors also show the experimental design for robust training by considering proposed unrestricted controllable examples.

3.1 Image Data Set

Well-labeled data is critical to deep learning models. Training networks need tons of image data. For example, a standard deep learning network needs over 10000 images for the training and inference. It is impractical to collect data personally. Nowadays, due to the vast need of the well-labeled image data, big IT companies or institutions such as Microsoft and Google, have released several public image datasets for the convenience of researchers. An image dataset for deep learning/ machine learning contains two parts: train set and test set. For the data in the train set, it has label information, such as classes, bounding box locations. During the training stage, researchers need to minimize the loss function between the models' prediction and label to improve the model's accuracy. For the test set, image data does not have the corresponding label. In this thesis, we select two large scale image segmentation datasets: ADE20K (Zhou et al., 2018) and Cityscapes (Cordts et al., 2016) to train and evaluate our method. ADE20K is a scene parsing image dataset, which contains over 20000 scenes images well labeled with objects. In detail, train set contains 20000 images, and 2000 images in the test set. There are 150 classes including stuff like cloud, wall, tree, and discrete objects like people, car, lamp. Cityscapes is a road view dataset that includes recorded videos and images in street scenes from 50 cities and regions with semantic annotations. There are over 3000 images in the train set and 500 images in the test set.

3.2 Generating Unrestricted Adversarial Examples

In this section, we introduce a well-designed additional loss item: Unrestricted Adversarial Loss, which is able to encourage CGAN architectures to generate unrestricted adversarial examples.

3.2.1 Unrestricted Adversarial Loss

An adversarial loss term was proposed for generating adversarial examples beyond the norm-bound restriction. At a high level, the generator is aimed to fool the target semantic segmentation neural network. Under such training strategy, the generated images need to fool both the discriminator and target segmentation network. We embedded the target model into the image generation framework and encouraged the framework to maximize the target model's loss during the training stage. Let S denote the target semantic segmentation neural network, G denote the image generator, y denote the semantic label and z as the input random vector. The Unrestricted Adversarial Loss is defined as follows:

$$L_{ATK} = -\mathbb{E}_{z \sim P_z(z)} \log f(S(G(z|y)), y) \tag{3.1}$$

Dice Loss (Sudre, Li, Vercauteren, Ourselin, & Jorge Cardoso, 2017) f was used in the Eq 3.2.1. A natural image I is fed into the encoder E and the outputs are a mean variable $\mu_{E(I)}$ and a variance variable $\sigma_{E(I)}$. Then, we reparameterize the random noise z (Kingma & Welling, 2013).

$$z = \mu_{E(I)} + \sigma_{E(I)} \cdot \varepsilon, \varepsilon \sim \mathscr{N}(0, I) \tag{3.2}$$

3.3 Generating Controllable Adversarial Examples

In order to generate controllable adversarial examples. Authors applied the disentanglement to learn the disentangled representation of input data.

3.3.1 Disentangled Representations

Inspired by Lee et al. (2018), we disentangled an image to domain-adversary and domain-benignity representations and considered adversarial examples generation as an I2I translation problem: transferring an image in a benign field to an adversarial field. We assumed that each image can be disentangled into two independent feature spaces: adversarial and benign feature space. Features in benign space are innocuous and hard to manipulate to make images adversarial. In detail, manipulating features in benign space may cause the generated adversarial images unnatural or to contain many noticeable artifacts. The discriminator aims to distinguish whether a feature is from adversarial space or benign space. During the training phase, authors only manipulate features in the adversarial space, by feeding modified adversarial features and benign features into a generator, authors are able to generate adversarial examples. The designed architecture is shown in the Fig 3.1.

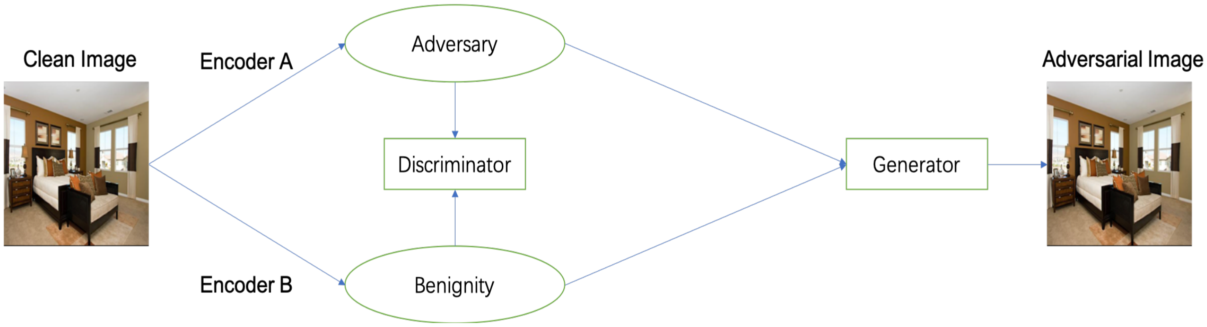

Figure 3.1. Disentangled Adversarial Generation Workflow

In total, there are 10 networks in our framework. The target of this project is to generate adversarial examples ($A \in R^{H*W*3}$) from natural clean images ($N \in R^{H*W*3}$) in the disentangled way. As shown in the Fig 3.1, the framework contains content encoders $\{E_N^c, E_A^c\}$, feature encoders $\{E_N^f, E_A^f\}$, generators $\{G_N, G_A\}$, and domain discriminators $\{D_N, D_A\}$, a content discriminator D_{adv}^c, and a target segmentation network S. In practice, the content encoder E_N^c maps natural clean image N to a content feature space ($E_N^c : N \to C$) and the feature encoder E_A^f projects image to a domain-independent feature space ($E_A^f : A \to F_a$). The generator G_N will generate an adversarial example conditioned on both content and feature variables ($G_N : \{C, F_a\} \to A$).

In order to achieve the representation disentanglement, we followed the same method suggested in Lee et al. (2018): a content discriminator and network weight sharing. Authors used the same parameters in the last layer of E_N^c and E_A^c. We also used the same parameters in the first layer of G_N and G_A. By doing this operation, the content features can to be projected to the same space. Besides, the content discriminator D_{adv}^c aims to distinguish the difference between encoded content features z_N^c and z_A^c. If the well-trained discriminator can not distinguish the above content features, we can guarantee that the two content encoders successfully project images from two different fields to the same feature space. In formal, the content adversarial loss can be written as follows:

$$\begin{aligned}\mathscr{L}_{adv}^{content}(E_N^c, E_A^c, D_{adv}^c) = \mathbb{E}_n[\frac{1}{2} log D_{adv}^c(E_N^c(n)) + \frac{1}{2} log(1 - D_{adv}^c(E_N^c(n)))] \\ + \mathbb{E}_a[\frac{1}{2} log D_{adv}^c(E_A^c(a)) + \frac{1}{2} log(1 - D_{adv}^c(E_A^c(a)))]\end{aligned} \tag{3.3}$$

Instead of content adversarial loss, authors also leverage several loss functions to stabilize the framework training process: Domain adversarial loss and KL loss.

Domain adversarial loss L_{adv}^{domain} is used to stimulate generators to generate high-quality images. In detail, D_N and D_A aim to differentiate natural images and synthetic adversarial images and G_A and G_N aim to generate realistic images. L_{adv}^{domain} can be written as follows:

$$\begin{aligned}\mathscr{L}_{adv}^{domain}(D_N, D_A, G_N, G_A) = \mathbb{E}_{n,a}[log D_A(a) + log(1 - D_A(G_A(n)))] \\ + \mathbb{E}_{n,a}[log D_N(n) + log(1 - D_N(G_N(a)))]\end{aligned} \tag{3.4}$$

KL loss is used to force the feature vector F_a to be close to a probability distribution under the KL divergence measurement which is helpful to perform stochastic sampling in the test phase. In this thesis, authors used a standard Gaussian distribution $N(0,1)$.

$$\mathscr{L}_{KL} = \mathbb{E}[D_{KL}((f_a)||N(0,1))], where\ D_{KL}(p,q) = -\int(p(z) log \frac{p(z)}{q(z)} dz) \tag{3.5}$$

After combining the above loss items with our designed unrestricted adversarial loss 3.2.1, the whole objective function of the proposed AdvDRIT is:

$$\min_{G,E^c,E^f} \max_{D,D^c} \lambda_{adv}^{content} \mathscr{L}_{adv}^{content} + \lambda_{adv}^{domain} \mathscr{L}_{adv}^{domain} + \lambda_{KL} \mathscr{L}_{KL} + \lambda_{ATK} \mathscr{L}_{ATK} \tag{3.6}$$

Appropriate hyper-parameters are crucial to the experiment results. In this thesis, authors leverage a two-step method to gain suitable hyper-parameters for each loss item. Grid search (Pedregosa et al., 2011) is a popular method to optimize the hyper-parameters. However, training AdvDRIT is time-consuming. It's impractical to apply grid search in a wide range. Instead, we optimized the hyper-parameters in two steps. In the first step, we manually adjusted the value of each λ to make the value of each loss item in the same order of magnitude. It is because each loss item should have the similar contribution to the total loss. After manually adjustment, we applied a simple grid search around the value of hyper-paramaters in the first step. For each hyper-parameter λ, authors applied grid search in a sub-region $\{\lambda$-5,λ+5$\}$ with step size 2.5. We introduce the value of each hyper-parameter after the optimization in the next chapter.

CHAPTER 4. EXPERIMENTS AND RESULTS

We designed three main experiments to thoroughly evaluate our proposed AdvDRIT on modern deep learning segmentation models. Firstly, we compared AdvDRIT with traditional norm-bound attack methods on several popular state-of-the-art segmentation models such as DRN-105 ,DRN-38,DRN-22 (Yu et al., 2017) and Upernet-105 (T. Xiao et al., 2018), on two datasets (ADE20K,Cityscapes). We leveraged the attack success rate, mIoU descending and FID score to evaluate the effectiveness of all attack methods under the same setting. Second, we applied our method on robust models defended by traditional norm-bounded adversarial examples to show the strength of our adversarial examples. Third, we used our controllable unrestricted adversarial examples to robust train segmentation models and test the model's robustness against various adversarial attacks.

4.1 Experimental Set-up

4.1.1 Model Training Details

Following Park et al. (2019), we applied the Spectral Norm (Miyato, Kataoka, Koyama, & Yoshida, 2018) in all layers in the generator and the discriminator. On Cityscapes, we trained AdvDRIT 70 epochs. On ADE20K, we trained AdvDRIT 120 epochs. The learning rate is 0.0002 for the generator and the discriminator. Authors applied the ADAM (Kingma & Ba, 2014) optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.9999$. After hyper-parameters optimization, we set $\lambda_{adv}^{content} = 1$, $\lambda_{adv}^{domain} = 11$, $\lambda_{KL} = 0.01$ on two evaluation dataset, and $\lambda_{ATK} = 10$ for Cityscapes, $\lambda_{ATK} = 70$ for ADE20K in the Eq 3.6. We used a workstation with Linux Ubuntu 18.04 and a single NVIDIA GTX 2080Ti GPU for all experiments in this thesis.

4.1.2 Evaluation Metric

Based on the intuition that a successful attack should mislead most of the prediction in a semantic segmentation prediction map, authors proposed the following evaluation metric: Denote $\mathscr{I}^{H\times W\times C}$ be a set in which all images with W width, H height and number of channels C. Denote $\mathscr{L}^{H\times W}$ be the semantic labels set for images in $\mathscr{I}$. Assume a oracle $o : \mathscr{O} \subseteq \mathscr{I} \to \mathscr{L}$ is able to project all images from its own field $\mathscr{O}$ to $\mathscr{L}$ correctly. $\mathscr{O}$ stands for a set including every image looking realistic from a human perspective. A segmentation neural network $\mathscr{S} : \mathscr{I} \to \mathscr{L}$ is able to generate dense output for all images in $\mathscr{I}$. Authors evaluated two different classes of adversarial examples: Firstly, an unrestricted adversarial example x should fulfill the requirements: $x \subseteq \mathscr{O}$, $\frac{\sum_{i,j}(o_{i,j}(x) \neq \mathscr{S}_{i,j}(x))}{H\cdot W} > \theta$. Secondly, Assume a constant value $\varepsilon > 0$ and a mathematical norm $||\cdot||$, a restricted adversarial example x should fulfill the requirements: $x \subseteq \mathscr{O}$, $\exists x' \subseteq \mathscr{O}\ ||x - x'|| < \varepsilon$, $\frac{\sum_{i,j}(o_{i,j}(x) \neq \mathscr{S}_{i,j}(x))}{H\cdot W} > \theta$.

$o_{i,j}(x), \mathscr{S}_{i,j}(x)$ mean the predicted classes given by oracle o and network $\mathscr{S}$ at location $x(i,j)$. θ is a hyper-parameter ranged in [0,1]. In this thesis, $\theta = 0.95$. In the Fig 4.2, we also showed the relation between attack success rate and different θ.

Instead of attack success rate, we applied two metrics to evaluate the quality and effectiveness of AdvDRIT: Mean Intersection-over-Union (mIoU) and Fréchet Inception Distance (FID). mIoU is a standard evaluation metric in semantic segmentation task which evaluates the network prediction accuracy over all classes. In the attack scenario, lower mIoU score stands for more valid adversarial examples. We also applied FID (Heusel, Ramsauer, Unterthiner, Nessler, & Hochreiter, 2017) to calculate the generated adversarial examples' quality. FID computes the distributional distance between the two different data manifolds. In particular, authors computed the FID between synthesis adversarial images and natural images from dataset. Small FID means adversarial examples are close to natural images and have good qualities.

4.1.3 Baseline Models

Authors compared AdvDRIT attack with traditional attacks in two ways: (1) natural images adding perturbation and (2) GAN-generated benign images adding perturbation. For (2), authors generated benign images with vanilla DIRT and then applied traditional attack methods to generate restrict perturbation. On cityscapes, authors chose DRN-D-105 as the target model. On ADE20K, authors selected Upernet-101. Besides, authors also selected some open-source advanced segmentation models to measure the transferability of AdvDRIT under the black-box manner. On cityscapes, we selected DRN-38, DRN-22 (Yu & Koltun, 2016; Yu et al., 2017), DeepLab-V3 (L.-C. Chen, Papandreou, Schroff, & Adam, 2017) and PSPNet-34-8s (Zhao, Shi, Qi, Wang, & Jia, 2017). On ADE20K, we selected PPM-18, MobilenetV2, Upernet-50 and PPM-101 (Zhou et al., 2018).

Authors compared AdvDRIT with two attacks using GAN (Song et al., 2018; C. Xiao et al., 2018). For AdvGAN, authors replaced the cross-entropy loss to dice loss which is consistent with AdvDRIT. We implemented AdvGAN with two hyper-parameter values ($\lambda_{adv} = -1, -100$) Experimental results are shown in the Table 4.5.

We implemented Song's method (Song et al., 2018) on DIRT. The input random noise z dimension is 256. Authors ran DIRT with 70 epochs and then ran 70 epochs to optimize adversarial loss. Authors also compared the transferability between AdvDRIT and Song's unrestricted attack with the black-box setting. The results can be seen in the Table 4.1.

Authors also considered DAG (Xie et al., 2017) and Houdini (Cisse et al., 2017), two adversarial attacks for segmentation tasks as baseline methods on Cityscapes. For DAG, we set hyper-parameter $\gamma = 0.5$ and number of iteration $N = 200$ as Xie et al. (2017) suggested in their paper. For Houdini, authors replaced the loss function to surrogate Houdini loss in PGD attack with setting l_∞ norm bound size $\varepsilon = 8$, number of iteration $N = 300$. Since Houdini and DAG are both white-box attack methods, authors also chose DRN-D-105 (Yu et al., 2017) for Cityscapes as target segmentation network for a fair comparison.

4.2 Experiment Results

4.2.1 Evaluating Generated Adversarial Images.

Authors compared the mIoU score and FID score between AdvDRIT adversarial examples and other types of images including natural real images and vanilla DRIT generated benign images. The Table 4.2 and the Table 4.1 illustrated the experiment results.

Table 4.1. *The mIoU metric of the AdvDRIT*

	Cityscapes					ADE20K			
Attack Methods	Target Model	Natural Image	DRIT	Song's Attack	AdvDRIT	Target Model	Natural Image	DRIT	AdvDRIT
White-box	DRN-105	0.756	0.618	0.465	0.010	Upernet-101	0.424	0.410	0.009
black-box	DRN-38	0.714	0.551	0.520	0.307	MobilenetV2	0.348	0.317	0.112
	DRN-22	0.68	0.526	0.489	0.257	PPM-18	0.340	0.362	0.009
	DeepLab-V3	0.68	0.54	0.501	0.302	Upernet-50	0.404	0.395	0.092
	PSPNet-34-8s	0.691	0.529	0.495	0.321	PPM-101	0.422	0.409	0.089

The Table 4.1 showed that AdvDRIT adversarial images can make a huge drop on mIoU score under white-box attack setting which indicated the effectiveness of AdvDRIT. In detail, AdvDRIT can make mIoU drop to 0.01 on cityscapes and 0.009 on ADE20K. Even under the transfer-based black-box setting, AdvDRIT adversarial examples can also make a non-trivial mIoU drop, which indicated the transferability of advDRIT examples between different architectures. For examples, AdvDRIT can drop the mIoU score of DRN-22 down to 0.257 on Cityscapes. Note that Song's attack can only drop mIoU down to 0.489 under the same setting.

Table 4.2. *FID score of AdvDRIT and several advanced GAN models*

Model Dataset	Vanilla DRIT	Pix2PixHD	CRN	**AdvDRIT**
Cityscapes	61.93	95.0	104.7	**63.32**
ADE20K	32.2	81.8	73.3	**43.59**

According to the Table 4.2, we noticed that the FID of AdvDRIT adversarial images only raised mildly compared to vanilla DIRT (61.32 to 63.32 on Cityscapes, 32.2 to 43.59 on ADE20K). Compared to other advanced generative models like Pix2PixHD and CRN, AdvDRIT achieved lower FID score on both datasets. The result showed that AdvDRIT adversarial images have comparable quality than other advanced image generation models.

The Fig 4.3 showed some AdvDRIT adversarial images. The first row showed the original real images. The second row showed the corresponding semantic labels. The third row showed the adversarial examples generated by AdvDRIT. The last row showed the prediction results of adversarial examples. We noticed that adversarial examples have similar semantic meaning with original real images. However, the prediction results were completely different with ground-truth semantic labels. It indicated that AdvDRIT generated adversarial examples can successfully mislead the target segmentation models.

Table 4.3. *White-Box Attack Success Rate*

DRN-105				Upernet-101			
Attacks	Pert Size (ε)	Natural Images +Pert	DRIT +Pert	Attacks	Pert Size (ε)	Natural Images +Pert	DRIT +Pert
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%
AdvDRIT	**89.2%**			**AdvDRIT**	**77.1%**		

We compared the attack success rate of AdvDRIT with FGSM and PGD (Goodfellow et al., 2015) on Cityscapes and ADE20K. The size of l_∞ norm bound ε is equal to $0.25, 1, 8, 32$ for both traditional attacks. For PGD, we followed the setting from Arnab et al. (2018). The number of attack iterations is computed by $min\{\lfloor \varepsilon + 4 \rfloor, \lceil 1.25\varepsilon \rceil\}$. Authors used PGD and FGSM to generate perturbation and added them to natural and synthetic images from vanilla DIRT. Then, we compared the their mIoU and FID scores with AdvDRIT. The Table 4.3 indicated that traditional attacks(FGSM, PGD) can not successfully attack target networks when bound size is limited ($\varepsilon = 0.25, 1, 8$). For instance, FGSM attack can only get 0% attack success rate when bound size $\varepsilon = 1$ on both benchmarks for target models. However, AdvDRIT was able to successfully attack the target model with high attack success rate (89.2% and 77.1% on target models respectively).

The Table 4.4 showed the mIoU and FID score comparison between AdvDRIT and norm-bounded attacks(FGSM and PGD). For the norm-bounded attacks, authors considered two different settings: generating perturbation based on real images and generating perturbation based on Vanilla DRIT generated clean images. Since FID score can only be used to evaluate the quality of generated images, authors did not report FID score of natural images with perturbation. The FID score can be seen from the brackets in the Table 4.4.

Table 4.4. *Attack Effectiveness*

DRN-105				Upernet-101			
Attack	Pert Size (ε)	Natural Image +Pert	DRIT +Pert	Attack	Pert Size (ε)	Natural Image +Pert	DRIT +Pert
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)
AdvDRIT		**0.01 (61.93)**		**AdvDRIT**		**0.09 (43.59)**	

The Table 4.4 revealed that traditional norm-bound attack methods such as FGSM, PGD, required large distortion to generate valid adversarial examples for segmentation networks. Since the distortion is so large, the quality of adversarial examples can not be guaranteed and humans can differentiate adversarial examples and benign images easily. High FID score showed the decline on the quality of traditional adversarial images. Meanwhile, PGD and FGSM adversarial examples can not decrease mIoU as much as AdvDRIT if keeping the high image quality. For instance, on Cityscapes, vanilla DIRT synthetic images with bound size $\varepsilon = 1$, the FID which with 64.455 is comparable with AdvDRIT samples. However, their mIoU score (0.355) is much larger than AdvDRIT examples (0.01).

The Fig 4.1 showed AdvDRIT samples and traditional images when their mIoU score is comparable (around 0.01). Authors can simply notice the noise in traditional adversarial samples instead of in AdvDRIT images.

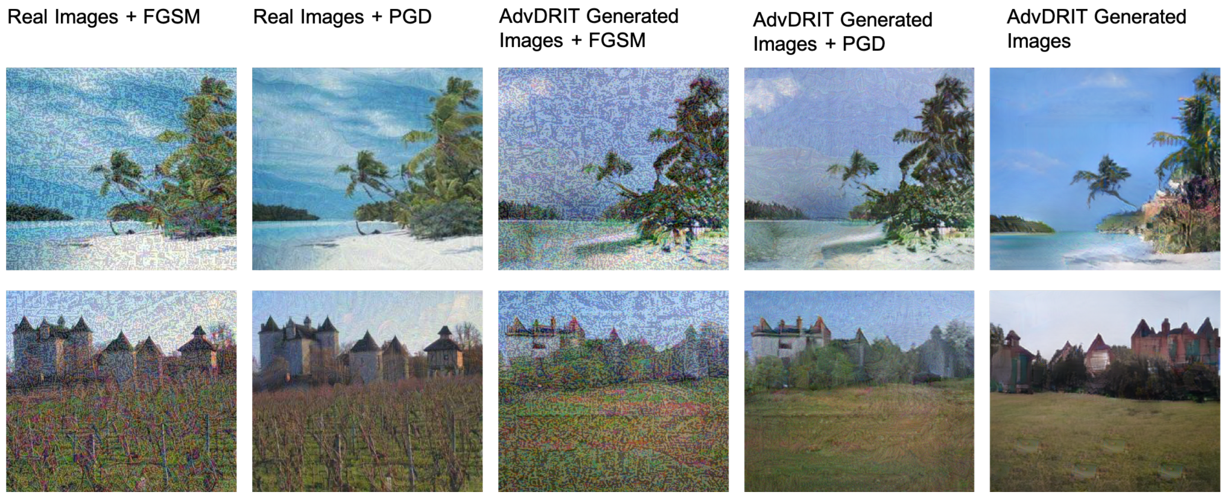
Real Images + FGSM
Real Images + PGD
AdvDRIT Generated Images + FGSM
AdvDRIT Generated Images + PGD
AdvDRIT Generated Images

Figure 4.1. Comparison of traditional and AdvDRIT adversarial images on ADE20K

.

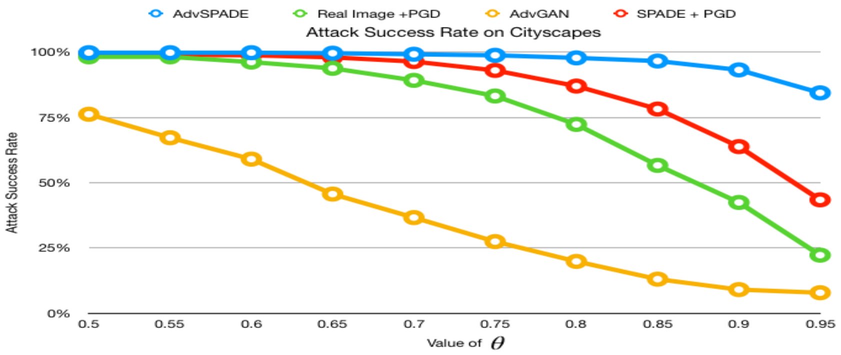

Figure 4.2. Relationship Between Attack Successful Rate and value of θ on Cityscapes

4.2.2 GAN-based and Segmentation Adversarial Attack.

Authors compared the mIoU decline between Song's GAN based unrestricted attack methods (Song et al., 2018) and AdvDRIT on Cityscapes.

The Table 4.1 showed the results. The results indicated Song et al. (2018) generated examples are not effective and can only cause a trivial mIoU decline (from 0.62 to 0.461). Instead, AdvDRIT adversarial images can drop mIoU to 0.010. The transferability of Song's examples is also limited (mIoU drop of 3%). The comparison experimental results demonstrated ineffectiveness of their attacks for semantic segmentation task.

Table 4.5. *Evaluation on Houdini,DAG,AdvGAN and AdvDRIT on Cityscapes*

Method	Bound Size	Attack Success Rate	mIoU	Performance
AdvGAN	0.25	0%	0.350	0.2s/Image
	1	0%	0.344	
	8	0%	0.264	
	32	0%	0.350	
AdvGAN (λ_{adv} = -100)	0.25	0%	0.340	0.2s/Image
	1	0%	0.338	
	8	8.1%	0.075	
	32	6.4%	0.044	
Houdini	8	80.4%	0.013	40s/Image
DAG	-	73.8%	0.014	30s/Image
AdvDRIT(Ours)	-	**89.2%**	**0.01**	**0.25s/Image**

We also compared AdvGAN (C. Xiao et al., 2018), Houdini (Cisse et al., 2017), DAG (Xie et al., 2017) and AdvDRIT using several metrics including the attack success rate, mIoU score and inference time performance on the same experimental environment. According to the Table 4.5, AdvGAN adversarial examples can not attack the target network with 0% attack success rate and 0.350 mIoU decrements for a large perturbation size 32. After fine-tuning the value of hyper-parameter λ_{adv}, AdvGAN can only successfully attack the target model when the bound size is very large. We noticed that the success rate of AdvGAN is 8.1% and mIoU is 0.075 with $\varepsilon = 8$. We noticed that noise in adversarial examples is noticeable when $\varepsilon = 8$. According to the experimental results, authors concluded that AdvGAN did not generalize well on segmentation models. Compared to Houdini and DAG, AdvDRIT also showed better attack result (8.8% higher than Houdini, 15.4% higher than DAG for attack success rate). Meanwhile, AdvDRIT is over 100 times faster than both DAG and Houdini while generating adversarial examples. In conclusion, AdvDRIT can generate adversarial examples successfully in an effective manner.

Table 4.6. *Ablation Results*

Loss Name	mIoU Score	FID Score	Attack Success Rate
w/o Content Adversarial Loss	0.027	77.26	57.6%
w/o Domain Adversarial Loss	0.026	86.14	56.0%
w/o Adv Loss	0.62	61.93	0%
w/o KL Loss	0.021	82.2	62%
AdvDRIT	0.01	63.32	89%

Table 4.7. *Generalization of proposed attack on different I2I architectures on DRN-105*

Metrics Model	mIoU Score	Attack Success Rate	FID Score
AdvDRIT	0.01	89.2%	63.32
AdvSPADE	0.010	84.4%	67.3
AdvPix2Pix	0.030	57.2%	102.7
AdvPix2PixHD	0.021	72.1%	98.4

The Table 4.6 contained the ablation experiments to evaluate each loss item's necessity in AdvDRIT on Cityscapes. The results indicated that content adversarial loss domain adversarial loss and KL loss terms are critical to maintain the adversarial examples quality and attack effectiveness during the training phase. Deleting any of loss items led to the raising of the FID score and the decline of the attack success rate.

The Table 4.7 showed the generalization of authors proposed method. The results showed that by adding our unrestricted adversarial loss, other I2I translation models can also generate adversarial samples successfully. Notice that AdvDRIT achieves the lowest FID score,mIoU and highest attack success rate compared to other two models which showed that the quality of adversarial images and effectiveness of the generation of adversarial samples are beneficial from the model design.

4.2.3 Robustness Evaluation.

Authors showed that robust training with traditional adversarial images can improve the model robustness against restricted adversarial attacks. However, AdvDRIT adversarial examples can easily bypass such robust models. Authors then showed the robustness results of a segmentation model based on proposed AdvDRIT examples. Authors followed the training strategy from Madry, Makelov, Schmidt, Tsipras, and Vladu (2017): authors leveraged Projected Gradient Descent to be the attack agency and trained models adversarially 150 epochs for both ADE20K and Cityscapes. During the training phase, perturbation size $\varepsilon = 8$, numbers of attack iteration is 10 and step size is 1. We used PGD under the same attack strength to construct norm-bounded adversarial examples on natural and vanilla DIRT generated images. We found that traditional norm-bound adversarial examples based on natural and generated images can only decrease mIoU slightly (0.35, 0.275 on adversarial trained DRN-105, 0.239 and 0.197 on adversarial trained Upernet-101). However, AdvDRIT adversarial images can gain 0.042 mIoU on adversarial-trained DRN and 0.027 on adversarial-trained Upernet, which shows that AdvDRIT examples can beat the model adversarial-trained with traditional PGD attack examples. Then, authors adversarial-trained a model with AdvDRIT adversarial images on the Cityscapes dataset. After the training phase, authors used PGD to attack the robust model. The result indicated PGD can gain 1.5% attack success rate and 0.101 mIoU drop on DRN-105. The low success rate showed that models achieve better robustness by considering AdvDRIT examples in the adversarial training stage.

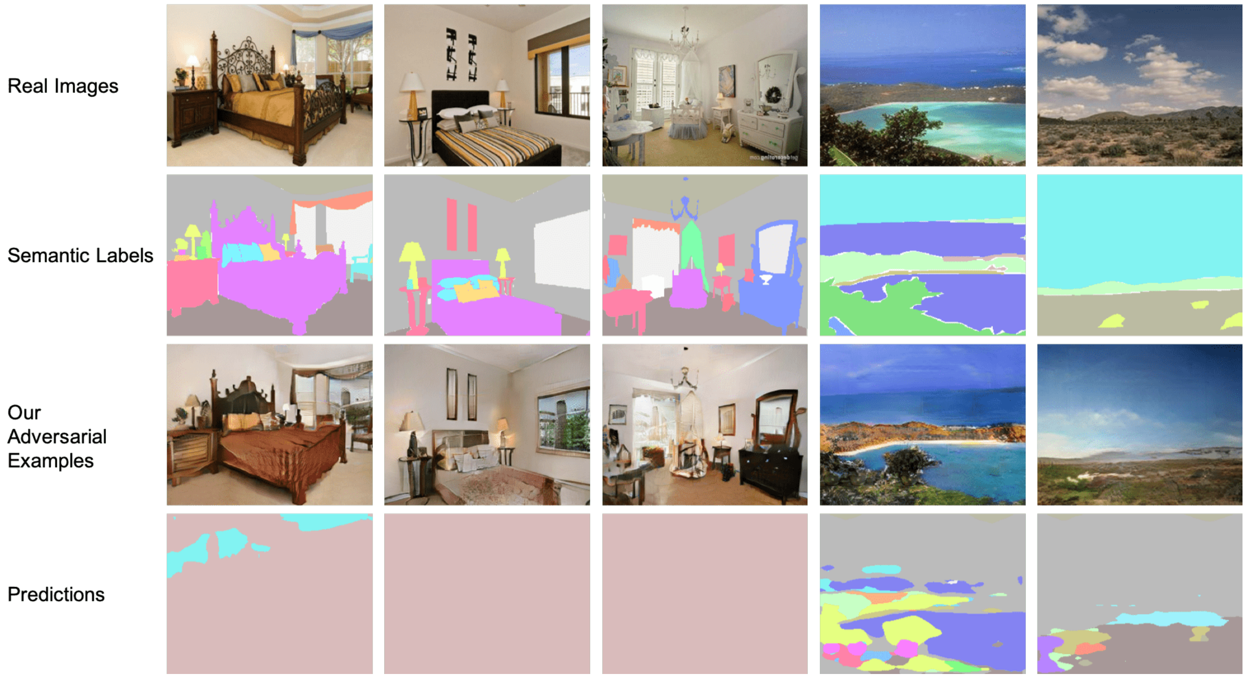

Figure 4.3. Visualized Results on ADE20K

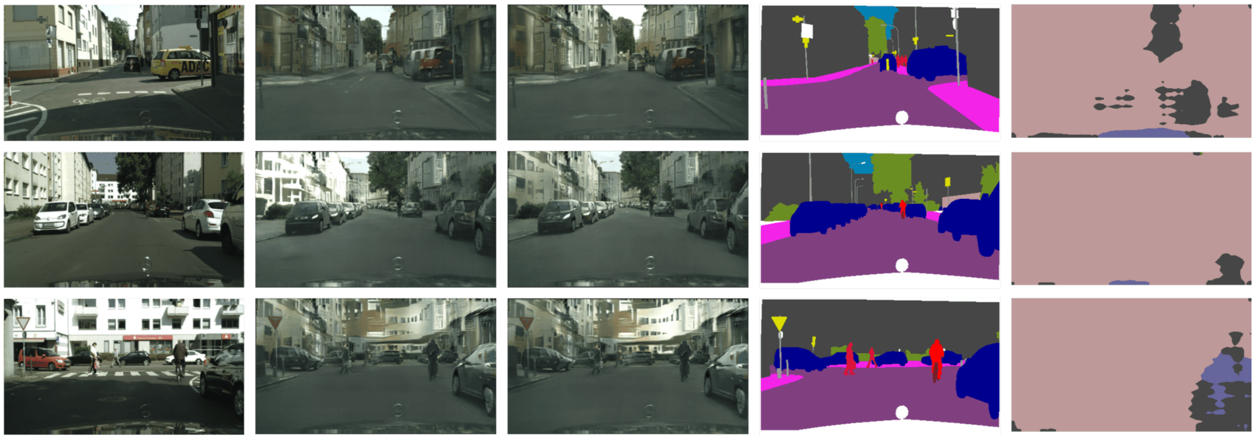

Figure 4.4. Visualized Results on Cityscapes

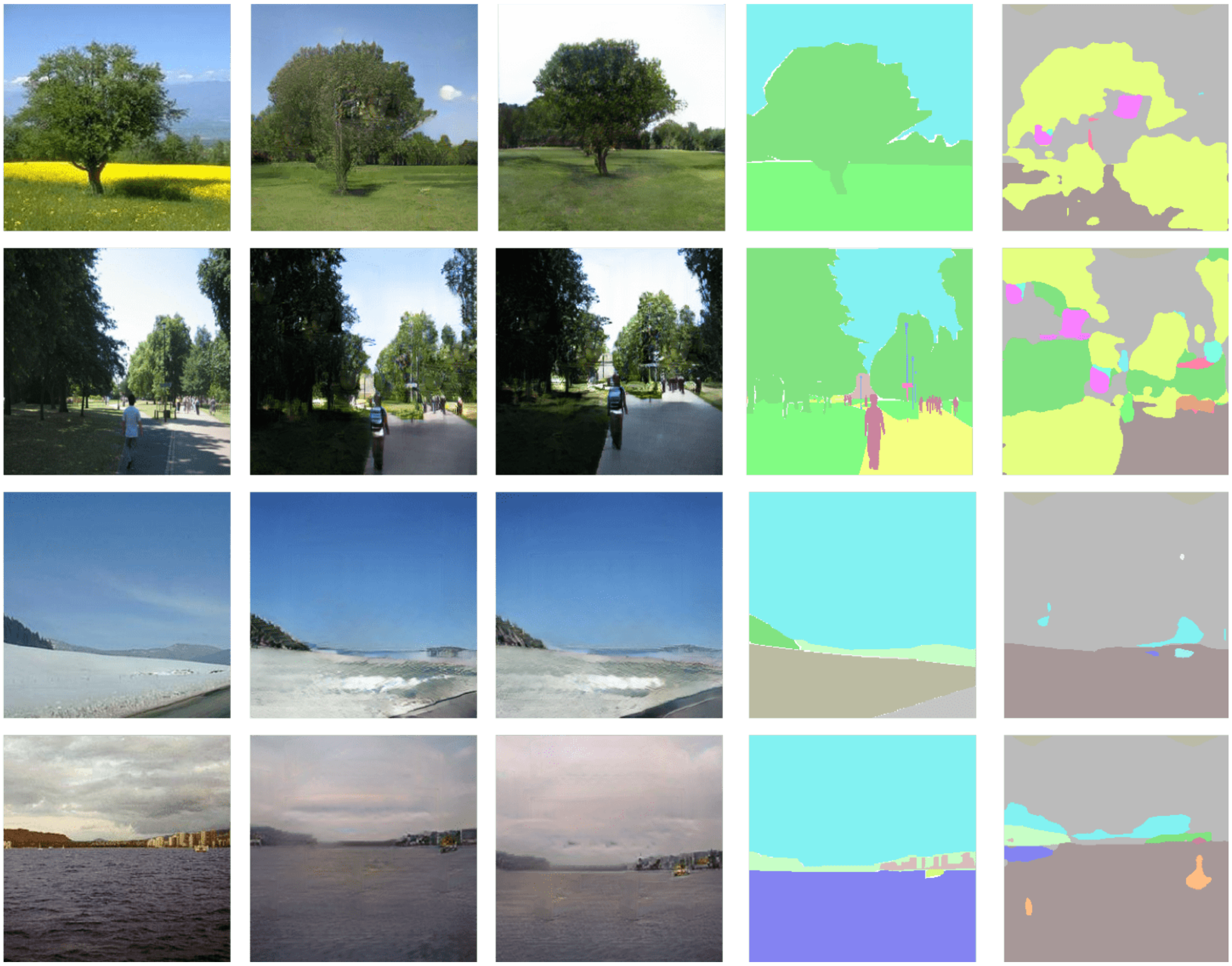

Figure 4.5. Variety of Proposed Unrestricted Adversarial Examples On Cityscapes

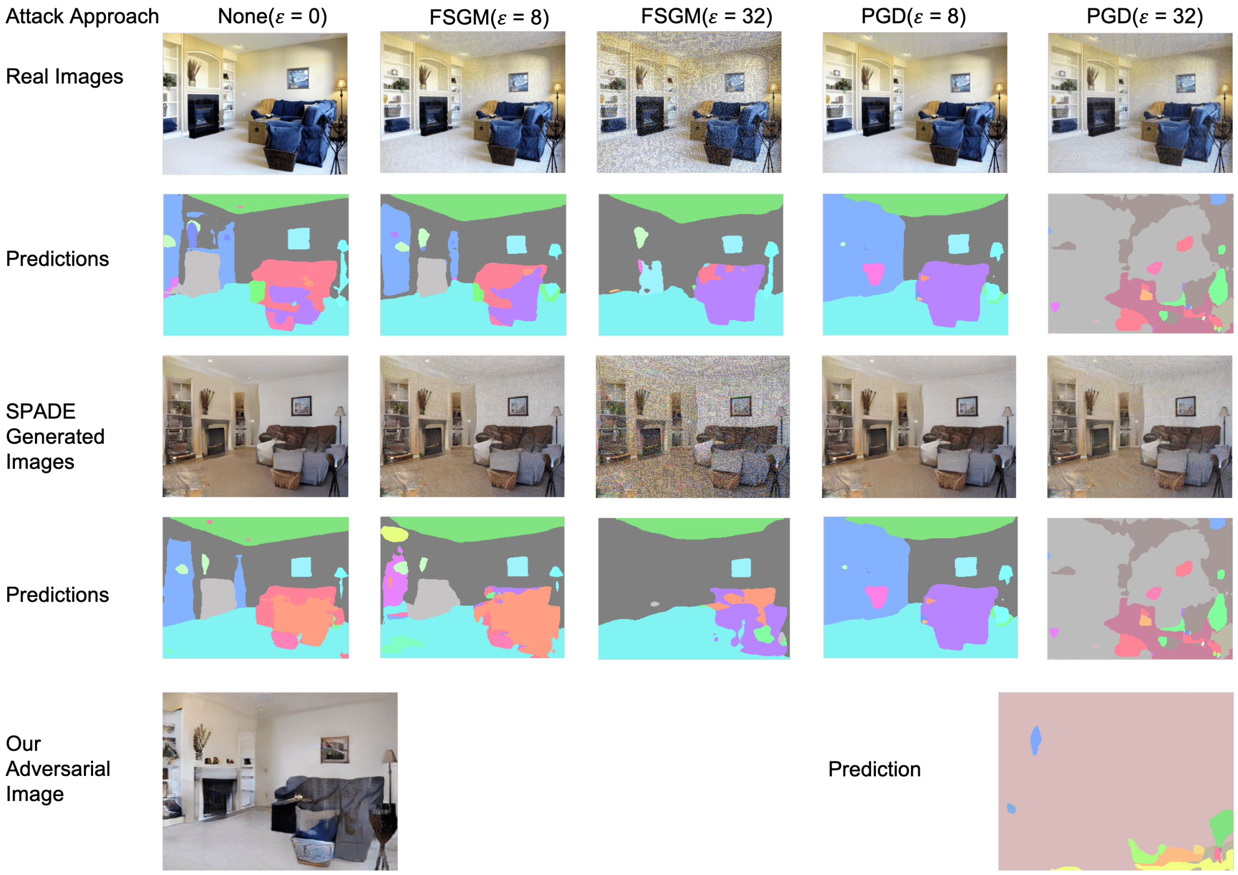

Figure 4.6. Variety of Proposed Unrestricted Adversarial Examples On ADE20K

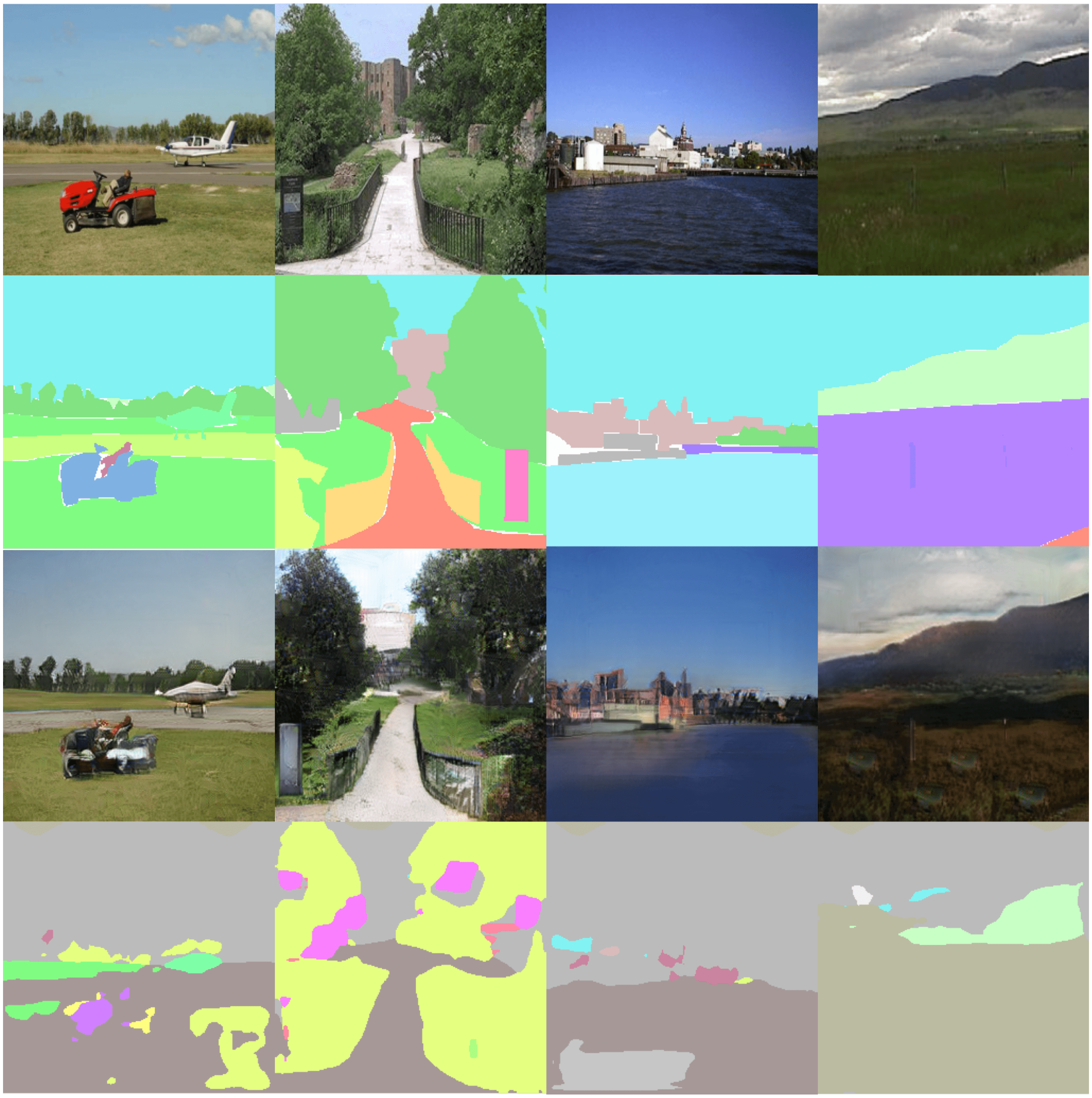

Figure 4.7. Comparison of norm-bounded samples and unrestricted adversarial examples on same mIoU level

CHAPTER 5. CONCLUSIONS

5.1 Conclusions

In this thesis, we presented AdvDIRT: a GAN based method to construct disentangled adversarial samples for modern semantic segmentation neural networks. Borrowing the idea from disentangled representation, we separated features into content feature space and adversarial feature space. Through modifying the loss function of DIRT framework, we further improved the quality of unrestricted adversarial examples beyond any l_p norms, which misled segmentation networks' predictions. We demonstrated the robustness and effectiveness of AdvDRIT by comparing with various advanced existed attack techniques. We also shown that generated adversarial examples can easily bypass the advanced defense method, including adversarial training. AdvDIRT raised new concerns towards security-sensitive fields which rely on deep learning techniques.

5.2 Future Work

Due to the limitation problems in proposed methods, we will further improve the proposed framework in following ways:

- We will further reduce the quality gap between generated adversarial examples and natural images and make them more practical in the real world scenarios.
- We will improve the GAN model design to stabilize the adversarial example generation process.
- We will improve the segmentation networks' robustness by considering the existence of norm-free unrestricted adversarial examples.

REFERENCES

Arnab, A., Miksik, O., & Torr, P. H. (2018, Jun). On the robustness of semantic segmentation models to adversarial attacks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Retrieved from `http://dx.doi.org/10.1109/cvpr.2018.00099` doi: 10.1109/cvpr.2018.00099

Athalye, A., Carlini, N., & Wagner, D. (2018). *Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples.*

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

Bengio, Y., Courville, A., & Vincent, P. (2013, Aug). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798-1828. doi: 10.1109/TPAMI.2013.50

Carlini, N., & Wagner, D. A. (2017). 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).

Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017). *Rethinking atrous convolution for semantic image segmentation.*

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems* (pp. 2172–2180).

Cheung, B., Livezey, J. A., Bansal, A. K., & Olshausen, B. A. (2014). *Discovering hidden factors of variation in deep networks.*

Cisse, M., Adi, Y., Neverova, N., & Keshet, J. (2017). Houdini: Fooling deep structured prediction models. *arXiv preprint arXiv:1707.05373*.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., … Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proc. of the ieee conference on computer vision and pattern recognition (cvpr).*

Denton, E. L., et al. (2017). Unsupervised learning of disentangled representations from video. In *Advances in neural information processing systems* (pp. 4414–4423).

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., & Madry, A. (2017). *A rotation and a translation suffice: Fooling cnns with simple transformations.*

Ess, A., Mueller, T., Grabner, H., & Van Gool, L. J. (2009). Segmentation-based urban traffic scene understanding. In *Bmvc* (Vol. 1, p. 2).

Feinman, R., Curtin, R. R., Shintre, S., & Gardner, A. B. (2017). Detecting adversarial samples from artifacts. *ArXiv*, *abs/1703.00410*.

Goodfellow, I., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *International conference on learning representations.* Retrieved from `http://arxiv.org/abs/1412.6572`

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). *Gans trained by a two time-scale update rule converge to a local nash equilibrium.*

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. In *Nips deep learning and representation learning workshop.* Retrieved from `http://arxiv.org/abs/1503.02531`

Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017, Jul). Image-to-image translation with conditional adversarial networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* Retrieved from `http://dx.doi.org/10.1109/CVPR.2017.632` doi: 10.1109/cvpr.2017.632

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.*

Kingma, D. P., Rezende, D. J., Mohamed, S., & Welling, M. (2014). *Semi-supervised learning with deep generative models.*

Kingma, D. P., & Welling, M. (2013). *Auto-encoding variational bayes.* Retrieved from `http://arxiv.org/abs/1312.6114` (cite arxiv:1312.6114)

Kurakin, A., Goodfellow, I., & Bengio, S. (2016). *Adversarial machine learning at scale.*

LeCun, Y., & Cortes, C. (2010). MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/. Retrieved 2016-01-14 14:24:11, from `http://yann.lecun.com/exdb/mnist/`

Lee, H.-Y., Tseng, H.-Y., Huang, J.-B., Singh, M., & Yang, M.-H. (2018). Diverse image-to-image translation via disentangled representations. *Lecture Notes in Computer Science*, 36–52. Retrieved from `http://dx.doi.org/10.1007/978-3-030-01246-5_3` doi: 10.1007/978-3-030-01246-5_3

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3431–3440).

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International conference on learning representations.* Retrieved from `https://openreview.net/forum?id=rJzIBfZAb`

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015). *Adversarial autoencoders.*

Mathieu, M., Zhao, J., Sprechmann, P., Ramesh, A., & LeCun, Y. (2016). *Disentangling factors of variation in deep representations using adversarial training.*

Milletari, F., Navab, N., & Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3d vision (3dv)* (pp. 565–571).

Mirza, M., & Osindero, S. (2014). *Conditional generative adversarial nets.*

Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*.

Odena, A., Olah, C., & Shlens, J. (2016). Conditional image synthesis with auxiliary classifier gans. In *Icml.*

Park, T., Liu, M.-Y., Wang, T.-C., & Zhu, J.-Y. (2019). *Semantic image synthesis with spatially-adaptive normalization.*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234–241. Retrieved from `http://dx.doi.org/10.1007/978-3-319-24574-4_28` doi: 10.1007/978-3-319-24574-4_28

Song, Y., Shu, R., Kushman, N., & Ermon, S. (2018). *Constructing unrestricted adversarial examples with generative models.*

Sudre, C. H., Li, W., Vercauteren, T., Ourselin, S., & Jorge Cardoso, M. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Lecture Notes in Computer Science*, 240–248. Retrieved from `http://dx.doi.org/10.1007/978-3-319-67558-9_28` doi: 10.1007/978-3-319-67558-9_28

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). *Intriguing properties of neural networks.*

Wang, X., He, K., & Hopcroft, J. E. (2019). *At-gan: A generative attack model for adversarial transferring on generative adversarial nets.*

Wong, E., Schmidt, F. R., & Kolter, J. Z. (2019). *Wasserstein adversarial examples via projected sinkhorn iterations.*

Xiao, C., Li, B., Zhu, J.-y., He, W., Liu, M., & Song, D. (2018, Jul). Generating adversarial examples with adversarial networks. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. Retrieved from `http://dx.doi.org/10.24963/ijcai.2018/543` doi: 10.24963/ijcai.2018/543

Xiao, T., Liu, Y., Zhou, B., Jiang, Y., & Sun, J. (2018). Unified perceptual parsing for scene understanding. In *Proceedings of the european conference on computer vision (eccv)* (pp. 418–434).

Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., & Yuille, A. (2017, Oct). Adversarial examples for semantic segmentation and object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*. Retrieved from `http://dx.doi.org/10.1109/ICCV.2017.153` doi: 10.1109/iccv.2017.153

Yu, F., & Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. In *International conference on learning representations (iclr).*

Yu, F., Koltun, V., & Funkhouser, T. (2017). Dilated residual networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 472–480).

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017, Jul). Pyramid scene parsing network. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Retrieved from `http://dx.doi.org/10.1109/cvpr.2017.660` doi: 10.1109/cvpr.2017.660

Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., & Torralba, A. (2018). Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision*.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017, Oct). Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*. Retrieved from `http://dx.doi.org/10.1109/ICCV.2017.244` doi: 10.1109/iccv.2017.244