Adversarial Examples for Semantic Segmentation and Object Detection

Zhang, Liqiang

June 18, 2018

Abstract

It has been well demonstrated that adversarial examples, i.e., natural images with visually imperceptible perturbations added, cause deep networks to fail on image classi?cation. In this paper, the author extend adversarial examples to semantic segmentation and object detection which are much more dif?cult. their observation is that both segmentation and detection are based on classifying multiple targets on an image. This inspires us to optimize a loss function over a set of targets for generating adversarial perturbations. Based on this, the author propose a novel algorithm named Dense Adversary Generation (DAG), which applies to the state-of-the-art networks for segmentation and detection. They ?nd that the adversarial perturbations can be transferred across networks with different training data, based on different architectures, and even for different recognition tasks.

1. Introduction

Convolutional Neural Networks (CNN) have become the state-of-the-art solution for a wide range of visual recognition problems. Based on a large-scale labeled dataset such as ImageNet and powerful computational resources like modern GPUs, it is possible to train a hierarchical deep network to capture different levels of visual patterns [2]. A deep network is also capable of generating transferrable features for different tasks such as image classi?cation and instance retrieval, or being ?ne-tuned to deal with a wide range of vision tasks, including object detection, visual concept discovery, semantic segmentation, boundary detection, etc.

In this paper, the author go one step further by generating adversarial examples for semantic segmentation and object detection, and showing the transferability of them [3]. To the best of their knowledge, this topic has not been systematically studied before. Note that these tasks are much more dif?cult, as the author need to consider orders of magnitude more targets. Motivated by the fact that each target undergoes a separate classi?cation process, we propose the Dense Adversary Generation (DAG) algorithm, which considers all the targets simultaneously and optimizes the overall loss

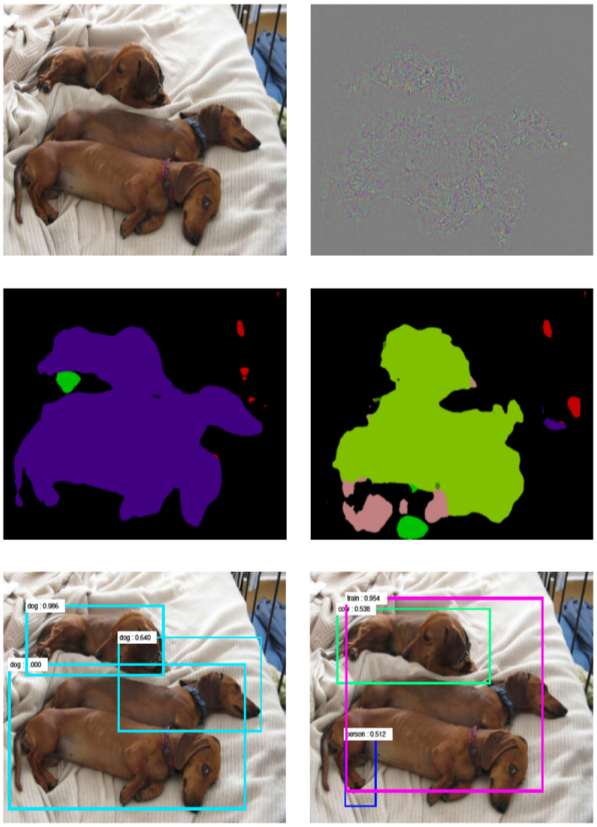

Figure 1. An adversarial example for semantic segmentation and object detection. FCN is used for segmentation, and Faster-RCNN is used for detection. Left column: the original image (top row) with the normal segmentation (the purple region is predicted as dog) and detection results. Right column: after the adversarial perturbation (top row, magnified by 10) is added to the original image, both segmentation (the light green region as train and the pink region as person) and detection results are completely wrong. Note that, though the added perturbation can confuse both networks, it is visually imperceptible (the maximal absolute intensity in each channel is less than 10)

function. The implementation of DAG is simple, as it only involves specifying an adversarial label for each target and performing iterative gradient back-propagation. In practice, the algorithm often comes to an end after a reasonable number of, say, 150 to 200, iterations. Fig. 1 shows an adversarial example which can confuse both deep segmentation and detection networks [1].

2. Generating Adversarial Examples

Let $\mathbf{X}$ be an image which contains N recognition targets $\mathcal{T} = \{t_1, t_2, \ldots, t_N\}$. Each target t_n, $n = 1, 2, \ldots, N$ is assigned a ground-truth class label $l_n \in \{1, 2, \ldots, C\}$, where C is the number of classes, *e.g.*, $C = 21$ in the PascalVOC dataset. Under this setting, the loss function covering all targets can be written as:

$$L(\mathbf{X}, \mathcal{T}, \mathcal{L}, \mathcal{L}') = \sum_{n=1}^{N} [f_{l_n}(\mathbf{X}, t_n) - f_{l'_n}(\mathbf{X}, t_n)] \quad (1)$$

Minimizing L can be achieved via making every target to be incorrectly predicted, *i.e.*, suppressing the con?dence of the original correct class $f_{l_n}(\mathbf{X} + r, t_n$, while increasing that of the desired (adversarial) incorrect class $f_{l'_n}(\mathbf{X} + r, t_n)$.

3. Selecting Input Proposals for Detection

Network	ORIG	ADVR	PERM
FCN-Alex	48.04	3.98	48.04
FCN-Alex*	48.92	3.98	48.91
FCN-VGG	65.49	4.09	65.47
FCN-VGG*	67.09	4.18	67.08
FR-ZF-07	58.70	3.61	58.33
FR-ZF-0712	61.07	1.95	60.94
FR-VGG-07	69.14	5.92	68.68
FR-VGG-0712	72.07	3.36	71.97

Table 1. Semantic segmentation (measured by mIOU, %) and object detection (measured by mAP, %) results of different networks. Here, ORIG is the accuracy obtained on the original image set, ADVR is obtained on the set after the adversarial perturbations are added, and PERM is obtained after the randomly permuted perturbations are added.

In Table 1, it can be found that permuted perturbations cause negligible accuracy drop, indicating that it is the spatial structure of $\mathbf{r}$, instead of its magnitude, that indeed contributes in generating adversarial examples.

References

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE TPAMI*, 40(4):834–848, 2017. 2

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1