上海交通大學

SHANGHAI JIAO TONG UNIVERSITY

学士学位论文

THESIS OF BACHELOR

论文题目： 基于语义的对抗攻击及迁移性研究

学生姓名:	陈思哲
学生学号:	516021910038
专　　业:	自动化
指导教师:	黄晓霖
学院 (系):	电子信息与电气工程学院

基于语义的对抗攻击及迁移性研究

摘 要

针对神经网络的对抗攻击已经被提出许多年了。但是，目前的攻击方法，仅当被攻击的网络内部信息完全已知，或可以通过结构相似性和大量访问来估计的情况下，才有较高的成功率。本文提出注意力攻击（AoA），也就是攻击模型的注意力，这是一种模型间共享的语义特征。由于 AoA 使用了全新的注意力损失，故其能产生迁移性极强的对抗样本。

我们用 AoA 从 ImageNet 验证集中生成了 50000 个对抗样本，还用带访问的 AoA 从其训练集中生成 96020 个对抗样本。它们使得 13 个训练极好的分类网络，产生超过 85% 的错误率。我们发现，通过访问黑盒模型，AoA 的迁移性可以得到进一步加强，产生的样本扰动也更小。我们将这些样本打包成第一个通用对抗数据集 DAmageNet，它可以作为鲁棒测试和对抗训练的基准。

相比于分类网络，目标检测网络有更多的应用，且在诸如自动驾驶和安全监视等涉及生命安全的系统中至关重要。由于检测网络具有多输出特性和结构上的多样性，因此很难对其进行黑盒攻击。为了追求攻击的强迁移性，我们提出了一种多节点注意力热图计算方法，并据此设计了一种对检测网络的攻击。

我们称这种攻击为目标检测网络多节点注意力攻击（ATTACTION），它能在检测网络上达到了目前最好的迁移性。在 MS-COCO 数据集上，7 个黑盒模型的 mAP 指标都被减半，且语义分割的性能受到极大影响。由于 ATTACTION 具有很强的迁移性，我们用其生成了数据集 AOCO。这是目标检测网络上的第一个对抗性数据集，它可以帮助神经网络的设计者快速评估和提高目标检测网络的鲁棒性。

关键词：对抗攻击，注意力热力图，神经网络，可解释性

SEMANTIC-BASED ADVERSARIAL ATTACK AND ITS TRANSFERABILITY

ABSTRACT

Adversarial attacks on deep neural networks (DNNs) have been found for several years. However, the existing adversarial attacks have high success rates only when the information of the victim DNN is well-known or could be estimated by the structure similarity and massive queries. In this thesis, we propose to *Attack on Attention* (AoA), a semantic feature commonly shared by DNNs. AoA enjoys a significant increase in transferability when adopting the attention loss.

We generate 50000 adversarial samples from ImageNet by AoA and 96020 ones by AoA-N with query. 13 well-trained DNNs in classification have an error rate over 85% on these samples. By AoA-N with query, the transferability is further increased with a smaller perturbation. We collect all these samples in *DAmageNet*, which is the first universal adversarial dataset. It may serve as a benchmark for robustness testing and adversarial training.

Object detection networks, compared to classification nets, are more crucial for life-concerning systems such as autonomous driving and security surveillance. Detection networks are hard to attack in a black-box manner because of their multiple-output property and diversity across architectures. To pursue a high transferability of the attack, we propose a novel network visualization method to calculate the attention heat map for multiple nodes and design an attack based on it for detection networks.

This attack, named ATTACk on multi-node attenTION for object detecTION network (ATTACTION), achieves a state-of-the-art transferability in numerical experiments. On MS COCO, the detection mAP for all 7 black-box models are halved and the performance of semantic segmentation is greatly influenced. Given the great transferability of ATTACTION, we generate Adversarial Objects in COntext (AOCO), the first adversarial benchmark on object detection networks, which could help designers to quickly evaluate and improve the robustness of detection networks.

Key words: adversarial attack, attention heat map, deep neural network, interpretability

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Background and Social Impact

Deep Neural Networks (DNNs) have grown into the mainstream tool in many fields, such as image classification[1-3], object detection[4-6], semantic segmentation[7, 8], face recognition[9] and so on. Given the massive applications of DNNs, their vulnerability has attracted much attention in the recent years. An obvious example is the existence of adversarial samples[10-12], which are quite similar with the clean ones, but are able to cheat the DNNs to produce incorrect predictions in high confidence.

Adversarial samples pose a great threat to our real world because DNNs are implemented in scenarios from mobile devices to large-scale systems. The lawbreakers may wear a specific glass to unlock a random cellphone or bypass the surveillance[13] and conduct an unauthorized entry if the security systems adopt DNNs. An autopilot car in high speed may be fooled by even one-frame adversarial patch[14] and crashed before manual control. The lack of theory and interpretability in DNNs induces the lack of their robustness, preventing their applications in automation industry[15].

Since the attacks proposed in this thesis focus on transferable black-box attacks, they would exert a greater damage compared to white-box attacks[11, 16, 17]. But our aim is not to beat DNNs maliciously, but to reveal their weakness, interpret them and help other researchers to improve their robustness. Accordingly, to mitigate the potential negative impacts of our work to the society, we release large-scale datasets, i.e., DAmageNet and AOCO. They contain thousands of high-transferable adversarial samples in our experiment, and therefore could be used by designers for robustness testing.

Although white-box attacks can easily cheat DNNs, the current users actually do not worry about them, since it is almost impossible to get the complete information including the structure and the parameters of the victim DNNs. If the information is kept well, one has to use black-box attacks, which can be roughly categorized into query approaches[18-20] and transfer approaches[21-23]. The former one is to estimate the gradient by querying the victim DNNs. However, until now, the existing query-based attacks still need massive queries, which can be easily detected by the defense systems. Transfer approach attacks rely on the similarity between the victim DNN and the surrogate model in the attacker's hands. It is expected that white-box attack on the surrogate model can also invade the victim DNN. Although there are some promising studies[24-26], the transferability is not satisfactory and a high attack rate could be reached only when two DNNs have similar structures[27], which however conflicts the aim of black-box attack.

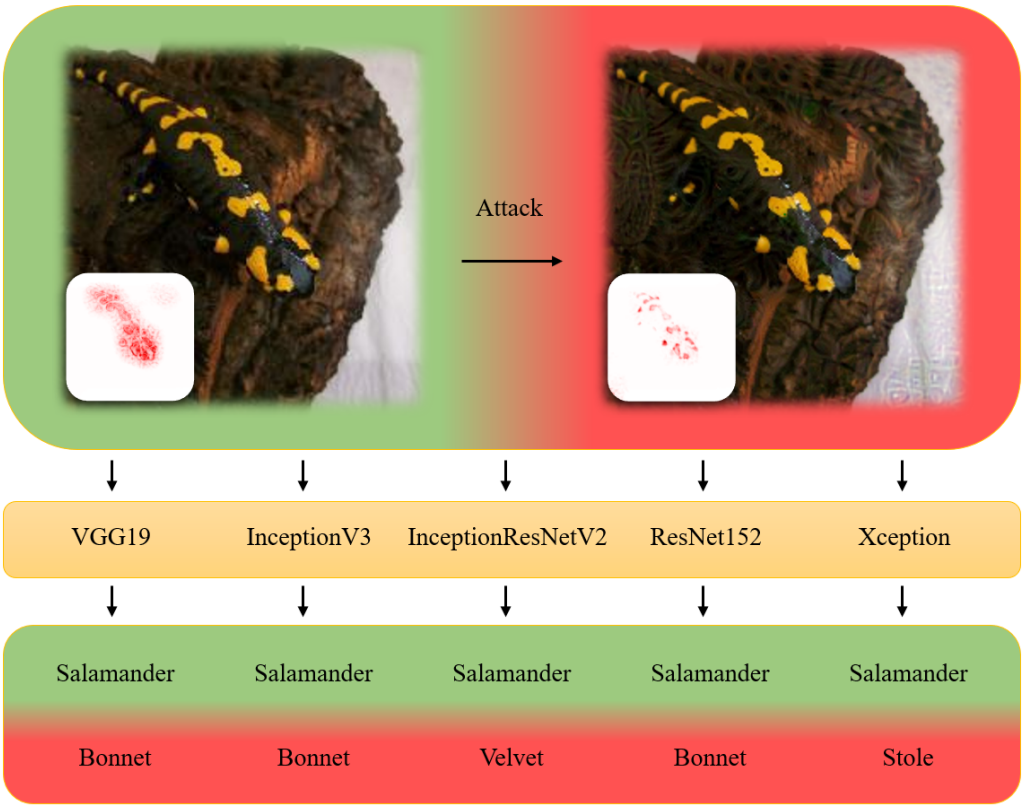

Figure 1–1 AoA adversarial sample and its attention heat map (calculated by DenseNet121). The original sample (in ImageNet: image n01629819_15314.JPEG, class No.25) is shown on the left. All well-trained DNNs (listed in the first row) correctly recognize this image as a salamander. The right image is the generated adversarial sample by AoA. Difference between the two images is slight, however, the heat map shown in lower left corner changes a lot, which fools all the listed DNNs to incorrect predictions, as shown in the bottom row.

1.2 Attack on Classification

Black-box adversarial samples that are applicable to vast DNNs need to attack their common vulnerability. Since DNNs are imitating human's intelligence, although DNNs have different structures and weights, they may share semantic features. In this thesis, we are focusing on the attention heat maps, on which different DNNs have similar results. By attacking the heat maps of one white-box DNN, we could make its attention lose its focus and therefore fail in judgement. In fact, some works[23, 28] have been aware of the importance of attention and put the change of heat map as an evidence of successful attacks, but none of them include the attention in loss as we do. In our study, we develop to *Attack on Attention (AoA)*. AoA has a very good white-box attack performance. More importantly, there is a high similarity in attention across different classification networks, making AoA highly transferable: replacing the cross-entropy loss by AoA loss increases the transfer rate by 10% to 15%. Combined with some ex-

isting transfer-enhancement methods, AoA achieves a state-of-the-art performance, e.g. over 85% transfer rate on all 12 black-box popular DNNs in numerical experiments.

Here, we first illustrate one example in Fig. 1–1. The original image is a "salamander" in ImageNet[29]. By attacking the attention, we generate an adversarial sample, which looks very similar to the original one but with a scattered heat map (in the lower left corner), leading to misclassification. The attack is carried out for VGG19[1] but other well-trained DNNs in ImageNet also classify it incorrectly.

Since AoA is for common vulnerabilities of DNNs, we successfully generate 50000 adversarial samples that can cheat many unknown classification networks, whose error rates increase to over 85%. We provide these samples in the dataset named as *DAmageNet*. DAmageNet is the first dataset that provides black-box adversarial samples. Those images *DAmage* many neural networks without any knowledge or query. But the aim is not to really damage them, but to point out the weak parts of neural networks. Those samples are valuable to improve the neural networks by adversarial training[30, 31], robustness certification[32], and so on.

1.3 Attack on Object Detection

In comparison to attacks in classification, attacking object detection in a black-box manner, e.g., hiding certain objects from being detected[33], exerts a huge impact on security systems[13]. In this way, the attacker may be able to bypass the surveillance for an unauthorized entry or induce a traffic accident for an autonomous-driving vehicle[14], even without knowledge or control of the inner system.

To the best of our knowledge, no existing attack is specifically designed for the black-box transferability in detection networks. The main reason is that high-transferable adversarial attack needs to be targeted on common property across architectures. In this thesis, we concentrate on the attention heat maps, on which different detection nets have similar results as shown in Fig. 1–1. Although some works have adopted the attention heat map as an indicator of success attack[23, 28], we are the first to directly attack the attention of detection networks, i.e., include it in the attack loss.

We design the ATTACk on multi-node attenTION for object detecTION network (ATTACTION). ATTACTION focuses on suppressing the attention rather than directly changing the prediction as in existing works[12, 34, 35]. Since the heat map is quite similar across models, those of other black-box models are influenced as well, leading to the black-box transferability.

However, the attention heat map for detection nets is difficult to obtain. The reason is that their predictions are formed by multiple outputs, i.e., the location and confidence

for several bounding boxes, but existing methods[36-38] all focus on visualizing classification nets whose decision is formed by a single output. Accordingly, we propose Multi-Node Softmax Gradient Layer-wise Relevance Propagation (MN-SGLRP), the first visualization method for object detection.

Figure 1–2 Attention heat maps for models with different architectures by our proposed MN-SGLRP. Three models not only predict the "stop sign" right, but also clearly capture its correct outline.

With the calculated heat map, a common property across architectures, ATTACTION achieves state-of-the-art transferability on 7 black-box models for COCO Challenge[39], nearly halving mAPs for all of them. ATTACTION is also flexible to be combined with other techniques[23, 25, 26] for better transferability. Interestingly, the adversarial samples from ATTACTION also greatly influence the performance of semantic segmentation, even the attacked model is for detection only.

Given the high transferability of ATTACTION, we create Adversarial Objects in COntext (AOCO), the first adversarial dataset for detection networks. AOCO contains 10000 samples that significantly decrease the performance of black-box models for detection and segmentation. AOCO may serve as a new benchmark to evaluate the robustness of DNNs or improve it by adversarial training.

1.4 Thesis Organization

The rest of this thesis is organized as follows. In Chapter 2, we will briefly review the related researches. AoA is described and experimentally evaluated in detail in Chapter 3. Chapter 4 introduces the variants of AoA and adopts the query approach to increase the transferability. ATTACTION to attack object detection would be illustrated and fairly evaluated in Chapter 5. Chapter 6 gives a conclusion and also discusses possible future work.

Chapter 2 Literature Review

2.1 Deep Neural Networks for Image Classification

Image classification is a typical problem in pattern recognition. Since the dimension of an image is large, this task is quite challenging. Before the thrive of deep learning[40], this problem is generally solved by machine learning methods such as Support Vector Machine, Random Forest[41] and AdaBoost[42]. These methods may achieve a satisfactory accuracy when handling low-resolution images such as Modified NIST (MNIST)[43], whose samples are 28×28 gray images in 10 classes. Even for CIFAR-10[44], the dataset containing 64×64 color images, they could not have a good performance. Deep learning thrives since AlexNet[45] achieved an overwhelming accuracy in 2012 ImageNet Large Scale Visual Recognition Challenge[46]. ImageNet[29] includes 1300 samples in each 1000 classes, so classifying them correctly is very difficult. Even humans have a top-5 error rate of 5.1%[46].

AlexNet, compared with earlier works of neural networks[43], adopts Rectified Linear Units (ReLU)[47] activation function, which excels in introducing both the non-linearity and sparsity into the model. It also adopts Dropout layers[48] and Local Response Normalization (LRN) to prevent over-fitting. Multiple GPUs are used to accelerate the training for the first time. AlexNet is a milestone in deep learning, leading to a new wave of neural networks.

The shallow neural network has certain limitations in large-scale image classification tasks. In order to explore their performance, VGG models are proposed[1]. The author thoroughly evaluate networks by increasing depth and using small (3×3) convolution filters, which shows that a significant improvement could be achieved by simply pushing the network deeper. VGG19 enjoys a top-5 error rate of 7.3% in ImageNet.

Competing with VGG is the GoogLeNet group[49, 50], which proposes 4 versions of DNNs, named InceptionV1 to InceptionV4. They adopt the inception module, which contains different kernel sizes (1, 3, 5) of convolution layers and concatenate them as the output. InceptionV2 also includes Batch Normalization layers[51]. InceptionV3 uses spatial factorization into asymmetric convolutions to further increase the accuracy, which reaches 3.58% (top-5) in ImageNet.

ResNet[2] was then proposed mainly to solve the side effects (degradation) caused by increasing network depth so that the performance can be simply improved by simply adding layers. ResNet constructs residual blocks and can break through a 100-layers barrier and even reach up to 1000 layers. The residual block adds a branch for direct

forward propagation so that the gradients are hard to vanish in back propagation. The top-5 error rate of ResNet is 3.57%, the lowest one in 2015. Variants of ResNet include ResNeXt[52], RegNet[53] and so on.

Based on ResNet, DenseNet was proposed[3], which uses the novel dense block. One main difference between DenseNet and the previous works is that DenseNet can accept fewer feature maps as the output of each layer. This gives DenseNet an advantage that the features extracted by earlier layers can still be used by deeper layers through dense connections. It achieved a 3.74% top-5 error in the final year ImageNet challenge.

2.2 Adversarial Attack and Its Defense

Adversarial attacks[10] could reveal the weakness of DNN by cheating it with adversarial samples, which differ from original ones with only a slight perturbation. In the humans' eyes, the adversarial sample does not differ from the original ones, but well-trained networks make false predictions on them in high confidence. The adversarial attack can be expressed as below,

$$\begin{aligned} &\text{find } \Delta x \\ &\text{s.t. } f(x) \neq f(x+\Delta x) \\ &\quad \|\Delta x\| \leq \varepsilon. \end{aligned}$$

When training a DNN, one updates the weights of the network by the gradients to minimize a training loss. While in adversarial attacks, one alters the image to increase the training loss. Based on this basic idea, there have been many variants on attacking spaces and crafting methods.

Adversarial attacks could be roughly categorized as gradient-based[11, 17] and optimization-based methods[10, 16]. Gradient-based methods search in the gradient direction and the magnitude of perturbation is restricted to avoid a big distortion. Optimization-based methods usually consider the magnitude restriction in the objective function. For both, the magnitude could be measured by l_1, l_2, l_∞-norm or other metrics.

For the attacking space, most of the existing methods directly conduct the attack in the image space[11, 54, 55]. It is also reasonable to target at the feature vector in latent space[56, 57] or the encoder/decoder[58, 59]. Attack on feature space may produce a unique perturbation unlike random noise.

Different from noise-based attacks, feature-based attacks are recently proposed. The attacker could change the framework of variational auto-encoder[60] to calculate the additive or multiplicative perturbation[61]. This attack alters latent feature vectors

in a minimum degree so the output of the classifier remains the same, but the decoded images belong to different classes. Attackers could also use an AC-GAN[62] to construct unrestricted adversarial samples[56]. Since it uses the auxiliary classifier (AC) as an oracle to guarantee that the samples remain in the same class in view of human, AC must be for the same task as the victim network. In[57], Type I attack was discussed compared to the traditional Type II attack. Type I attack cheats the classifiers with significant changes, but the output of victim model remains unchanged.

Given the threat of adversarial attack, many researches provide explanations for them. It is found that the attack influences the attention of DNNs, which may account for the aggression of adversarial samples[23, 28] as also emphasised in this thesis. Another thriving topic is to study the boundary of attack's threat and therefore certify the robustness of DNNs[63, 64]. In this way, the degree of aggression or robustness could be theoretically calculated. Adversarial samples could also be interpreted as a result of learning from non-robust features[65], which are defined in the image space and disentangled from robust ones by optimization in encoding. They are falsely exploited by DNNs to increase accuracy, which reveals their limited generalization.

To secure the DNN, many defense methods have been proposed to mitigate the adversarial attack. Defense can be achieved by simply adding adversarial samples to the training set, which is called adversarial training[66-68]. It is very effective, but consumes several-fold time. Another technique is to design certain blocks in network structure to prevent attacks or detect adversarial samples[69, 70]. Attack can also be mitigated by preprocessing images before input to the DNN[71-73], which does not require modification on the pre-trained network.

2.3 Attack Object Detection

Adversarial attack could be extended to detection networks[12], which attacks the classification loss and proposes to target on densely generated bounding boxes. After that, losses about localization and classification are designed in[34] for attack. Physical adversarial patches is feasible[33, 74]. There are also works that propose to attack detectors with perturbation in a restricted area[35, 75]. Existing works achieve good results in white-box scenarios, but are not specifically designed for black-box transferability, which is quite limited (5 to 10% decrease from the original mAP) even when two models only differ in backbone[12, 34, 76]. Black-box attack for detection nets has been considered[77], but it is based on queries rather than transferability as we do. The performance is satisfactory, but it requires over 30K queries, which is easy to be detected.

2.4 Black-Box Attack

When the victim DNNs are totally known, the attacks have high success rates. However, it is almost impossible to have access to the victim model in real-world scenarios and thus black-box attacks are required[78-80]. Black-box attacks rely on either query[18, 19] or transferability[21, 78].

For the query approach, the attacker adds a slight perturbation to the input image and observes the reaction of the victim model. By a series of queries, the gradients could be roughly estimated and then one can conduct an attack in the way similar to white-box cases. To choose the next pixel to alter for gradient estimation, attackers adopt methods including Bayes optimization[81], evolutional algorithms[82], meta learning[83] etc. Since the practical DNNs are generally very complicated, good estimation of the gradients needs a massive number of queries, leading to an easy detection.

For the transfer approach, one conducts the white-box attack in a well-designed surrogate model and expects that the adversarial sample remains aggressive to other models. The underlying assumption is that the distance between decision boundaries across different classes is significantly shorter than that across different models[78]. Although a good transfer rate has been shown recently[24-26, 84], it is mainly across models in the same family, e.g., InceptionV3 and InceptionV4, or models with the same block, e.g., residual blocks[27]. Until now, cross-family transferability of adversarial samples with small perturbation is limited and there is no publicly available dataset of that.

2.5 Variants of ImageNet

To demonstrate and evaluate our attack, we will modify images from ImageNet as other transfer attacks[24-26, 84] and create our dataset. Recently, many interesting variants of ImageNet have been developed.

ImageNet-A[85] contains real-world images in ImageNet classes, which mislead current classifiers to output false predictions. Objects in ObjectNet have random background and viewpoint so that models in ImageNet cannot distinguish. ImageNet-C[86] is produced by adding 15 diverse corruptions. Each type of corruptions has 5 levels from the lightest to the severest. ImageNet-P[86] is designed from ImageNet-C and differs from it in possessing additional perturbation by image transformations.

The datasets above are valuable for testing and improving the network generalization, but DAmageNet and AOCO are for robustness. In other words, samples in the above datasets differ from the samples in ImageNet and the low accuracy is due to the poor generalization. In DAmageNet and AOCO, the samples are quite similar to the original ones in ImageNet and the low accuracy is due to the over-sensitivity of DNNs.

2.6 Attention Heat Map

In making judgements, humans tend to concentrate on certain parts of an object and allocate attention efficiently. This attention mechanism in human intelligence has been exploited by researchers in natural language processing[87, 88]. In computer vision, the same idea has been applied to visualize and interpret DNNs[15].

To attack on attention, we need to calculate the attention heat map of each pixel, for which network visualization methods[89-91] are applicable. Forward visualization adopts the intuitive idea to obtain the attention by observing the changes in the output by changes in the input. The input can be modified by noise[92], masking[93], or perturbation[94]. However, these methods consume much time and may introduce randomness.

In contrast, backward visualization[93, 95, 96] obtains the heat map by calculating the relevance between adjacent layers from the output to the input. The layer-wise attention is obtained by the attention in the next layer and the network weights in this layer. Significant works include Layer-wise Relevance Propagation (LRP)[36], Contrastive LRP (CLRP)[37] and Softmax Gradient LRP (SGLRP)[38]. These methods extract the high-level semantic attention features for the images from the perspective of the network and make DNNs more interpretable and explainable. However, they could only visualize the attention for single target node for classification nets.

Chapter 3 Attack on Attention without Query

To pursue a high transferability for the black-box attack, we need to find common vulnerabilities and attack semantic features shared by different DNNs. Attention heat maps for three images are illustrated in Fig. 3–1. Even with different architectures, the models have similar attention, the pixel-wise heat maps showing how the input contribute to the prediction. Inspired by the similarity across different DNNs, we propose to Attack on Attention (AoA).

Figure 3–1 Attention heat maps for VGG19[1], InceptionV3[50], DenseNet121[3], which are similar even the architectures are different.

3.1 Method

Different to the existing methods that focus on directly attacking the output, the proposed AoA aims to change the attention heat map. Let $h(x, y)$ stand for the attention heat map for the input x and a specified class y. The basic idea of AoA is to shift the attention away from the original class, e.g. decrease the heat map for the correct class

y_{ori}, as illustrated in Fig. 3–2. In this thesis, we utilize SGLRP[38] to calculate the attention heat map $h(x, y)$, which is good at distinguishing the attention for the target class from the others. There are of course many techniques for obtaining the heat map to attack, as long as $h(x, y)$ and its gradient on x could be effectively calculated.

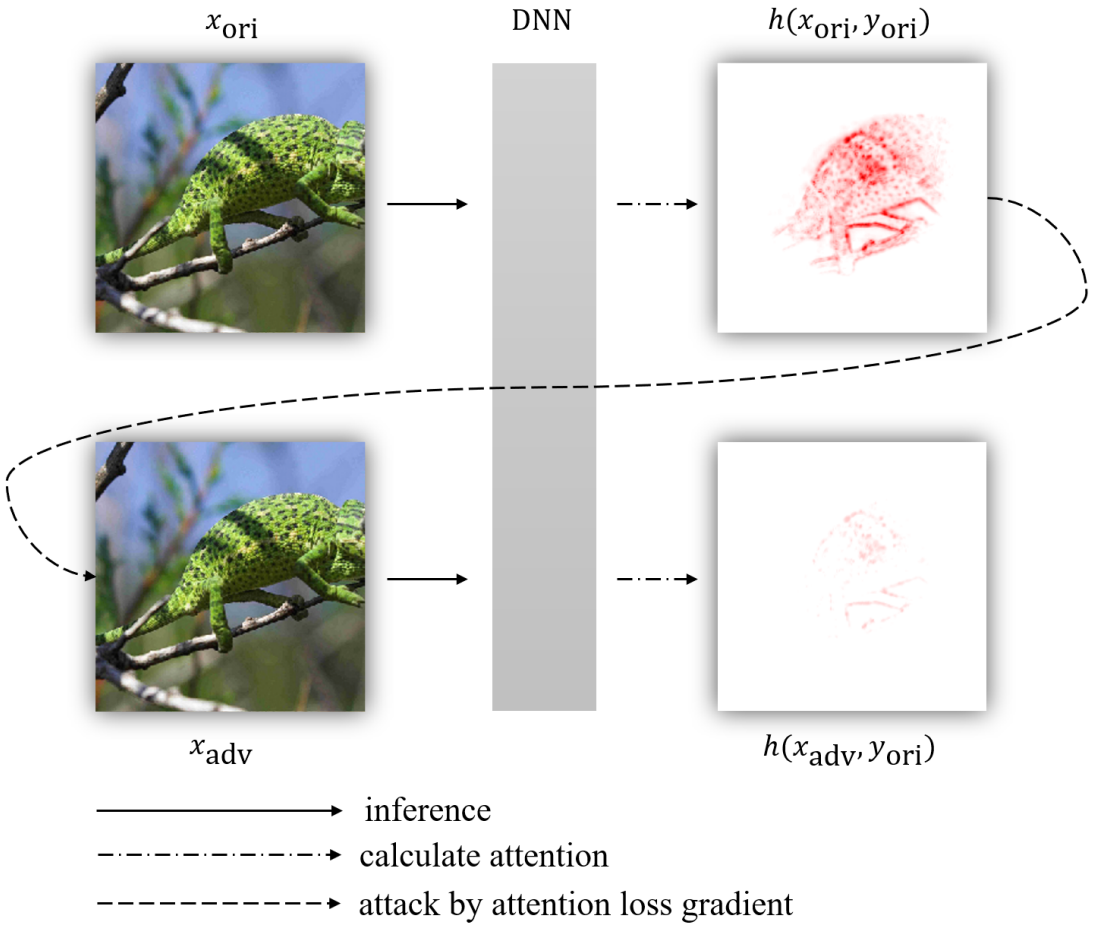

Figure 3–2 The design of AoA. AoA calculates the attention heat map by SGLRP after inference. The gradient from the heat map back-propagates to the input and updates the sample iteratively. By suppressing the attention heat map value, one can change the network decision by fooling its focus. Constantly doing this, the produced adversarial sample could beat several black-box models.

Intuitively, there are several potential ways to change the attention heat maps.

1. Suppress the magnitude of attention heat maps for the correct class $h(x, y_{\text{ori}})$: when the network attention degree on the correct class decreases, attention for other classes would increase and finally exceed the correct one, which leads the model to seek for information on other classes rather than the correct one and thus make an incorrect prediction. We call this design as *suppress loss* and specify the loss function as

$$L_{\text{supp}}(x) = ||h(x, y_{\text{ori}})||_1.$$

2. Distract the focus of $h(x, y_{\text{ori}})$: it could be expected that when the attention is distracted from the original regions of interest, the model may lose its capability for prediction. In this case, we do not require the network to focus on information of any incorrect class, but lead it to concentrate on irrelevant regions of the image. The loss could be expressed as the following *distract loss*,

$$L_{\text{dstc}}(x) = -||\frac{h(x, y_{\text{ori}})}{max(h(x, y_{\text{ori}}))} - \frac{h(x_{\text{ori}}, y_{\text{ori}})}{max(h(x_{\text{ori}}, y_{\text{ori}}))}||_1.$$

   Here, self-normalization is conducted to eliminate the influence of attention magnitude.

3. Decrease the gap between $h(x, y_{\text{ori}})$ and $h(x, y_{\text{sec}}(x))$, the heat map for the second largest probability: If the attention magnitude for the second class exceeds that for the correct class, the network would focus more on information about the false prediction, which is inspired by CW attack[16]. We call it *boundary loss* and take the following formulation,

$$L_{\text{bdry}}(x) = ||h(x, y_{\text{ori}})||_1 - ||h(x, y_{\text{sec}}(x))||_1.$$

   The values of attention heat maps vary a lot for different models, so the self-normalization may improve the transferability of adversarial samples. Therefore, rather than the boundary loss, we can also consider the ratio between $h(x, y_{\text{ori}})$ and $h(x, y_{\text{sec}}(x))$, resulting the following *logarithmic boundary loss*

$$L_{\text{log}}(x) = \log(||h(x, y_{\text{ori}})||_1) - \log(||h(x, y_{\text{sec}}(x))||_1).$$

Table 3–1 Transfer rate from ResNet50 to other neural networks

Loss	DN121[3]	VGG19[1]	RN152[2]	IncV3[50]	IncRNV2[97]	NASNetL[98]
CW[16]	66.6%	54.2%	47.3%	39.6%	37.9%	28.8%
PGD[17]	67.8%	54.2%	46.8%	38.7%	35.6%	28.4%
$L_{\text{supp}}(x)$	66.8%	57.2%	54.8%	43.9%	41.6%	33.0%
$L_{\text{dstc}}(x)$	67.1%	56.5%	55.5%	45.4%	40.0%	31.0%
$L_{\text{bdry}}(x)$	50.2%	49.8%	44.0%	34.1%	32.9%	21.7%
$L_{\text{log}}(x)$	74.9%	64.2%	59.2%	50.1%	46.2%	36.3%
$L_{\text{AoA}}(x)$	**78.7%**	**64.9%**	**63.9%**	**53.3%**	**48.9%**	**41.0%**

In Table 3–1, we compare the performance of losses above (attack on ResNet50[2] and send the adversarial samples to other DNNs as the setting in experiments below).

The white-box attack success rates are all near 100% but they have a different transferability. The suppress loss and the distract loss have a slightly better transferability compared to PGD and CW. In contrast, the logarithmic boundary loss is the best. Moreover, attack on attention could be readily combined with the existing attack on prediction (the loss in PGD), resulting in the following *AoA loss*,

$$L_{\mathrm{AoA}}(x) = L_{\log}(x) - \lambda L_{\mathrm{ce}}(x, y_{\mathrm{ori}}), \tag{3–1}$$

where λ is a trade-off between the attack on attention and prediction. In this thesis, $\lambda = 1000$ is suggested. The combination increases the transferability as in Table 3–1.

Basically, the adversarial samples are generated in an update process by minimizing the AoA loss L_{AoA}. Specifically, set $x^0_{\mathrm{adv}} = x_{\mathrm{ori}}$ and the update procedure could be generally described as the following

$$\begin{aligned} x^{k+1}_{\mathrm{adv}} &= \mathrm{clip}_{\varepsilon}\left(x^k_{\mathrm{adv}} - \alpha \frac{g(x_k)}{||g(x_k)||_1/N}\right), \\ g(x) &= \frac{\partial L_{\mathrm{AoA}}(x)}{\partial x}. \end{aligned} \tag{3–2}$$

The gradient g is normalized by its average l_1-norm, i.e., $||g(x_k)||_1/N$ where N is the size of the image. Further, to keep the change invisible, we restrict our attack by the distance from the original clean sample, i.e., the sample is l_∞-norm-bounded around the original sample by ε. By iterative update, we could gradually change the image x until its prediction is incorrect. AoA is different from other attacks merely on the loss. Therefore, transferability enhancement techniques developed for directly attacking prediction are also applicable to AoA. In fact, with optimization modification[24] or input modification[25, 26], the transfer performance of AoA gets further improved, which would be numerically verified later.

Because of its good transferability on heat maps, AoA could be used for the black-box attack. The basic scheme is to choose a white-box DNN, which serves as the surrogate model for the black-box model, to attack by update (3–2). The adversarial samples generated tend to be aggressive to other black-box victim models.

Now let us first illustrate the attack result on the attention heat map. In Fig. 3–3, a clean sample in the class "pinwheel" is drawn together with its heat maps on this class. Aiming at VGG19[1], we apply AoA and successfully change the heat map (at the bottom). This common property shared by the attention in different DNNs makes the attack transferable, which is the motivation of AoA. The generated adversarial sample is shown in the leftmost in the bottom, which is incorrectly recognized by all the DNNs in Fig. 3–3. Additionally, we could see that the heat map for VGG19 is much clearer, which might explain the high transferability of its adversarial samples as shown later and also in other works[27].

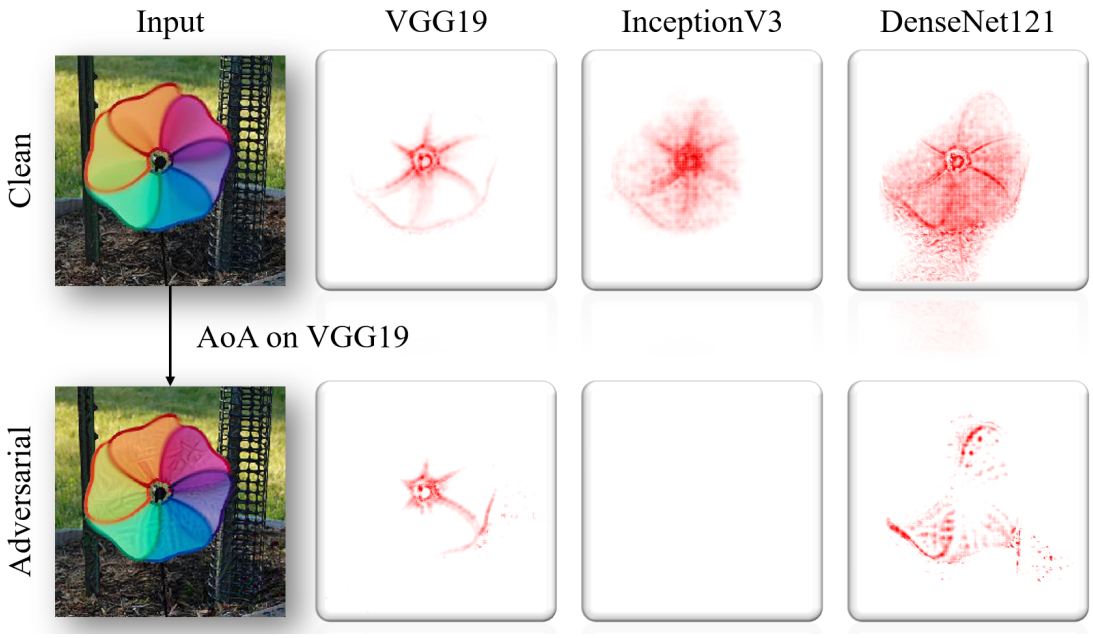

Figure 3–3 Attention heat map (on the correct label "pinwheel") for clean and AoA adversarial sample. For clean samples, these heat maps have some common features, implying the possibility of transferability. By AoA on VGG19, the heat maps for not only VGG19 but also other black-box models are disturbed a lot, thus they all make incorrect predictions.

3.2 Experiments

In this section, we will evaluate the performance of our Attack on Attention, especially its black-box attack capability compared to other state-of-the-art methods. Since AoA is a very good black-box attack, it provides adversarial samples that can defeat many DNNs in the zero-query manner. These samples are collected in the dataset DAmageNet. This section will also introduce DAmageNet and report the performance of different DNNs on it. We further test the AoA performance under several defenses and find that AoA is the most aggressive method in most cases.

The experiments for AoA are conducted on ImageNet[29] validation set. For attack and test, several well-trained models in Keras Applications[99] are used, including VGG19[1], ResNet50[2], DenseNet121[3], InceptionV3[50] and so on. We also use other adversarial-trained models (not by AoA, indicated by underline). We pre-process with Keras pre-processing function, central cropping and resizing (to 224). The experiment is based on TensorFlow[100], Keras[99] with 4 NVIDIA GeForce RTX 2080Ti GPUs.

For the attack performance, we care about two aspects: the success/transfer rate of attack and how large the image is changed. Denote the generated adversarial sample as x_{adv}. The change from its corresponding original image x_{ori} could be measured by the Root Mean Squared Error (RMSE) in each pixel: $d\left(x_{\text{adv}}, x_{\text{ori}}\right) =$

$\sqrt{\sum \left(x_{\text{adv}} - x_{\text{ori}}\right)^2 / N}$, where N is the size of the image. In the experiments, 1000 images are randomly selected from ImageNet validation set and the samples incorrectly predicted by the victim model are skipped as the same setting in[27]. All the compared attacks will be fairly stopped when RMSE exceeds $\eta = 7$ and the perturbation is bounded as $\varepsilon = 0.1 * 255$. In this way, the number of iterations is about 10 with step size $\alpha = 2$ as the setting of[84] and other attacking experiments. We alter $\alpha = 0.5$ for MI[24] to avoid large RMSE distortion.

3.3 Transferability of AoA

We first compare AoA with popular attacks CW[16] and PGD[17], which use the hinge loss and cross entropy loss respectively. For CW, the gradient-based method is applied to update the original to keep the perturbation the same. We carefully tune the parameters, resulting in a better transferability than reported by[27]. We use AoA, CW, and PGD to attack different neural networks, and then send the generated adversarial samples to different models. The average error rates are reported in Table 3–2. AoA, CW, and PGD all have a high white-box attack success rate but the transfer performance varies a lot, which depends on both the surrogate model and the victim model. But in all the situation, AoA achieves a better black-box attack performance.

The essential difference of AoA from CW/PGD is the attack target. The existing effort on improving attack transferability is mainly on modifying the optimization process. For example, DI[25] proposes to transform 4 times when calculating the gradient with probability ($p = 1$ here for better transferability as suggested). MI[24] tunes the momentum parameter to $\mu = 1$ for boosting attacks. SI[26] divides the sample by the power 2 for 4 times to calculate the gradient. Those state-of-the-art transfer enhancement methods could improve CW/PGD and are also applicable to AoA.

In Table 3–3, we report the black-box attack performance when attacking ResNet with MI-DI and SI. We found that SI is very helpful and can prominently increase the error ratio for PGD and CW. For other surrogate models, the performance of MI-DI and SI is similar and could be found in the Appendix. When applying SI in AoA, the obtained SI-AoA achieves the highest transfer rate, which is significantly better than other state-of-the-art methods.

3.4 AoA under Defenses

Our main contribution in this thesis is for black-box attack by increasing the transferability. It is not necessary that we can also break defenses, but indeed, it is interesting

Table 3–2 Error rate (Top-1) of different attack baselines

Surrogate	Method	DN121	IncRNV2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
RN50	CW	66.6%	37.9%	39.6%	28.8%	47.3%	100.0%	54.2%	37.4%
	PGD	67.8%	35.6%	38.7%	28.4%	46.8%	**100.0%**	54.2%	37.4%
	AoA	**78.4%**	**49.0%**	**52.2%**	**39.6%**	**63.4%**	99.9%	**65.6%**	**51.1%**
DN121	CW	100.0%	33.5%	39.5%	31.9%	39.6%	64.6%	53.2%	39.4%
	PGD	100.0%	34.0%	41.7%	31.9%	41.5%	68.9%	55.5%	41.5%
	AoA	**100.0%**	**46.1%**	**53.5%**	**46.1%**	**55.0%**	**76.7%**	**64.6%**	**52.1%**
IncV3	CW	31.0%	22.7%	100.0%	21.3%	26.1%	42.3%	40.7%	33.4%
	PGD	32.7%	24.2%	100.0%	21.3%	27.3%	45.3%	40.7%	33.7%
	AoA	**39.0%**	**30.2%**	**100.0%**	**32.7%**	**34.0%**	**52.8%**	**45.9%**	**45.1%**
VGG19	CW	85.5%	62.0%	69.8%	62.7%	60.0%	77.8%	100.0%	68.0%
	PGD	87.1%	64.1%	71.8%	63.9%	63.1%	82.5%	100.0%	71.9%
	AoA	**91.4%**	**73.7%**	**79.8%**	**74.2%**	**73.5%**	**86.6%**	**100.0%**	**81.0%**
RN152	CW	42.4%	36.2%	35.3%	25.6%	100.0%	57.7%	46.0%	31.9%
	PGD	42.7%	35.0%	34.9%	24.5%	98.1%	55.3%	43.6%	30.5%
	AoA	**55.9%**	**54.2%**	**49.6%**	**36.4%**	**100.0%**	**71.5%**	**57.2%**	**45.6%**

Table 3–3 Error rate (Top-1) of different transfer attacks on ResNet50

Method	DN121	IncRNV2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
CW	66.6%	37.9%	39.6%	28.8%	47.3%	100.0%	54.2%	37.4%
MI-DI-CW	66.9%	39.4%	42.9%	32.3%	50.2%	99.8%	57.9%	39.9%
SI-CW	80.3%	46.4%	51.6%	38.3%	63.9%	99.9%	66.5%	48.8%
PGD	67.8%	35.6%	38.7%	28.4%	46.8%	100.0%	54.2%	37.4%
MI-DI-PGD	70.5%	43.3%	45.8%	35.7%	55.9%	99.5%	62.1%	43.3%
SI-PGD	81.2%	48.7%	53.0%	38.6%	66.1%	100.0%	69.5%	49.1%
AoA	78.4%	49.0%	52.2%	39.6%	63.4%	99.9%	65.6%	51.1%
MI-DI-AoA	74.1%	50.4%	52.0%	44.2%	58.7%	99.8%	66.4%	50.6%
SI-AoA	**90.5%**	**64.6%**	**66.1%**	**57.9%**	**78.8%**	**100.0%**	**80.4%**	**64.6%**

to evaluate the attack performance under several defenses. In this experiment, we consider PGD, CW, and AoA all combined with SI (given their best transferability) and use them to attack ResNet50. There have been lots of defenses, but many of them is not effective to large-scale dataset[101]. Hence, we only consider defenses that have been verified effective on ImageNet. Those defense methods can be roughly categorized as pre-processing and adversarial training, which could be used together.

Pre-processing based defenses are to eliminate the adversarial perturbation. Following this idea, JPEG Compression[71], Pixel Deflection[72], Total Variance Minimization (TVM)[102] have been proposed and we use the optimal parameters they provided. Another idea is to add randomness to observe the variance of the output. Random Smoothing[103] makes prediction by m intermediate images, which is crafted by Gaussian noise from the input. We choose $m = 100$ and the noise scale $\sigma = 0.25 * 255$.

Adversarial training is to re-train the neural networks by adversarial samples. In[104], InceptionV3adv and InceptionResNetV2adv are designed and ResNetXt101denoise[70] is proposed based on ResNetXt101[52] with denoising blocks in architectures to secure the model.

Table 3–4 gives the comprehensive black-box attack performance under defense. Generally speaking, the pre-processing defenses decrease the error rate for about 5% to 10% and SI-AoA maintains the highest transfer rate. adversarial-trained models (not by AoA, indicated by underlines) exhibit a strong robustness to adversarial attacks, including SI-AoA (but still, it is better than SI-PGD, SI-CW). That means although samples generated by SI-AoA are different to others, the distribution can still be captured by adversarial training. Developing adversarial attacks that can defeat adversarial training is interesting but out of our scope. Random smoothing generally has a low error rate but its interpretation time is generally m times than other defense methods, which is an unfair comparison. An observation is that the random smoothing does not work well in adversarial-trained models, sometimes even oppositely, which is also interesting but in the field of defenses.

3.5 DAmageNet

The above experiments verify that AoA has a very promising transferability, which then makes it possible to generate adversarial samples that are able to beat many well-trained classifiers. An adversarial dataset will be very useful for evaluating robustness and defense methods. To establish an adversarial dataset, we use SI-AoA to attack VGG19 to generate samples from all 50000 samples from ImageNet validation set. Since the original image samples come from ImageNet training set and the adversar-

Table 3–4 Error rate (Top-1) under defenses (ResNet50 as surrogate model)

Method	Victim	None	JPEG	Pixel	Random	TVM	Smooth
SI-CW	DN121	80.3%	64.9%	67.2%	64.5%	70.2%	60.0%
	IncRNV2	46.4%	38.0%	38.3%	40.3%	41.0%	31.7%
	InceptionV3	51.6%	43.2%	42.7%	46.2%	46.1%	33.5%
	NASNetLarge	38.3%	31.3%	32.4%	35.2%	34.0%	23.7%
	ResNet152	63.9%	51.4%	51.6%	48.9%	56.6%	41.2%
	ResNet50	99.9%	98.5%	98.7%	89.5%	99.6%	93.4%
	VGG19	66.5%	60.7%	60.6%	62.9%	63.3%	89.8%
	Xception	48.8%	40.6%	40.9%	44.0%	44.7%	36.5%
	IncV3adv	31.2%	33.8%	35.0%	38.1%	37.0%	96.5%
	IncRNV2adv	26.4%	27.4%	27.6%	30.1%	28.2%	81.7%
	RNX101denoise	18.0%	18.2%	18.2%	44.4%	18.1%	70.4%
SI-PGD	DN121	81.2%	65.1%	66.4%	64.0%	69.7%	60.0%
	IncRNV2	48.7%	39.8%	39.3%	40.0%	42.1%	31.8%
	InceptionV3	53.0%	44.8%	45.0%	47.9%	48.3%	32.6%
	NASNetLarge	38.6%	30.8%	31.5%	34.3%	34.6%	23.5%
	ResNet152	66.1%	52.8%	54.1%	51.5%	58.4%	40.2%
	ResNet50	100.0%	99.1%	99.4%	90.8%	99.6%	92.4%
	VGG19	69.5%	62.8%	61.4%	65.7%	65.2%	89.6%
	Xception	49.1%	40.8%	43.0%	43.5%	44.7%	37.1%
	IncV3adv	31.5%	34.3%	35.8%	39.2%	38.4%	96.2%
	IncRNV2adv	26.1%	27.9%	28.5%	29.7%	29.8%	81.5%
	RNX101denoise	18.2%	18.5%	18.9%	44.6%	18.4%	70.5%
SI-AoA	DN121	90.5%	81.0%	82.1%	78.0%	83.7%	63.4%
	IncRNV2	64.6%	56.7%	58.2%	57.8%	59.5%	34.6%
	InceptionV3	66.1%	62.3%	62.4%	62.9%	64.1%	37.5%
	NASNetLarge	57.9%	49.2%	53.0%	52.7%	53.0%	29.3%
	ResNet152	78.8%	70.3%	72.8%	67.1%	75.6%	44.2%
	ResNet50	100.0%	99.9%	99.8%	95.6%	99.9%	94.1%
	VGG19	80.4%	77.7%	78.5%	77.1%	79.8%	89.9%
	Xception	64.6%	57.6%	58.4%	61.1%	59.0%	40.9%
	IncV3adv	53.7%	52.7%	54.9%	55.1%	56.2%	96.2%
	IncRNV2adv	44.0%	44.2%	46.2%	48.0%	47.0%	82.3%
	RNX101denoise	18.7%	19.2%	19.1%	44.6%	19.0%	70.5%

ial samples are going to cheat and therefore damage well-trained neural networks in it, we name this dataset as DAmageNet.

DAmageNet contains 50000 adversarial samples and could be downloaded from https://pan.baidu.com/s/1WQpQTIt-PA2L-bIR6QF5yw by "098k". The samples are named the same as the original ones in ImageNet validation set. Accordingly, user could easily find the corresponding samples as well as its label. The average RMSE between samples in DAmageNet and those in ImageNet is about 7.231. In Fig. 3–4, we show part of the image pairs in ImageNet and DAmageNet.

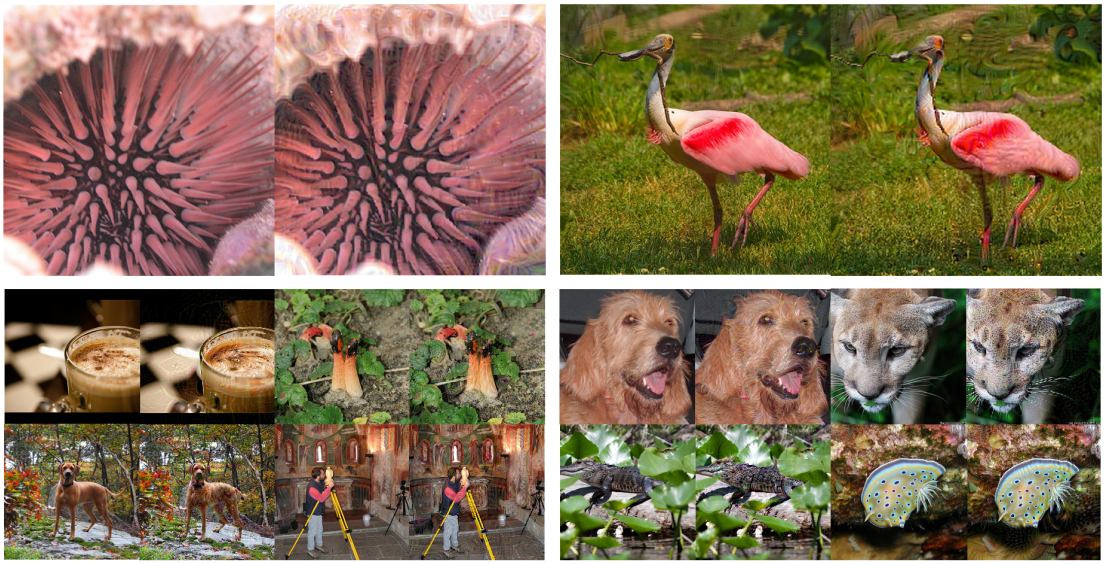

Figure 3–4 Samples in ImageNet and DAmageNet. The images on the left are original samples from ImageNet. The images on the right are adversarial samples from DAmageNet. One could observe that these images look similar and human beings have no problem to recognize them as the same class.

We use several well-trained models to recognize the images in DAmageNet. Several neural networks strengthened by adversarial training are considered as well. The error rate (top-1) is reported in Table 3–5. The models are from Keras Application and the test error may differ from original references. DAmageNet increases the error rate of all 13 undefended models to over 85% and that of 5 direct adversarial-trained models to over 70%. Moreover, DAmageNet resists 4 tested defenses with almost no drop on the error rate compared to other methods in Table 3–4. To the best of our knowledge, this is the first adversarial dataset, it can be used to evaluate model robustness and the effect of defenses.

Table 3–5 Error rate (Top-1) on ImageNet and DAmageNet

	No defense		Defenses on DAmageNet			
Victim	ImageNet	DAmageNet	JPEG	Pixel	Random	TVM
VGG16[1]	38.51	99.85	99.67	99.70	99.19	99.76
VGG19[1]	38.60	99.99	99.99	99.99	99.96	99.99
ResNet50[2]	36.65	93.94	91.88	92.48	92.52	93.08
ResNet101[2]	29.38	88.13	85.44	86.23	86.12	87.06
ResNet152[2]	28.65	86.78	83.93	84.83	84.71	85.68
NASNetMobile[98]	27.03	92.81	90.42	91.43	90.31	91.86
NASNetLarge[98]	17.77	86.32	83.31	84.87	84.91	85.53
InceptionV3[50]	22.52	89.84	87.82	89.01	88.49	89.59
IncRNV2[97]	24.60	88.09	85.01	85.95	89.04	86.79
Xception[105]	21.38	90.57	88.53	89.77	86.03	90.32
DenseNet121[3]	26.85	96.14	93.96	94.85	93.82	95.30
DenseNet169[3]	25.16	94.09	91.72	92.78	91.78	93.36
DenseNet201[3]	24.36	93.44	90.52	91.71	90.86	92.45
IncV3adv[104]	22.86	82.23	82.03	83.35	82.88	83.95
IncV3advens3[106]	24.12	80.72	80.35	81.68	81.57	82.36
IncV3advens4[106]	24.45	79.26	78.86	79.96	79.76	80.8
IncRNV2adv[104]	20.03	76.42	75.71	76.85	76.86	77.73
IncRNV2advens[106]	20.35	70.70	71.09	72.32	73.32	73.04
RNX101denoise[70]	32.20	35.40	36.27	36.65	55.53	36.21

3.6 Summary

To improve the transferability of adversarial attack, we are the first to attack on attention and achieve a great performance on black-box scenarios. Success of AoA relies on the semantic features shared by different DNNs. To effectively attack on attention, we apply network visualization in designing the attention loss. AoA enjoys a significant increase in transferability when the traditional cross entropy loss is replaced with the attention loss. Since AoA alters loss function only, it could be easily combined with other transfer-enhancement attacks and achieve a state-of-the-art performance.

By SI-AoA, we generate DAmageNet, the first dataset containing samples with a small perturbation and a high black-box transfer rate (an error rate over 85% for undefended models and over 70% for direct adversarial-trained models). DAmageNet provides a benchmark to evaluate the robustness of DNNs by elaborately-crafted samples.

Chapter 4 Attack on Attention with Query

4.1 Method

The proposed AoA in the last chapter enjoys a great transferability across models. However, it does not use the output information of the victim model, which belongs to the query approach black-box attack. In this section, we further propose the variants of AoA, i.e., AoA-N and AoA-L. By querying the black-box model, we found that they achieve an amazing transferability in the targeted case and the non-targeted case.

Suppose the attacked surrogate neural network is f. Then for an input x_{ori} with the label y_{ori}, an adversarial sample x_{adv} could be generated basically by i) minimizing the distance between y_{tar} and $f(x_{\text{adv}})$, where y_{tar} is the attack target label, in the targeted-case; ii) maximizing the distance between $f(x_{\text{adv}})$ and $f(x_{\text{ori}})$ in the non-targeted case, both in the constraint that $\|x_{\text{ori}} - x_{\text{adv}}\|$ is small.

Different to the existing attacks that focus on the final output, the proposed method aims to change the attention heat map. Specifically, let $h(x, y)$ stand for the attention heat map for the input x and the specified class y. The basic idea of our attack is to *shift* the attention away from the original class (non-targeted) or close to the targeted class (targeted). The corresponding loss is hence called *shift loss*. In this thesis, we utilize SGLRP[38] to calculate the attention heat map $h(x, y)$. There are of course many techniques for obtaining the heat map, as long as $h(x, y)$ and its gradient on x could be effectively calculated.

In non-targeted attack, we do not have a clear target, so we push the adversarial class away from the original class to the class with the highest prediction probability. Specifically,

$$p_{\text{non}}(x) = \frac{||h(x, y_{\text{ori}})||_1}{||h(x, y_{\overline{\text{ori}}}(x))||_1}, \tag{4–1}$$

where $y_{\overline{\text{ori}}}(x)$ is the class with highest prediction probability for x, except of y_{ori}. In this case, we encourage $||h(x, y_{\text{ori}})||_1$ to be weakened relative to $||h(x, y_{\overline{\text{ori}}}(x))||_1$, so $p_{\text{non}}(x)$ is desired to be small.

In targeted attack, we need to draw the adversarial class close to the targeted class and hence we design as the following,

$$p_{\text{tar}}(x) = \frac{||h(x, y_{\text{tar}})||_1}{||h(x, y_{\overline{\text{tar}}}(x))||_1}, \tag{4–2}$$

where $y_{\overline{\text{tar}}}(x)$ is the class with highest prediction probability for x, except of y_{tar}. We

encourage $||h(x, y_{\text{tar}})||_1$ to be strengthened relative to $||h(x, y_{\overline{\text{tar}}}(x))||_1$, so $p_{\text{tar}}(x)$ is desired to be large.

There are two phases for the targeted attack. In phase I, starting from the original label, $y_{\overline{\text{tar}}}(x) = y_{\text{ori}}$ and then maximizing p_{tar} has similar performance as in the non-targeted case that we encourage the heat map to differ from the original. In phase II, $y_{\overline{\text{tar}}}(x) \neq y_{\text{ori}}$, where the prediction is already incorrect, but it is not necessarily the target label. Then maximizing p_{tar} is actually to enhance $h(x, y_{\text{tar}})$ until the targeted attack is achieved.

Notice that we use proportion, not subtraction, to measure the distance because proportion has a good transferability across DNNs, which may have different absolute values of heat maps. However, using the proportion makes the gradient quite small when $p(x) \leq 1$. Therefore, we turn to optimize $-1/p(x)$ in that case and use the sigmoid membership function to connect the two segments, i.e.,

$$\begin{aligned} v(p) &= p\beta(p) - \frac{1}{p}(1 - \beta(p)), \\ \beta(p) &= h_{\text{sigmoid}}(p - 1). \end{aligned} \tag{4–3}$$

Finally, we design activation functions on $v(p)$ and obtain two versions of shift loss (NRU and LLU) as follows

- Normal Rectified Unit (NRU):

$$F_{\text{shift-n}}(p) = (1 + 4e^{-\frac{(p-1)^2}{8}})v(p).$$

- Log Linear Unit (LLU):

$$F_{\text{shift-l}}(p) = \begin{cases} log(v(p) + 1) & v(p) \geq 0 \\ -log(-v(p) + 1) & v(p) < 0. \end{cases}$$

In our experience, NRU is good for restricting the change magnitude and LLU is a better choice for a high attack success rate. Other activation functions are also possible.

Now we summarize the possible loss functions $L(x)$ in the AoA-N and AoA-L.

- Non-targeted case: we will minimize the shift loss, which has two choices

$$\begin{aligned} L_{\text{AoA-N}}(x) &= F_{\text{shift-n}}(p_{\text{non}}(x)), \\ L_{\text{AoA-L}}(x) &= F_{\text{shift-l}}(p_{\text{non}}(x)). \end{aligned} \tag{4–4}$$

- Targeted case: we will maximize the shift loss, as the following

$$\begin{aligned} L_{\text{AoA-NT}}(x) &= -F_{\text{shift-n}}(p_{\text{tar}}(x)), \\ L_{\text{AoA-LT}}(x) &= -F_{\text{shift-l}}(p_{\text{tar}}(x)). \end{aligned} \tag{4–5}$$

The adversarial samples are generated in an update process by minimizing a loss function L. Specifically, set $x_{\text{adv}}^0 = x_{\text{ori}}$ and the update procedure could be generally described as the following

$$\begin{aligned} x_{\text{adv}}^{k+1} &= x_{\text{adv}}^k - \alpha \frac{g(x_k)}{||g(x_k)||_1/N}, \\ g(x) &= \frac{\partial L(x)}{\partial x}. \end{aligned} \tag{4–6}$$

The gradient g is normalized by its average l_1-norm, i.e., $||g(x_k)||_1/N$ where N is the size of the image. By iterative update, we could gradually change the image x, until attack succeeds, i.e., $\arg\max f(x) \neq y_{\text{ori}}$ in non-targeted case or $\arg\max f(x) = y_{\text{tar}}$ in targeted attack.

For the attacking step length α in (4–6), there are some differences given the attack purposes. For non-targeted attack, we simply keep a constant α. But the targeted attack needs careful modification since the target and the original heat map could be quite different. In our experiments, we set the following adaptive schemes: i) when the white-box attack is achieved, i.e., $\arg\max f(x) = y_{\text{tar}}$, the step length is doubled to across the decision boundary of black-box models, because the gradient now is towards the right direction. ii) when the loss value is smaller than a pre-given threshold ζ, the step length is halved, because the gradient may not be precise.

AoA-N and AoA-L could be used for black-box attack for its good transferability on heat maps. The basic scheme is to choose a (white-box) surrogate DNN, then update x by (4–6), and the generated adversarial samples could be used for black-box attack on other models. A good attack needs to change the label with small distortion, for which there are three black-box attack strategies.

1. Gradually update x until the black-box victim model changes the decision.
2. Set a referred black-box model and stop when the referred model changes the decision.
3. Set a threshold for the number of update iteration or the magnitude of distortion. Stop the update when the threshold is achieved.

Strategy 1) is actually a decision-based black-box attack with few queries. Strategy 2) and 3) are zero-query black-box attacks. In few-query black-box attack, we choose strategy 1) for a good trade-off between distortion magnitude and attack success rate. In zero-query black-box attack, we choose strategy 2) where one needs an surrogate model and a referred model for query to generate images that are then used to attack other models in the zero-query manner.

4.2 Experiments

In this section, we will evaluate the performance of our variants of Attack on Attention, especially their black-box attack capability. Since AoA-N and AoA-L are very good black-box attacks, they provide adversarial samples that can defeat many DNNs in the zero-query manner. These samples are collected in the dataset DAmageNet2. This section will also introduce DAmageNet2 and report the performance of different DNNs on it.

The experiments are conducted on ImageNet. For attack and test, several well-trained models in Keras Applications[99] are used, including VGG19[1], InceptionV3[50], NASNetLarge[98], Xception[105], DenseNet121[3]. We also use CondenseNet[107] and other adversarial-trained models. We pre-process with Keras preprocessing function, central cropping and resizing (to 224). The experiment is implemented in TensorFlow[100], Keras[99] with 4 NVIDIA GeForce RTX 2080Ti GPUs.

For the attack performance, we care about two aspects: the success rate of attack and how large the image is changed. Denote the generated adversarial sample as x_{adv} with size N. The change from the original image x_{ori} could be measured by$d\left(x_{\text{adv}}, x_{\text{ori}}\right) = \sqrt{\sum \left(x_{\text{adv}} - x_{\text{ori}}\right)^2 / N}$, i.e., the Root Mean Squared Error (RMSE) in each pixel.

In AoA-N, we set $\alpha = 1$ in (4–6). In AoA-L, we choose $\alpha = 2$ in (4–6), and $\zeta = -15$. The attack will be stopped when the iteration exceeds 100. For attack performance evaluation, we randomly choose 100 images. The target labels in targeted case are also chosen randomly.

4.3 Transferability of AoA-N and AoA-L

We first consider the transferability of AoA variants in ImageNet validation set with several queries of decision only, i.e., we will attack in Strategy 1). The average of the success rate, the difference (root mean squared deviation), and the number of queries of AoA-L are reported in Table 4–1, 4–3. The query time is below 8 for the non-targeted attack and below 30 for the targeted attack. Very recent researches[18, 20] still require query times of over 50 to guarantee a high success rate. Note that we only query the output label, not the probabilities.

From Table 4–1, 4–3, one could find that VGG19 is a good candidate for generating transferable samples with small distortion, which also coincides with our guess from Fig. 3–3. For VGG19, we also try AoA-N and report the performance in Table 4–2 and Table 4–4 (unanimity requires the same false prediction). Generally speaking, AoA-N can also achieve a good success rate in non-targeted and targeted case. Although the

Table 4–1 AoA-L performance on non-targeted case

Surrogate	Victim	rate	RMSE	# query
VGG19 [1]	VGG19[1]	100%	1.776	—
	DenseNet121[3]	100%	10.27	3.12
	ResNet152[2]	100%	10.36	3.21
	InceptionV3[50]	100%	9.595	2.61
	NASNetLarge[98]	100%	11.08	3.32
	Xception[105]	100%	10.72	3.20
	InceptionResNetV2[97]	100%	10.89	3.26
InceptionV3 [50]	VGG19[1]	99%	15.77	6.52
	DenseNet121[3]	100%	15.27	6.61
	ResNet152[2]	99%	17.25	7.79
	InceptionV3[50]	100%	2.063	—
	NASNetLarge[98]	100%	15.34	5.91
	Xception[105]	100%	13.12	5.04
	InceptionResNetV2[97]	98%	14.72	6.10
DenseNet121 [3]	VGG19[1]	100%	12.43	3.69
	DenseNet121[3]	100%	1.833	—
	ResNet152[2]	100%	13.38	4.26
	InceptionV3[50]	100%	13.99	4.76
	NASNetLarge[98]	100%	16.24	6.26
	Xception[105]	100%	13.43	4.43
	InceptionResNetV2[97]	100%	14.54	4.53

Table 4–2 AoA-N performance on non-targeted case

		Non-Targeted success			Non-Targeted unanimity		
Surrogate	Victim model	rate	RMSE	# query	rate	RMSE	# query
VGG19 [1]	VGG19[1]	100%	1.717	—	—	—	—
	DenseNet121[3]	89%	4.128	7.17	70%	4.726	9.32
	ResNet152[2]	87%	4.042	7.24	72%	4.517	8.67
	InceptionV3[50]	96%	4.224	7.52	86%	4.791	9.88
	NASNetLarge[98]	88%	4.097	6.72	81%	4.699	9.09
	Xception[105]	98%	3.717	6.17	92%	4.153	7.62
	InceptionResNetV2[97]	85%	4.525	8.41	77%	5.064	10.47

Table 4–3 AoA-L performance on targeted case (AoA-LT)

Surrogate	Victim	rate	RMSE	# query
VGG19 [1]	VGG19[1]	91%	6.9191	—
	DenseNet121[3]	89%	10.826	12.95
	ResNet152[2]	87%	11.816	13.23
	InceptionV3[50]	88%	10.432	12.80
	NASNetLarge[98]	91%	11.448	13.45
	Xception[105]	96%	10.621	13.01
	InceptionResNetV2[97]	91%	11.345	12.34
InceptionV3 [50]	VGG19[1]	92%	18.814	24.05
	DenseNet121[3]	96%	21.175	27.27
	ResNet152[2]	88%	24.460	32.57
	InceptionV3[50]	97%	9.0882	—
	NASNetLarge[98]	84%	21.136	26.82
	Xception[105]	87%	17.227	20.85
	InceptionResNetV2[97]	96%	22.716	29.36
DenseNet121 [3]	VGG19[1]	92%	12.012	14.86
	DenseNet121[3]	94%	5.7744	—
	ResNet152[2]	90%	13.544	13.80
	InceptionV3[50]	94%	12.917	12.17
	NASNetLarge[98]	94%	15.234	14.52
	Xception[105]	95%	13.678	14.27
	InceptionResNetV2[97]	92%	14.927	13.46

Table 4–4 AoA-N performance on targeted case (AoA-NT)

		Targeted success			Targeted unanimity		
Surrogate	Victim model	rate	RMSE	# query	rate	RMSE	# query
VGG19 [1]	VGG19[1]	78%	6.967	—	—	—	—
	DenseNet121[3]	70%	6.865	17.64	50%	7.345	17.65
	ResNet152[2]	71%	7.571	19.87	31%	8.406	21.51
	InceptionV3[50]	71%	6.052	14.90	51%	7.395	19.80
	NASNetLarge[98]	66%	6.967	19.08	66%	7.226	18.80
	Xception[105]	68%	6.356	15.94	54%	6.654	16.89
	InceptionResNetV2[97]	67%	6.614	16.70	43%	6.930	17.11

success rate is not as high as AoA-L, the perturbation is smaller. Therefore, AoA-N is a good choice for generating transferable adversarial samples. Notice that most of the attack failures could be found during our white-box attack procedure. Therefore, in the adversarial dataset, we can pre-select and only provide samples that can cheat DNNs, which is the reason why the attack success rates reported in the next subsection are higher than that in Table 4–2 and Table 4–4.

4.4 DAmageNet2

The above experiments show that AoA-N and AoA-L have a very promising transferability such that we could also generate adversarial samples able to beat many well-trained DNNs. Accordingly, we choose VGG19 as the surrogate model and InceptionV3 as the referred model in Strategy 2). For the loss, we choose the non-targeted AoA-N since we pursue a small distortion in the adversarial dataset. The distortion is restricted by limiting maximum iteration of attack to 20 and the step length $\alpha = 1$.

To guarantee the quality of the dataset, we manually discard samples with a low transferability, i.e., the samples that cannot fool InceptionV3 to produce the same false label as VGG19 or that are correctly predicted by any networks in NASNetLarge, InceptionResNetV2, Xception, DenseNet121. The image selection scheme and conditions of query can be summarized below. Except VGG19, which is used as the surrogate model to generate adversarial samples, InceptionV3, which is used as the referred model for the stop condition, and NASNetLarge, InceptionResNetV2, Xception, DenseNet121, which are queried once for checking the attack performance, other DNNs are only used as victim models for test in the zero-query attack manner.

Since the original images are coming from ImageNet training set and the adversarial samples are going to cheat neural networks, we name this dataset as DAmageNet2. The samples in DAmageNet2 are very similar to those in ImageNet training set and the average root mean square deviation is about 4.2. In Fig. 4–1, we demonstrate part of the image pairs in ImageNet and DAmageNet2.

DAmageNet2 contains 96020 adversarial samples in total and could be downloaded from https://pan.baidu.com/s/1qhmd4-M0cb6XDLB7rJhC3Q by code “d3cq” (separately-packed .tar files). Directories include 1000 folders with the correct labels as the folder names, which are in the same order of ImageNet. In each folder, one can find around 100 adversarial samples for this class. The file name is the same as that of the original image in ImageNet, with which one can find the corresponding sample.

DAmageNet2 provides adversarial samples that can cheat many DNNs. Here, we use several well-trained models to recognize the images in DAmageNet2 and ImageNet.

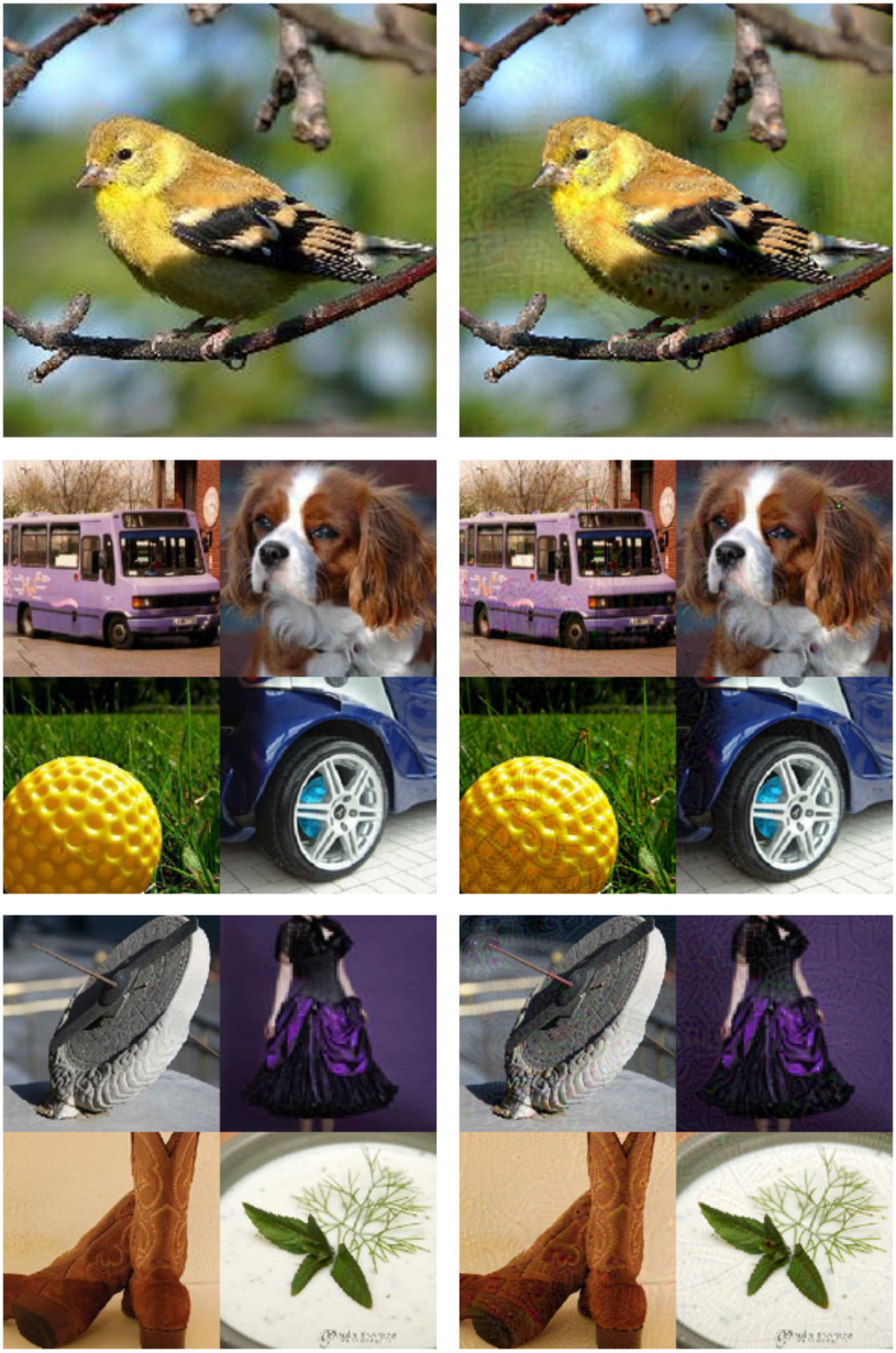

Figure 4–1 Samples in ImageNet and DAmageNet2. The images on the left column are original samples from ImageNet. The images on the right column are adversarial samples from DAmageNet2. One could observe that these images look similar and human beings have no problem to recognize them as the same class.

For ImageNet, we only consider the images that generate DAmageNet2 in order to show the attack performance. For the error rates on the whole ImageNet, please refer to the references. Besides, several neural networks strengthened by adversarial training are considered as well. The error rate (top-1) is reported in Table 4–5. These samples that DNNs fail to recognize reveal common vulnerabilities of DNNs.

Table 4–5 Error rate (Top-1) on ImageNet and DAmageNet2

Victim model	ImageNet	DAmageNet2
VGG16[1]	12.6%	99.7%
VGG19[1]	5.1%	99.9%
ResNet50[2]	11.4%	92.5%
ResNet101[2]	17.3%	84.6%
ResNet152[2]	16.6%	81.8%
NASNetMobile[98]	13.2%	90.3%
NASNetLarge[98]	4.8%	99.9%
InceptionV3[50]	6.4%	96.7%
InceptionResNetV2[97]	11.7%	99.9%
Xception[105]	8.8%	99.9%
DenseNet121[3]	15.2%	99.9%
DenseNet169[3]	10.8%	94.3%
DenseNet201[3]	9.5%	91.6%
CondenseNet74-4[107]	18.3%	95.5%
CondenseNet74-8[107]	22.5%	93.1%
Inception$V3_{adv}$[104]	3.8%	77.1%
Inception$V3_{advens3}$[104, 106]	6.9%	72.6%
Inception$V3_{advens4}$[104, 106]	7.4%	67.4%
InceptionResNet$V2_{adv}$[104]	2.3%	60.4%
InceptionResNet$V2_{advens}$[104, 106]	4.0%	52.7%

Based on the high transferability of common attention features shared by DNNs, AoA-N successfully provides DAmageNet2, an improved version of DAmageNet with a great transferability on different architectures. We are curious about the reason of its aggression and whether it can be mitigated by defense methods although our main contribution is for producing attack transferability instead of breaking the defense.

In the recent years, there is great progress on adversarial defense methods in pre-processing, structure modification and adversarial training. Here, we only adopt popular pre-processing defenses given its practicability. The error rate (top-1) of the secured DNNs on DAmageNet2 is reported in Table 4–6. Although there are some improvements in the accuracy due to the defense, the results are far from satisfactory.

Table 4–6 Error rate (Top-1) on DAmageNet2 with defense methods

Victim model	JPEG[71]	Pixel[72]	Super[73]	Random[108]	TVM[102]
VGG16[1]	99.5%	98.9%	98.5%	99.8%	99.7%
VGG19[1]	99.9%	99.9%	99.9%	99.9%	99.9%
ResNet50[2]	85.2%	83.6%	80.6%	86.5%	89.8%
ResNet101[2]	75.3%	73.9%	71.5%	76.6%	80.3%
ResNet152[2]	72.2%	71.0%	68.4%	73.8%	77.2%
NASNetMobile[98]	80.7%	79.8%	76.1%	82.6%	85.7%
NASNetLarge[98]	75.9%	76.4%	70.6%	82.4%	85.8%
InceptionV3[50]	75.6%	75.4%	71.0%	80.7%	83.9%
InceptionResNetV2[97]	80.7%	78.9%	74.8%	83.4%	88.0%
Xception[105]	85.3%	84.3%	80.2%	86.9%	91.9%
DenseNet121[3]	90.8%	89.9%	86.5%	91.5%	95.0%
DenseNet169[3]	84.7%	83.7%	80.1%	87.4%	90.2%
DenseNet201[3]	80.5%	79.6%	76.0%	83.6%	86.6%
CondenseNet74-4[107]	88.6%	87.9%	85.2%	95.1%	92.3%
CondenseNet74-8[107]	87.1%	86.4%	84.1%	93.0%	90.7%
IncV3adv[104]	67.0%	65.6%	62.2%	69.0%	73.4%
IncV3advens3[104, 106]	63.6%	62.1%	59.4%	65.1%	69.6%
IncV3advens4[104, 106]	59.4%	58.0%	55.5%	60.9%	64.9%
IncRNV2adv[104]	52.9%	51.0%	48.3%	55.5%	58.1%
IncRNV2advens[104, 106]	47.8%	46.1%	44.3%	50.2%	52.2%

4.5 Summary

To further improve the transferability of adversarial attack, we propose two variants of AoA in this section and evaluate them with query. Experiments show that AoA-N and AoA-L have a promising transferability in both the non-targeted case and the targeted case. By AoA-N, we generate DAmageNet2, which is larger than DAmageNet in the last chapter and could induce DNNs to have a higher error rate with a smaller perturbation. DAmageNet2 provides a benchmark to evaluate the robustness of DNN by elaborately-crafted adversarial samples.

Chapter 5 Attack on Attention for Object Detection

In this chapter, we extend attack on attention to object detection. Since detection networks are broadly implemented in security system, attack them in black-box manner would be more aggressive. However, the attention for detection is quite different from that for classification. Accordingly, we propose a novel network visualization method, named Multi-Node Softmax Gradient Layer-wise Relevance Propagation (MN-SGLRP), which is the first visualization method for object detection. Then based on it, we design our attack, named ATTACk on multi-node attenTION for object detecTION network (ATTACTION). Experiments show that ATTACTION has state-of-the-art transferability across models and tasks. A new dataset, Adversarial Objects in COntext (AOCO), is also created for robustness testing.

We propose an attack specifically designed for black-box transferability, named ATTACk on multi-node attenTION for object detecTION (ATTACTION). ATTACTION works by suppressing multi-node attention for several bounding boxes. Since the attention heat map is commonly shared by different architectures as shown in Fig. 1–2, attacking on it in the white-box surrogate model achieves a high transferability towards black-box models. Below we first analyse why we do not adopt the high-transferable single-node attack on attention in classification. Then we present our ATTACTION framework and introduce its details by addressing three crucial issues.

5.1 Why Attention and Why Not Single-Node Attetnion

Classification nets[1-3, 97] generally output a vector y. The decision is formed by the largest probability $\arg\max(y)$, which is single-node information. So, we propose to attack on single-node attention (AoA), i.e., suppress the attention for the correct class, and achieve a good transferability.

However, it does not work well in attacking detection nets given their large difference compared to classification ones. In contrast to classification nets, predictions of detection nets[4, 7, 8, 109], or the messages presented to the users, are formed by the location and confidence of several bounding boxes, which is multi-node information. The necessity of attacking multi-node output in detection nets has been discussed[12] and many works also validate that[34, 75, 76].

According to the analysis above, it can be expected that single-node attention attacks in classification are ineffective for attacking detection networks. Experimental validation of this is presented in the Appendix.

5.2 Who is ATTACTION

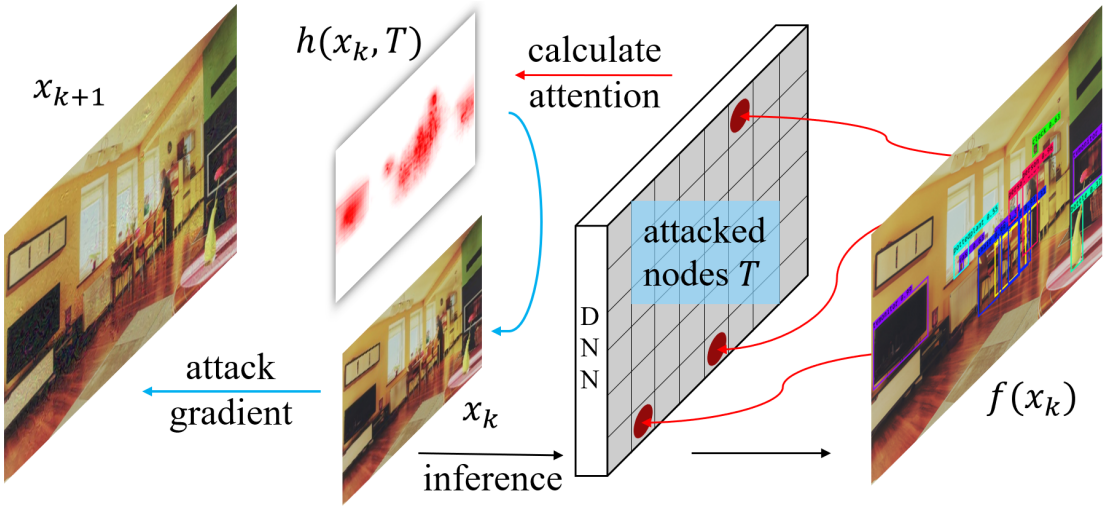

Figure 5–1 Framework of ATTACTION. x_k is the sample in iteration k and $f(x_k)$ is the network prediction for it. MN-SGLRP calculates the network attention heat map $h(x_k, T)$ for certain target nodes in set T. Gradients of $h(x_k, T)$ back propagate to x_k, which is then modifies it to x_{k+1}.

To achieve a high transferability in attacking object detection, we propose to ATTACk on multi-node attenTION for object detecTION (ATTACTION). We present the framework of ATTACTION in Fig. 5–1. Starting from the original sample x_0, the input image x_k in the k-th iteration forward propagates in the surrogate model and gets the prediction $f(x_k)$. Then the network visualization method calculates the attention heat map $h(x_k, T)$ for all attacked nodes in set T. Gradients of $h(x_k, T)$ back propagate through the whole route to x_k, which is then modified to x_{k+1}.

In this framework, there are three crucial issues: (i) how to get the attention heat map $h(x, T)$; (ii) how to update the sample x by gradients of $h(x, T)$; (iii) how to choose the multiple nodes to attack.

5.3 What is the Attention Heat Map for Object Detection

Among the visualization methods for classification[36, 37], SGLRP[38] excels in discriminating ability against irrelevant regions of a certain target node. It visualizes how the input contributes to one output node by back-propagating the relevance from the output to the input. R is the initial relevance in the output layer and its n-th component is calculated as

$$R_n = \begin{cases} y_n \left(1 - y_n\right) & n = t \\ -y_t y_n & n \neq t, \end{cases} \tag{5–1}$$

where y_n is the predicted probability of class n, and y_t is that for the single-node target t. The pixel-wise attention heat map $h(x,t)$ for the single-node target t is calculated by back propagating the relevance $\boldsymbol{R}$ from the final layer to the input with certain rules[38].

In detection nets, we mostly wonder how the input contributes to m bounding boxes. This multi-node attention could not be directly calculated by (5–1). The intuitive idea of simply adding m single-node heat maps is inefficient and not applicable. Since in this way, the heat map needs to be calculated m times for m single-node targets. Besides, the added heat map is different from the multi-node attention. Because in SGLRP, the "irrelevant" regions for one target node (all $n \neq t$ in (5–1)) are suppressed, which may be the relevant regions for other target nodes. Adding all single-node heat maps together actually suppresses all relevant regions for $m-1$ times but only highlights it for 1 time, which is not our intention to calculate the multi-node attention.

Here we present our Multi-Node Softmax Gradient Layer-wise Relevance Propagation (MN-SGLRP), the first network visualization method for detection networks and therefore contribute to the high-transferable attack, which is modified from SGLRP as

$$R_n = \begin{cases} y_n\left(1-y_n\right) & n \in T \\ -\frac{1}{m}\sum_{i=1}^{m} y_{t_i} y_n & n \notin T, \end{cases} \tag{5–2}$$

where y_{t_i} is the predicted probabilities for one target node t_i. T is the set containing all target nodes $\{t_1, t_2, ..., t_m\}$. In this way, relevance for all target nodes is calculated in one time as we desire. Therefore, the heat map is obtained very quickly, i.e., m times faster than adding single-node ones.

We illustrate the difference between single-node SGLRP and our multi-node MN-SGLRP in Fig. 5–2. SGLRP only displays the network attention for one node of one bounding box, e.g. TV, chair, bottle, as shown in the right 3 images. MN-SGLRP, in contrast, visualizes the overall attention for several nodes. It could also demonstrate the relative attention strength for different nodes, e.g., the attention for bottle is smaller than that for TV as shown in the second image.

Figure 5–2 Difference between heat maps from SGLRP and MN-SGLRP. The heat maps are for YOLOv3[109]. For SGLRP, we choose the object confidence nodes for the predicted bounding boxes "TV", "chair" and "bottle" for demonstration. For MN-SGLRP, 20 object confidence nodes with the highest values are chosen.

5.4 How to Update the Sample

For the update, we choose the simple one in our baseline as

$$\begin{aligned} x_{k+1} &= \mathrm{clip}_{\varepsilon}\left(x_k - \alpha \frac{g(x_k)}{||g(x_k)||_1/N}\right), \\ g(x) &= \frac{\partial h(x,T)}{\partial x}, \end{aligned} \tag{5–3}$$

where α stands for the step length. x is l_∞-norm bounded by ε from the original sample in each iteration. Gradient $g(x)$ is normalized by its average l_1-norm,i.e., $||g(x)||_1/N$ to prevent numerical errors and control the degree of perturbation. N is the dimension of the image. In Section 5.6, we will show that the transferability of ATTACTION is further increased from the baseline by adopting other update formulas[23, 25, 26].

5.5 Where to Attack

It is important to choose a proper node set T to attack. Choices of localization or classification nodes are described in[13]. Attack that adopts the localization loss moves or shrinks the bounding boxes. Attacking by the classification loss, in contrast, leads the bounding boxes to be in a different class or even disappear.

Although these two losses are coupled by the shared layers and the Non-Maximum Suppression in the output[13]. We observe that they differ in the performance as losses for attack. The results are presented in Table 5–1, where Dfool[75] uses the classification loss and Loc adopts the localization loss. Generally, localization loss leads to a greater drop in white-box cases, but the classification loss induces better black-box transferability. Accordingly, we attack m classification nodes T with the highest confidence.

Intuitively, a large m leads to attack on more bounding boxes, which, meanwhile, leads to a large GPU memory occupation. A smaller m may cause a focus on fewer bounding boxes. In the extreme case when $m = 1$, the multi-node ATTACTION is reduced to single-node AoA. To trade off, we suggest $m = 20$ empirically.

5.6 Experiments

In the sections below, we evaluate the performance of ATTACTION, especially its transferability. The results are presented numerically and visually. In comprehensive evaluation, ATTACTION achieves a great transferability in across models and even across tasks. Furthermore, ATTACTION is flexible be easily combined with existing transfer-enhancing techniques for a better performance.

Our experiments are conducted on Keras[99], Tensorflow[100] and PyTorch[110] in 4 NVIDIA GeForce RTX 2080Ti GPUs. Library iNNvestigate[111] is used to implement

MN-SGLRP.

We experiment on MS COCO 2017 dataset[39], which is a large-scale benchmark for object detection, instance segmentation and captioning. It contains over 118K training images and 5K validation samples. For a fair evaluation, we generate adversarial samples from all 5K samples in its validation set each time and test several black-box models on their mAP, a standard criteria in many works[4, 7, 8]. mAP is calculated by APIs provided in[39]. All attacks are conducted with the step size $\alpha = 2$ for 10 iterations and the perturbation is l_∞-bounded in $\varepsilon = 16$ to guarantee the imperceptibility.

The black-box well-trained models from MMDetection[112] are M1 (SSD512[113]), M4 (Faster R-CNN[4]), M6 (Cascade R-CNN[6]), M7 (Cascade Mask R-CNN[6]), M8 (Hyrbrid Task Cascade[8]). We choose the best backbones for all models we use in MMDetection and specify them in the Supplementary Material. The surrogate models we attack include M2[114] (YOLOv3[109]), M3[115] (RetinaNet[116]) and M5[117] (Mask R-CNN[7]) as representations for single-task models and multi-task ones. To pre-process, we resize the image with its long side as 416 for YOLOv3 or RetinaNet and 448 for Mask R-CNN, and then zero-pad it to a square. The resolution is kept relatively the same for a fair evaluation. Images are normalized to [0,1] in YOLOv3 or subtracted by the mean of COCO training set in RetinaNet and Mask R-CNN.

To validate that the aggression comes from the attack method rather than resizing or random perturbation, we add the Gaussian noise ($\sigma = 9$) to images resized to 416 and make the perturbation larger than any other experiments. This ablation results are reported as "Ablation". The original mAP for MS COCO validation set provided in MMDetection is reported as "None" (attack).

5.7 Transferability of ATTACTION

We first evaluate the transferability of ATTACTION baseline with other detection attacks in the same setting. For DAG[12], we follow the setting of generating dense proposals. The classification probabilities of 3000 bounding boxes with highest confidence are attacked. But we alter its optimization to (5–3) because its original update produces quite small perturbation, leading to a poor transferability, which is unfair for comparison. Dfool[75] suppresses the classification confidence for the original bounding boxes, which is the same in our experiment. Localization loss is shown to be useful in[13], and here we suppress the width and height of the original bounding boxes. In ATTACTION, there are rare cases (less than 5% iterations) when the attention is too small to calculate the gradient. In these cases, we use DAG loss, and the gradient recovers immediately.

We present the mAP of several victim networks in adversarial samples crafted by

attacking M2 (YOLOv3[109]) with different methods in Table 5–1. It could be seen that ATTACTION enjoys a state-of-the-art transferability towards 7 black-box models, outperforming other methods for about 10% (2 to 3 mAP). ATTACTION also maintains a good performance in white-box attack, but is not the best since it is specifically designed for black-box scenarios. The mAP50 and mAP75 follow the same trend and are reported in the Appendix. In ablation experiment in the second row, we find that the aggression of adversarial samples comes from method rather than the perturbation.

Table 5–1 mAP of object detection in different attacks on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
None	29.3	33.4	38.1	40.7	42.1	42.5	45.7	46.9
Ablation	24.9	31.4	31.2	31.6	35.0	34.3	37.5	38.8
Dfool	23.3	2.5	29.2	29.8	33.3	32.9	36.5	38.0
Loc	21.9	**0.2**	25.8	26.6	29.8	29.4	33.2	33.2
DAG	20.8	0.6	22.8	23.4	26.8	25.6	28.9	31.0
ATTACTION	**18.1**	1.2	**19.9**	**20.5**	**24.3**	**22.6**	**26.4**	**28.2**

5.8 Transferability with Transfer-Enhancing Techniques

Some techniques are validated to be effective in enhancing the transferability in classification. Among them, Diverse Input (DI)[25], Translation-Invariant (TI)[23] and[26] Scale-Invariant (SI) are three of the best ones. We are curious about whether they also work well in object detection.

In our experiment, DI[25] transforms 4 times with probability p ($p = 1$ for better transferability as suggested) and averaging the gradients. SI[26] divides the sample numerically by the power 2 for 4 times before gradient calculation. TI[23] adopts a kernel size of 15 as suggested.

From the results in Table 5–2, we discover that DI and TI do not have a significant increase of transferability in object detection. In comparison, SI is quite effective, further decreasing the mAP from the baseline. The mAP50 and mAP75 follow the same trend as in the Appendix.

5.9 Transferability to Semantic Segmentation

Detection and segmentation are similar in some aspects, so they could be solved by one network[6-8]. Given the shared layers of two tasks, adversarial samples for object detection may transfer to semantic segmentation[12]. Accordingly, we evaluate this

Table 5–2 mAP of object detection by ATTACTION with transfer techniques on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
ATTACTION	18.1	1.2	19.9	20.5	24.3	22.6	26.4	28.2
DI-ATTACTION	18.1	1.0	19.9	20.5	23.9	22.4	26.3	27.9
TI-ATTACTION	17.0	2.4	20.8	20.8	25.2	23.0	27.9	29.7
SI-ATTACTION	**14.6**	**0.7**	**16.3**	**17.0**	**20.4**	**19.1**	**22.3**	**23.8**

cross-task transferability by adversarial samples generated from SI-ATTACTION on surrogate model M2 (YOLOv3[109]), M3 (RetinaNet[116]) and M5 (Mask R-CNN[7]).

From the results in Table 5–3, we find that SI-ATTACTION, the highest-transferable method, could also greatly hurt the performance of semantic segmentation, leading to a drop on mAP of over 70%. This might inspire the segmentation attackers to indirectly attack object detection.

Table 5–3 Segmentation mAP of SI-ATTACTION on M2 (YOLOv3)

	mAP			mAP50			mAP75		
Surrogate	M5	M7	M8	M5	M7	M8	M5	M7	M8
None	38.0	39.4	40.8	60.6	61.3	63.3	40.9	42.9	44.1
Ablation	31.0	31.9	33.5	51.2	51.0	53.7	32.4	34.3	35.4
M2	17.9	18.6	20.3	31.6	31.7	34.5	18.0	18.9	20.7
M3	11.6	11.9	12.9	19.2	19.1	20.7	12.1	12.6	13.7
M5	1.2	11.1	11.8	2.4	17.9	18.9	1.0	11.9	12.6

5.10 Visual Results

We present the visual results for ATTACTION to further illustrate its influence. For white-box scenarios in Fig. 5–3, the three models correctly detect a “person” and a “skateboard”, and the attention heat map is clear and structured. After ATTACTION, the heat maps are induced to be meaningless and without correct focus, leading to incorrect predictions, i.e., no or false detection.

For black-box scenarios in Fig. 5–4, we visualize several predictions on the same adversarial sample by black-box models. The objects in the image, e.g., laptop and keyboard, are quite large and obvious to detect. However, with a small perturbation from ATTACTION, 5 black-box models all fail to detect the laptop, keyboard and mouse. Surprisingly, 4 of them even detect a non-existent “bed”, which is neither relevant nor similar in the image.

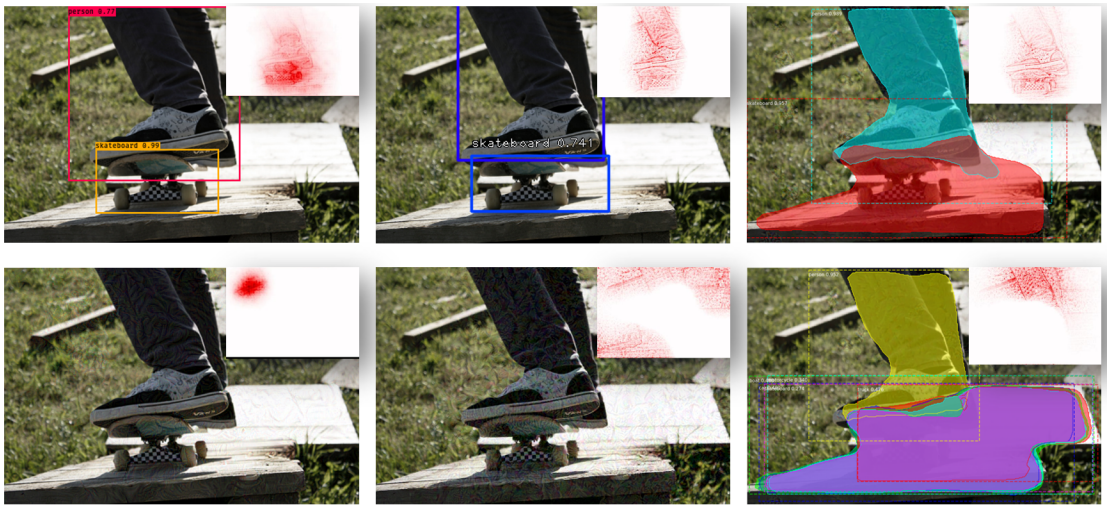

Figure 5–3 White-box attack illustration of ATTACTION. The image contains a person and a skateboard. The top row shows the original correct predictions and the corresponding heat maps for YOLOv3, RetinaNet and Mask R-CNN. The bottom row displays those for adversarial samples generated by attacking corresponding models. Note that there are several incorrect overlapped bounding boxes and masks ("car", "motorcycle" and "boat") in the image at the bottom right.

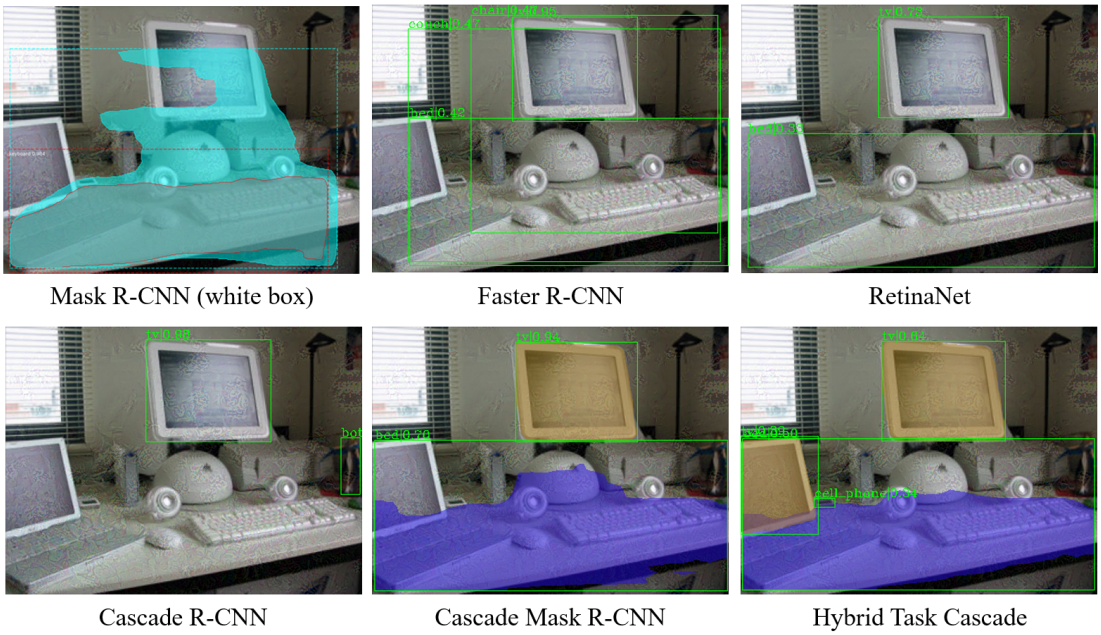

Figure 5–4 Black-box attack illustration of ATTACTION. Here are the predictions of the same adversarial sample generated by attacking Mask R-CNN. The sample contains "TV", "laptop", "mouse" and two "keyboard"s, but several black-box models only detect TV. They also produce incorrect bounding boxes or masks such as "bed", "coach" and "chair".

5.11 Adversarial Objects in Context

Given the great transferability of ATTACTION, we create Adversarial Objects in COntext (AOCO), the first adversarial dataset for object detection. AOCO is generated from the full COCO 2017 validation set[39] with 5k samples. It contains 5K adversarial samples for evaluating object detection (AOCO detection) and 5K for semantic segmentation (AOCO segmentation). Samples in AOCO detection have the long side 416 and that for AOCO segmentation is 448. AOCO could be downloaded from https://pan.baidu.com/s/1fjy9toJDRLTp8RSN-q1gZQ by "gbug".

All 10K samples in AOCO are crafted by SI-ATTACTION, the highest-transferable method on object detection. The surrogate model we attack is YOLOv3 for AOCO detection and Mask R-CNN for AOCO segmentation given the results in Table 5–1 and Table 5–3.

We measure the degree of perturbation Δx in AOCO by Root Mean Squared Error (RMSE) as in[12, 118]. It is calculated as $\sqrt{\sum_i (\Delta x_i)^2 / N}$ in a pixel-wise way, and N is the size of the image. Performance of AOCO is reported in Table 5–4 (**bold** for white-box results). The RMSE in AOCO is lower than that in[84], and the perturbation is quite imperceptible. Adversarial samples in AOCO are demonstrated in the Appendix.

Table 5–4 Detection mAP and segmentation mAP on COCO and AOCO

	RMSE	M1	M2	M3	M4	M5	M6	M7	M8
COCO detection	0.000	29.3	33.4	38.1	40.7	42.1	42.5	45.7	46.9
AOCO detection	6.469	14.6	**0.7**	16.3	17	20.4	19.1	22.3	23.8
COCO segmentation	0.000	\	\	\	\	38.0	\	39.4	40.8
AOCO segmentation	6.606	\	\	\	\	**1.2**	\	11.1	11.8

5.12 Summary

To pursue a high attack transferability, this thesis proposes ATTACTION, which suppresses the multi-node attention, a common property across calculated by our MN-SGLRP, to attack and therefore achieves a state-of-the-art transferability towards black-box models. We also empirically find that transfer techniques may help as well and the adversarial samples in detection transfer towards segmentation. Given the great transferability of ATTACTION, we generate Adversarial Objects in COntext (AOCO), the first adversarial dataset on object detection networks, which could help network designers to quickly evaluate and improve the robustness of detection networks.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

To improve the transferability of adversarial attack, we are the first to attack on attention and achieve a great transferability in black-box scenarios. The success of Attack on Attention (AoA) relies on the semantic features shared by different DNNs. To effectively attack on attention, we apply network visualization method in designing the attention loss. AoA and its variants enjoy a significant increase of the transferability when the traditional cross entropy loss is replaced with attention loss. Furthermore, AoA could be combined with transfer-enhancement methods and achieve a state-of-the-art transferability.

By AoA and its varients, we generate DAmageNet/DAmageNet2, the first dataset containing samples with a small perturbation and a high transfer rate (an error rate over 85% for undefended models). It is a benchmark to evaluate the robustness of DNNs by elaborately-crafted adversarial samples.

We also extend our method towards object detection networks. To clearly calculated the attention heat map for detection nets, we propose a novel network visualization method called MN-SGLRP. Based on it, we design ATTACTION, a high-transferable attack on object detection. Experiments show ATTACTION enjoys a state-of-the-art transferability over all existing methods. Accordingly, we create the dataset AOCO by SI-ATTACTION, which could greatly hurt several black-box models.

6.2 Future Work

However, our work is not perfect. AoA has found the common vulnerability of DNNs in attention. Other versions of attention loss may achieve a better transferability. Attention is only one semantic feature and attacking on other semantic features shared by DNNs is promising to have a good transferability. In ATTACTION, other choices of multiple-node for attention might lead to a better performance, and it is valuable to test its transferability by attacking more surrogate models.

The comprehensive explanation on why adversarial samples exist, how many are they, what is the theoretical bound of their transferability are not fully discussed in this thesis. Deep understandings and real innovations on deep learning theory and model interpretability are essential to answer the questions above, which are the promising works in the future.

Appendix A More Comparative Results of AoA

Table A–1 Error rate (Top-1) of different transfer attacks on DenseNet121

Method	DN121	IncRN2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
CW	100.0%	33.5%	39.5%	31.9%	39.6%	64.6%	53.2%	39.4%
MI-DI-CW	100.0%	37.5%	41.3%	36.1%	41.3%	62.8%	56.3%	40.4%
SI-CW	100.0%	48.6%	56.3%	49.1%	55.1%	78.4%	66.8%	55.9%
PGD	100.0%	34.0%	41.7%	31.9%	41.5%	68.9%	55.5%	41.5%
MI-DI-PGD	100.0%	39.7%	44.6%	39.4%	44.5%	68.3%	60.1%	43.7%
SI-PGD	100.0%	48.2%	57.0%	48.2%	56.0%	80.3%	70.0%	57.0%
AoA	100.0%	46.1%	53.5%	46.1%	55.0%	76.7%	64.6%	52.1%
MI-DI-AoA	100.0%	47.0%	48.8%	46.1%	50.5%	69.7%	63.3%	47.3%
SI-AoA	100.0%	61.8%	69.7%	63.3%	68.6%	85.0%	77.9%	69.6%

Table A–2 Error rate (Top-1) of different transfer attacks on InceptionV3

Method	DN121	IncRN2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
CW	31.0%	22.7%	100.0%	21.3%	26.1%	42.3%	40.7%	33.4%
MI-DI-CW	34.4%	26.5%	100.0%	22.6%	30.2%	46.5%	42.6%	32.5%
SI-CW	41.3%	30.9%	100.0%	34.7%	33.1%	52.0%	44.9%	47.4%
PGD	32.7%	24.2%	100.0%	21.3%	27.3%	45.3%	40.7%	33.7%
MI-DI-PGD	36.8%	29.2%	100.0%	25.2%	33.0%	46.7%	43.4%	34.3%
SI-PGD	41.9%	30.7%	100.0%	34.2%	33.5%	52.4%	46.2%	48.1%
AoA	39.0%	30.2%	100.0%	32.7%	34.0%	52.8%	45.9%	45.1%
MI-DI-AoA	41.1%	34.8%	100.0%	30.9%	38.0%	53.8%	49.4%	39.4%
SI-AoA	49.8%	43.2%	100.0%	51.2%	44.4%	60.2%	52.6%	62.8%

Table A–3 Error rate (Top-1) of different transfer attacks on VGG19

Method	DN121	IncRN2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
CW	85.5%	62.0%	69.8%	62.7%	60.0%	77.8%	100.0%	68.0%
MI-DI-CW	74.3%	54.7%	59.1%	51.4%	52.8%	71.2%	99.9%	59.6%
SI-CW	87.8%	73.0%	74.6%	67.8%	69.5%	84.6%	100.0%	75.7%
PGD	87.1%	64.1%	71.8%	63.9%	63.1%	82.5%	100.0%	71.9%
MI-DI-PGD	79.6%	60.4%	65.2%	56.9%	59.4%	78.2%	99.8%	64.6%
SI-PGD	90.7%	74.8%	78.6%	69.8%	73.7%	88.3%	100.0%	79.6%
AoA	91.4%	73.7%	79.8%	74.2%	73.5%	86.6%	100.0%	81.0%
MI-DI-AoA	84.4%	65.8%	70.9%	63.0%	66.3%	82.7%	99.9%	70.4%
SI-AoA	95.2%	84.4%	87.2%	82.1%	82.3%	91.4%	100.0%	86.9%

Table A–4 Error rate (Top-1) of different transfer attacks on ResNet152

Method	DN121	IncRN2	IncV3	NASNetL	RN152	RN50	VGG19	Xception
CW	42.4%	36.2%	35.3%	25.6%	100.0%	57.7%	46.0%	31.9%
MI-DI-CW	42.6%	39.3%	37.2%	28.8%	100.0%	55.7%	47.3%	33.6%
SI-CW	53.6%	47.4%	41.6%	33.6%	100.0%	64.4%	53.3%	40.3%
PGD	42.7%	35.0%	34.9%	24.5%	98.1%	55.3%	43.6%	30.5%
MI-DI-PGD	44.5%	41.7%	39.3%	30.6%	97.8%	56.8%	49.8%	35.9%
SI-PGD	50.1%	43.4%	42.5%	30.1%	98.2%	62.0%	50.6%	38.4%
AoA	55.9%	54.2%	49.6%	36.4%	100.0%	71.5%	57.2%	45.6%
MI-DI-AoA	48.8%	47.9%	45.4%	36.2%	100.0%	62.2%	54.7%	43.7%
SI-AoA	66.7%	65.2%	58.9%	46.9%	100.0%	76.8%	65.3%	55.9%

Appendix B Explain the Transferability of AoA-N

A possible reason for failure of the defense is that the perturbation generated by AoA-N is not simple noise. One example is shown in Fig. B–1, where the perturbations generated by PGD attack[17] (in l_1, l_2, l_∞ norm) looks like random noise, so they may be eliminated by preprocessing easily. By contrast, perturbations generated by AoA-N has semantic meaning and concentrates on discriminative regions with great magnitude, so many preprocessing-based defense methods look no effect on AoA-N.

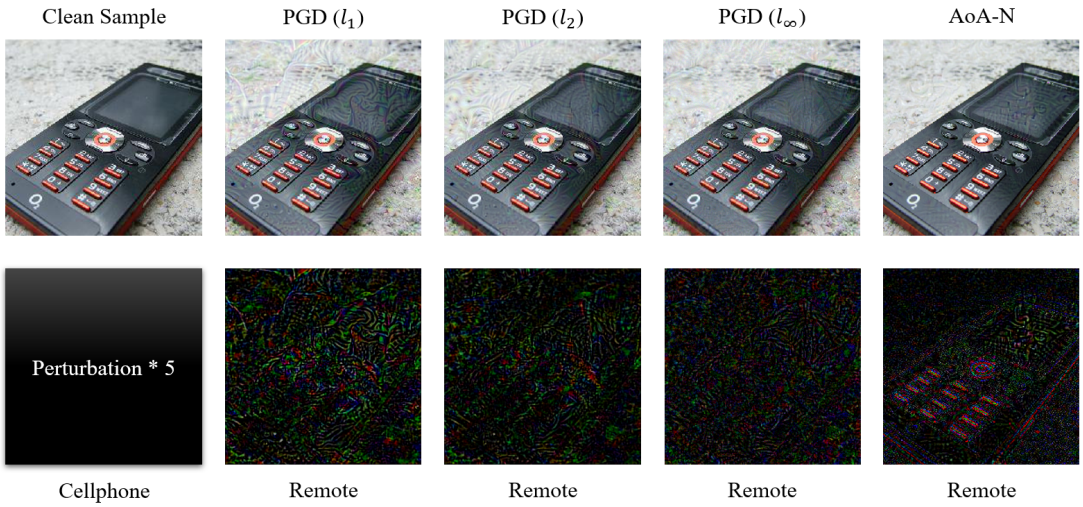

Figure B–1 By different attacks, the original sample (”cellphone”) are perturbed and the adversarial samples are incorrectly recognized as ”remote”. However, the perturbations given by AoA-N has semantic meaning. On the one hand, common filters cannot deal with it. On the other hand, semantic perturbations have better transferability. In fact, the adversarial sample generated by PGD can only cheat the attacked DNN while the rightmost image in the top (this is one sample in DAmageNet2) can cheat all the DNNs listed before.

Appendix C Models for Object Detection

Table C–1 Model backbone and mAPs

ID	Model	Backbone	mAP	mAP50	mAP75
M1	SSD512	VGG16	29.3	49.2	30.8
M2	YOLOv3	Darknet	33.4	56.4	35.8
M3	RetinaNet	ResNet-101	38.1	58.1	40.6
M4	Faster R-CNN	ResNeXt-101-64*4d	40.7	62.0	44.6
M5	Mask R-CNN	ResNeXt-101-64*4d	42.1	63.8	46.3
M6	Cascade RCNN	ResNet-101	42.5	60.7	46.3
M7	Cascade Mask R-CNN	ResNeXt-101-64*4d	45.7	64.1	50.0
M8	Hyrbrid Task Cascade	ResNeXt-101-64*4d	46.9	66.0	51.2

Appendix D Visual Illustration of ATTACTION Process

By ATTACTION, the heat map is attacked to be meaningless and loss its focus. In Fig. D–1, the initial prediction is correct and the heat map is clear. ATTACTION constantly misleads the heat map to be unstructured without outline of objects. Finally, all bounding boxes vanish.

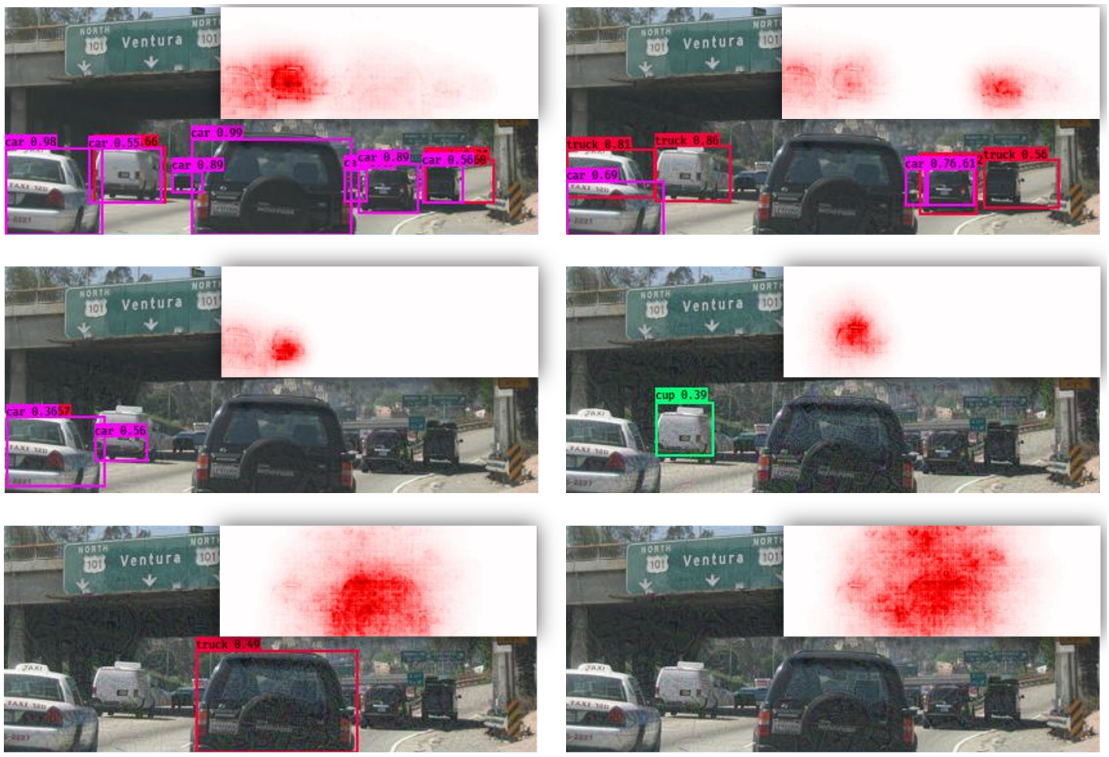

Figure D–1 Transition of prediction and heat map during ATTACTION (from top to bottom and left to right).

Appendix E More Comparative Results of ATTACTION

Table E–1 mAP50 of object detection in different attacks on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
None	49.2	56.4	58.1	62.0	63.8	60.7	64.1	66.0
Dfool	40.2	4.1	46.6	48.3	52.9	48.8	53.1	55.7
Loc	38.8	0.9	42.4	44.8	49.1	45.2	49.6	49.6
DAG	35.8	2.0	36.5	37.9	42.7	38.3	42.5	46.2
ATTACTION	32.6	2.5	33.3	34.7	40.4	35.3	40.0	43.3

Table E–2 mAP75 of object detection in different attacks on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
None	30.8	35.8	40.6	44.6	46.3	46.3	50.0	51.2
Dfool	23.6	2.8	30.7	31.5	35.5	35.4	39.3	40.7
Loc	22.0	0.0	26.6	27.6	31.1	30.8	35.2	35.2
DAG	21.4	0.3	23.9	25.0	28.6	27.3	31.1	32.9
ATTACTION	17.9	1.2	20.2	21.0	25.3	23.6	27.9	29.4

Table E–3 mAP50 of object detection by ATTACTION with transfer techniques on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
ATTACTION	32.6	2.5	33.3	34.7	40.4	35.3	40.0	43.3
TI-ATTACTION	30.4	5.5	34.1	35.1	41.5	35.5	42.1	45.3
DI-ATTACTION	32.5	2.2	33.2	34.5	39.7	34.7	39.8	43.1
SI-ATTACTION	26.7	1.6	27.6	29.1	34.5	29.9	34.3	37.4

Table E–4 mAP75 of object detection by ATTACTION with transfer techniques on M2 (YOLOv3)

Method	M1	M2	M3	M4	M5	M6	M7	M8
ATTACTION	17.9	1.2	20.2	21.0	25.3	23.6	27.9	29.4
TI-ATTACTION	17.1	1.9	21.4	21.3	26.3	24.3	29.8	31.4
DI-ATTACTION	17.9	0.9	20.4	21.1	25.0	23.6	27.6	29.1
SI-ATTACTION	14.2	0.6	16.5	17.1	20.8	19.9	23.3	24.6

Appendix F More about Adversarial Objects in Context

Table F–1 Detection mAP50 and segmentation mAP50 on COCO and AOCO

	M1	M2	M3	M4	M5	M6	M7	M8
COCO detection	49.2	56.4	58.1	62.0	63.8	60.7	64.1	66.0
AOCO detection	26.7	1.6	27.6	29.1	34.5	29.9	34.3	37.4
COCO segmentation	/	/	/	/	60.6	/	61.3	63.3
AOCO segmentation	/	/	/	/	2.4	/	17.9	18.9

Table F–2 Detection mAP75 and segmentation mAP75 on COCO and AOCO

	M1	M2	M3	M4	M5	M6	M7	M8
COCO detection	30.8	35.8	40.6	44.6	46.3	46.3	50.0	51.2
AOCO detection	14.2	0.6	16.5	17.1	20.8	19.9	23.3	24.6
COCO segmentation	/	/	/	/	40.9	/	42.9	44.1
AOCO segmentation	/	/	/	/	1.0	/	11.9	12.6

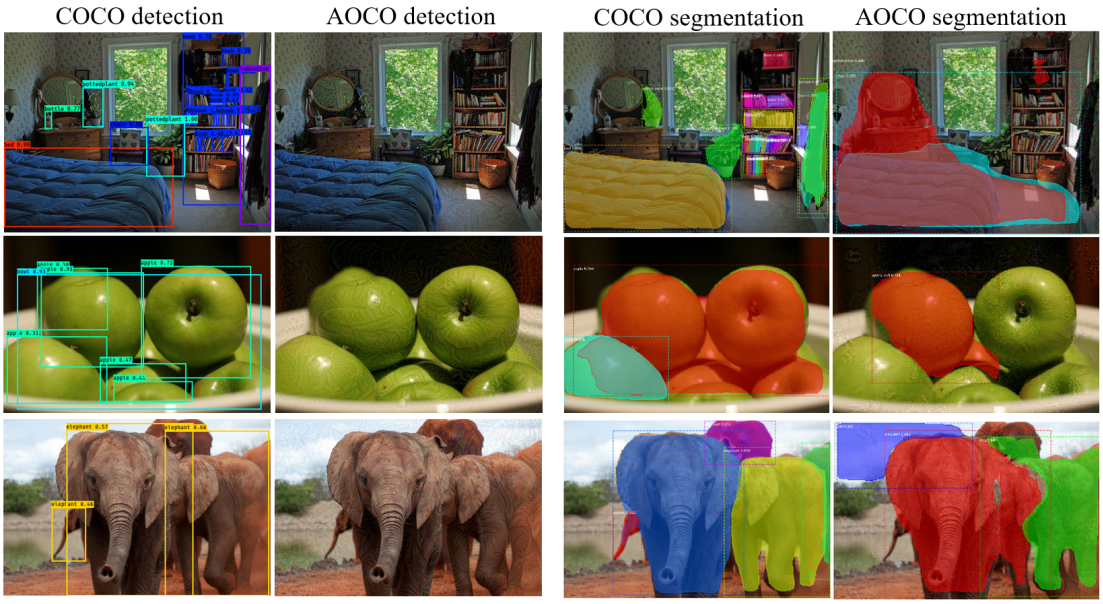

Figure F–1 Detection and segmentation results in COCO and AOCO by YOLOv3 and Mask R-CNN. For COCO, both networks predict correctly. For AOCO segmentation, the top image contains two big masks for “chair” and “potted plan”; the second image contains one false mask for “sports ball”; the bottom image contains “dog” in green, “car” in purple and “elephant” in red.

Bibliography

[1] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[C]//3rd International Conference on Learning Representations. 2015.

[2] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 770-778.

[3] HUANG G, LIU Z, van der MAATEN L, et al. Densely connected convolutional networks[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR. 2017: 2261-2269.

[4] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[C]//Advances in Neural Information Processing Systems. 2015: 91-99.

[5] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.

[6] CAI Z, VASCONCELOS N. Cascade R-CNN: Delving into high quality object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 6154-6162.

[7] HE K, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 2961-2969.

[8] CHEN K, PANG J, WANG J, et al. Hybrid task cascade for instance segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 4974-4983.

[9] SCHROFF F, KALENICHENKO D, PHILBIN J. Facenet: A unified embedding for face recognition and clustering[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 815-823.

[10] SZEGEDY C, ZAREMBA W, SUTSKEVER I, et al. Intriguing properties of neural networks[C]//2nd International Conference on Learning Representations, ICLR. 2014.

[11] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples[J]. STAT, 2015, 1050: 20.

[12] XIE C, WANG J, ZHANG Z, et al. Adversarial examples for semantic segmentation and object detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 1369-1378.

[13] ZHANG H, WANG J. Towards adversarially robust object detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 421-430.

[14] JIA Y, LU Y, SHEN J, et al. Fooling detection alone is not enough: Adversarial attack against multiple object tracking[C]//International Conference on Learning Representations. 2019.

[15] SAMEK W. Explainable AI: Interpreting, explaining and visualizing deep learning[M]. Springer Nature, 2019.

[16] CARLINI N, WAGNER D. Towards evaluating the robustness of neural networks[C]//2017 IEEE Symposium on Security and Privacy (SP). 2017: 39-57.

[17] MADRY A, MAKELOV A, SCHMIDT L, et al. Towards deep learning models resistant to adversarial attacks[C]//6th International Conference on Learning Representations, ICLR. 2018.

[18] CHENG S, DONG Y, PANG T, et al. Improving black-box adversarial attacks with a transfer-based prior[J]., 2019.

[19] ILYAS A, ENGSTROM L, MADRY A. Prior convictions: Black-box adversarial attacks with bandits and priors[C]//7th International Conference on Learning Representations, ICLR. 2019.

[20] GUO Y, YAN Z, ZHANG C. Subspace attack: Exploiting promising subspaces for query-efficient black-box attacks[C]//Advances in Neural Information Processing Systems. 2019: 3820-3829.

[21] PAPERNOT N, MCDANIEL P, GOODFELLOW I, et al. Practical black-box attacks against machine learning[C]//Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security. 2017: 506-519.

[22] MOOSAVI-DEZFOOLI S M, FAWZI A, FAWZI O, et al. Universal adversarial perturbations[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1765-1773.

[23] DONG Y, PANG T, SU H, et al. Evading defenses to transferable adversarial examples by translation-invariant attacks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 4312-4321.

[24] DONG Y, LIAO F, PANG T, et al. Boosting adversarial attacks with momentum[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 9185-9193.

[25] XIE C, ZHANG Z, ZHOU Y, et al. Improving transferability of adversarial examples with input diversity[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 2730-2739.

[26] LIN J, SONG C, HE K, et al. Nesterov accelerated gradient and scale invariance for improving transferability of adversarial examples[C]//8th International Conference on Learning Representations, ICLR. 2020.

[27] SU D, ZHANG H, CHEN H, et al. Is robustness the cost of accuracy?–A comprehensive study on the robustness of 18 deep image classification models[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[28] ZHANG T, ZHU Z. Interpreting adversarially trained convolutional neural networks[C]//Proceedings of the 36th International Conference on Machine Learning, ICML. 2019: 7502-7511.

[29] DENG J, DONG W, SOCHER R, et al. Imagenet: A large-scale hierarchical image database[C]//2009 IEEE Conference on Computer Vision and Pattern Recognition. 2009: 248-255.

[30] GANIN Y, USTINOVA E, AJAKAN H, et al. Domain-adversarial training of neural networks[J]. Journal of Machine Learning Research, 2016.

[31] SHRIVASTAVA A, PFISTER T, TUZEL O, et al. Learning from simulated and unsupervised images through adversarial training[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 2107-2116.

[32] SINHA A, NAMKOONG H, DUCHI J. Certifiable distributional robustness with principled adversarial training[J]. Proceedings of the International Conference on Learning Representations, 2018: 29.

[33] THYS S, VAN RANST W, GOEDEMÉ T. Fooling automated surveillance cameras: adversarial patches to attack person detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2019.

[34] LI Y, TIAN D, BIAN X, et al. Robust adversarial perturbation on deep proposal-based models[J]. ArXiv preprint arXiv:1809.05962, 2018.

[35] LI Y, BIAN X, LYU S. Attacking object detectors via imperceptible patches on background[J]. CoRR, abs/1809.05966, 2018.

[36] BACH S, BINDER A, MONTAVON G, et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation[J]. PloS one, 2015, 10(7): e0130140.

[37] GU J, YANG Y, TRESP V. Understanding individual decisions of CNNs via contrastive backpropagation[C]//Asian Conference on Computer Vision. 2018: 119-134.

[38] IWANA B K, KUROKI R, UCHIDA S. Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation[J]. ArXiv preprint arXiv:1908.04351, 2019.

[39] LIN T Y, MAIRE M, BELONGIE S, et al. Microsoft coco: Common objects in context[C]//European Conference on Computer Vision. 2014: 740-755.

[40] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.

[41] BREIMAN L. Random forests[J]. Machine Learning, 2001, 45(1): 5-32.

[42] FREUND Y, SCHAPIRE R E. A desicion-theoretic generalization of on-line learning and an application to boosting[C]//European conference on computational learning theory. 1995: 23-37.

[43] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[44] KRIZHEVSKY A, HINTON G, et al. Learning multiple layers of features from tiny images[J]., 2009.

[45] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[C]//Advances in Neural Information Processing Systems. 2012: 1097-1105.

[46] RUSSAKOVSKY O, DENG J, SU H, et al. Imagenet large scale visual recognition challenge[J]. International Journal of Computer Vision, 2015.

[47] NAIR V, HINTON G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of the 27th International Conference on Machine Learning (ICML-10). 2010: 807-814.

[48] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.

[49] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 1-9.

[50] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the inception architecture for computer vision[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2818-2826.

[51] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. 2015.

[52] XIE S, GIRSHICK R, DOLLÁR P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1492-1500.

[53] RADOSAVOVIC I, KOSARAJU R P, GIRSHICK R, et al. Designing network design spaces[J]. ArXiv preprint arXiv:2003.13678, 2020.

[54] MOOSAVI-DEZFOOLI S M, FAWZI A, FROSSARD P. Deepfool: a simple and accurate method to fool deep neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2574-2582.

[55] SU J, VARGAS D V, SAKURAI K. One pixel attack for fooling deep neural networks[J]. IEEE Transactions on Evolutionary Computation, 2019.

[56] SONG Y, SHU R, KUSHMAN N, et al. Constructing unrestricted adversarial examples with generative models[C]//Advances in Neural Information Processing Systems. 2018: 8312-8323.

[57] TANG S, HUANG X, CHEN M, et al. Adversarial attack type I: Cheat classifiers by significant changes[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019.

[58] BALUJA S, FISCHER I. Adversarial transformation networks: Learning to generate adversarial examples[J]. ArXiv preprint arXiv:1703.09387, 2017.

[59] HAN J, DONG X, ZHANG R, et al. Once a MAN: Towards multi-target attack via learning multi-target adversarial network once[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 5158-5167.

[60] KINGMA D P, WELLING M. Auto-encoding variational bayes[J]., 2013.

[61] CRESWELL A, BHARATH A A, SENGUPTA B. Latentpoison-adversarial attacks on the latent space[J]. ArXiv preprint arXiv:1711.02879, 2017.

[62] ODENA A, OLAH C, SHLENS J. Conditional image synthesis with auxiliary classifier GANSs[C]//Proceedings of the 34th International Conference on Machine Learning-Volume 70. 2017: 2642-2651.

[63] COHEN J M, ROSENFELD E, KOLTER J Z. Certified adversarial robustness via randomized smoothing[C]//Proceedings of the 36th International Conference on Machine Learning, ICML 2019. 2019.

[64] SALMAN H, LI J, RAZENSHTEYN I P, et al. Provably robust deep learning via adversarially trained smoothed classifiers[C]//Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019. 2019.

[65] ILYAS A, SANTURKAR S, TSIPRAS D, et al. Adversarial examples are not bugs, they are features[C]//Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019. 2019: 125-136.

[66] MIYATO T, DAI A M, GOODFELLOW I J. Adversarial training methods for semi-supervised text classification[C]//5th International Conference on Learning Representations, ICLR. 2017.

[67] SANKARANARAYANAN S, JAIN A, CHELLAPPA R, et al. Regularizing deep networks using efficient layerwise adversarial training[C]//Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

[68] ZHANG D, ZHANG T, LU Y, et al. You only propagate once: Painless adversarial training using maximal principle[C]//Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019. 2019.

[69] LIAO F, LIANG M, DONG Y, et al. Defense against adversarial attacks using high-level representation guided denoiser[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[70] XIE C, WU Y, MAATEN L V D, et al. Feature denoising for improving adversarial robustness[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 501-509.

[71] LIU Z, LIU Q, LIU T, et al. Feature distillation: DNN-oriented JPEG compression against adversarial examples[C]//IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019. 2019: 860-868.

[72] PRAKASH A, MORAN N, GARBER S, et al. Deflecting adversarial attacks with pixel deflection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8571-8580.

[73] MUSTAFA A, KHAN S H, HAYAT M, et al. Image super-resolution as a defense against adversarial attacks[J]. IEEE Transactions on Image Processing, 2020, 29: 1711-1724. DOI: 10.1109/TIP.2019.2940533.

[74] CHEN S T, CORNELIUS C, MARTIN J, et al. Robust physical adversarial attack on faster R-CNN object detector[J]. ArXiv preprint arXiv:1804.05810, 2018.

[75] LU J, SIBAI H, FABRY E. Adversarial examples that fool detectors[J]. ArXiv preprint arXiv:1712.02494, 2017.

[76] LI Y, BIAN X, CHANG M C, et al. Exploring the vulnerability of single shot module in object detectors via imperceptible background patches[C]//30th British Machine Vision Conference 2019, BMVC 2019. 2018.

[77] WANG Y, TAN Y A, ZHANG W, et al. An adversarial attack on DNN-based black-box object detectors[J]. Journal of Network and Computer Applications, 2020: 102634.

[78] PAPERNOT N, MCDANIEL P, GOODFELLOW I. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples[J]., 2016.

[79] BRENDEL W, RAUBER J, BETHGE M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models[J]., 2018.

[80] ILYAS A, ENGSTROM L, ATHALYE A, et al. Black-box adversarial attacks with limited queries and information[C]//Proceedings of the 35th International Conference on Machine Learning, ICML 2018. 2018.

[81] RU B, COBB A, BLAAS A, et al. BayesOpt adversarial attack[C]// International Conference on Learning Representations. 2020.

[82] MEUNIER L, ATIF J, TEYTAUD O. Yet another but more efficient black-box adversarial attack: Tiling and evolution strategies[J]. ArXiv preprint arXiv:1910.02244, 2019.

[83] DU J, ZHANG H, ZHOU J T, et al. Query-efficient meta attack to deep neural networks[C]//8th International Conference on Learning Representations, ICLR. 2019.

[84] WU D, WANG Y, XIA S, et al. Skip connections matter: On the transferability of adversarial examples generated with ResNets[C]//Proceedings of the International Conference on Learning Representations. 2019.

[85] HENDRYCKS D, ZHAO K, BASART S, et al. Natural adversarial examples[J]. ArXiv preprint arXiv:1907.07174, 2019.

[86] HENDRYCKS D, DIETTERICH T. Benchmarking neural network robustness to common corruptions and perturbations[J]. Proceedings of the International Conference on Learning Representations, 2019.

[87] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems. 2017: 5998-6008.

[88] LING W, TSVETKOV Y, AMIR S, et al. Not all contexts are created equal: Better word representations with variable attention[C]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015.

[89] ZHOU B, KHOSLA A, LAPEDRIZA A, et al. Learning deep features for discriminative localization[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2921-2929.

[90] LIN M, CHEN Q, YAN S. Network In network[C]//2nd International Conference on Learning Representations, ICLR. 2014.

[91] SELVARAJU R R, COGSWELL M, DAS A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 618-626.

[92] ZHOU B, KHOSLA A, LAPEDRIZA À, et al. Object detectors emerge in deep scene CNNs[C]//3rd International Conference on Learning Representations, ICLR. 2015.

[93] ZEILER M D, FERGUS R. Visualizing and understanding convolutional networks[C]//European Conference on Computer Vision. 2014: 818-833.

[94] ZHOU J, TROYANSKAYA O G. Predicting effects of noncoding variants with deep learning–based sequence model[J]. Nature Methods, 2015, 12(10): 931.

[95] SIMONYAN K, VEDALDI A, ZISSERMAN A. Deep inside convolutional networks: Visualising image classification Models and Saliency Maps[C]//2nd International Conference on Learning Representations, ICLR. 2014.

[96] SPRINGENBERG J T, DOSOVITSKIY A, BROX T, et al. Striving for simplicity: The all convolutional net[C]//3rd International Conference on Learning Representations, ICLR. 2015.

[97] SZEGEDY C, IOFFE S, VANHOUCKE V, et al. Inception-v4, Inception-ResNet and the impact of residual connections on Learning[C]//Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. 2017: 4278-4284.

[98] ZOPH B, VASUDEVAN V, SHLENS J, et al. Learning transferable architectures for scalable image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 8697-8710.

[99] CHOLLET F, et al. Keras[Z]. https://keras.io. 2015.

[100] Martín Abadi, Ashish Agarwal, Paul Barham, et al. TensorFlow: Large-scale mchine learning on heterogeneous systems[Z]. Software available from tensorflow.org. 2015.

[101] CARLINI N, WAGNER D. Adversarial examples are not easily detected: Bypassing ten detection methods[C]//Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. 2017: 3-14.

[102] GUO C, RANA M, CISSÉ M, et al. Countering adversarial images using input transformations[C]//6th International Conference on Learning Representations, ICLR. 2018.

[103] COHEN J M, ROSENFELD E, KOLTER J Z. Certified adversarial robustness via randomized smoothing[C]//Proceedings of the 36th International Conference on Machine Learning, ICML 2019. 2019.

[104] KURAKIN A, GOODFELLOW I J, BENGIO S. Adversarial examples in the physical world[C]//5th International Conference on Learning Representations, ICLR. 2017.

[105] CHOLLET F. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1251-1258.

[106] TRAMÈR F, KURAKIN A, PAPERNOT N, et al. Ensemble adversarial training: Attacks and defenses[C]//6th International Conference on Learning Representations, ICLR. 2018.

[107] HUANG G, LIU S, VAN DER MAATEN L, et al. Condensenet: An efficient densenet using learned group convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 2752-2761.

[108] XIE C, WANG J, ZHANG Z, et al. Mitigating adversarial effects through randomization[C]//6th International Conference on Learning Representations, ICLR. 2018.

[109] REDMON J, FARHADI A. Yolov3: An incremental improvement[J]. ArXiv preprint arXiv:1804.02767, 2018.

[110] PASZKE A, GROSS S, MASSA F, et al. PyTorch: An imperative style, high-performance deep learning library[C]//Advances in Neural Information Processing Systems. 2019: 8024-8035.

[111] ALBER M, LAPUSCHKIN S, SEEGERER P, et al. INNvestigate neural networks[J]. Journal of Machine Learning Research, 2019, 20(93): 1-8.

[112] CHEN K, WANG J, PANG J, et al. MMDetection: Open mmlab detection toolbox and benchmark[J]. ArXiv preprint arXiv:1906.07155, 2019.

[113] LIU W, ANGUELOV D, ERHAN D, et al. Ssd: Single shot multibox detector[C]//European Conference on Computer Vision. 2016: 21-37.

[114] Qqwweee. YOLOv3 on Keras and TensorFlow[Z]. https://github.com/qqwweee/keras-yolo3. 2018.

[115] SUNG-YI L, PRIYA G, ROSS G, et al. Keras implementation of RetinaNet object detection[Z]. https://github.com/fizyr/keras-retinanet. 2017.

[116] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 2980-2988.

[117] ABDULLA W. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow[Z]. https://github.com/matterport/Mask_RCNN. 2017.

[118] LIU Y, CHEN X, LIU C, et al. Delving into transferable adversarial examples and black-box attacks[J]., 2017.

Acknowledgements

Time slips through my fingers. It seems to be yesterday that I walked in SJTU campus with my eyes shining for growth, and tomorrow is the time to set out again. Nevertheless, the knowledge, ability, insight and philosophy I learn from SJTU benefit my whole life. Words fail me when I try to describe my gratitude towards everyone I met and everything I went through.

I extend my sincere appreciation to my academic advisor, Prof. Xiaolin Huang, who constantly provides constructive guidance. He instructs comprehensively from a high-level to the practical skills, and I am grateful for the full harvest every time after our communication. He always encourages me to find my own interest and aim high. These words are kept deep in my heart, driving me forward as a continuous power.

I would like to thank Prof. Kun Zhang in Carnegie Mellon University, who offers invaluable suggestions on how to present the work. His comments exert a huge impact on polishing this thesis. I also thank all reviewers for their helpful comments, which plays a crucial role in inspiring me to improve my method design and experiments. I present my thanks to Prof. Xiang Yin, Prof. Cailian Chen, Prof. Weidong Zhang, Prof. Yuanlong Li and Dr. Haifeng Zhang for suggestions on this thesis.

I must express my great thanks to Fan He, Chengjin Sun and other senior fellow apprentices. They provide much help and suggestions in this work, which contributes to significant improvements of this thesis. I would also thank Zhengbao He for many helpful discussions.

I would thank Yue Gao, Enmei Tu and all my class instructors for imparting valuable knowledge. I thank all my teammates, especially my dear friends Peidong Zhang and Tianhao Mou, who offer me much help and our discussions are always efficient and interesting. I must say thanks to my family for their constant support of my life.

Lastly, this work was partially supported by National Key Research Development Project (No. 2018AAA0100702) and National Natural Science Foundation of China (No. 61977046).

Publications

[1] Chen S, He Z, Sun C, Huang X. Universal Adversarial Attack on Attention and the Resulting Dataset DAmageNet[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, under second-round review.

[2] Sun C, Chen S, Huang X. Double Backpropagation for Training Autoencoders against Adversarial Attack[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, under first-round review.

[3] Chen S, He F, Zhang K, Huang X. Attack on Multi-Node Attention for Object Detection[C]. 34th Conference on Neural Information Processing Systems (NeurIPS), 2020, under review.

[4] Sun C, Chen S, Cai J, Huang X. Type I Attack for Generative Models[C]. International Conference on Image Processing (ICIP), 2020.

[5] Chen S, Gao Y, Zhang P. Aspect-based Review Summary Generation with Diversification[C]. 2019 3rd International Symposium on Autonomous Systems (ISAS). IEEE, 2019: 311-316.