EURASIP Journal on Information Security

RESEARCH Open Access

Multitask adversarial attack with dispersion amplification

Pavlo Haleta[1], Dmytro Likhomanov[1*] and Oleksandra Sokol[1]

Abstract

Recently, adversarial attacks have drawn the community's attention as an effective tool to degrade the accuracy of neural networks. However, their actual usage in the world is limited. The main reason is that real-world machine learning systems, such as content filters or face detectors, often consist of *multiple* neural networks, each performing an individual task. To attack such a system, adversarial example has to pass through many distinct networks at once, which is the major challenge addressed by this paper. In this paper, we investigate *multitask* adversarial attacks as a threat for real-world machine learning solutions. We provide a novel black-box adversarial attack, which significantly outperforms the current state-of-the-art methods, such as Fast Gradient Sign Attack (FGSM) and Basic Iterative Method (BIM, also known as Iterative-FGSM) in the multitask setting.

Keywords: Adversarial attack, Neural network, Multitask adversarial attack, Attack transferability

1 Introduction

Deep neural networks (DNN) have reached outstanding accuracy in many tasks related to computer vision. This led to their adaptation to safety-critical systems, such as autonomous driving and medical scanning. Despite of their effectiveness, many security flaws [1] in machine learning systems were found recently, including vulnerability to adversarial attacks [2]. Slight perturbation applied to model input can lead to completely unexpected model prediction or classification output. Two different threat models are considered in adversarial research: white-box and black-box. In the white-box scenario, the attacker has full access to the model architecture and parameters. In the black-box case, malicious actor's knowledge is limited to either model output probabilities or to the final prediction. While white-box setting is of special interest for the researchers (due to fully controlled environment), from cyber security perspective, black-box attacks are more meaningful. In real-world applications, the model is usually available through API. Thus, the attacker cannot access the inner state of the model directly. In this scenario, the attacker utilizes a substitute network to reproduce the behavior of AI model behind the API.

Adversarial attacks is now a hot academic topic. To understand the risks and even opportunities of adversarial attacks for the business, let us start with an example from Avito company described in their blog [3]. One of Avito businesses is advertising a platform for the customers that wish to sell their cars. To add extra layer of privacy for the users, Avito replaced number plates on the cars with branded watermark. An issue popped up when rival market player started to scrap Avito ads, detect and replace Avito watermark with their own, then place those same adds on a different platform without notifying neither Avito nor car owners. These actions negatively affected Avito's business: jeopardized relations with the customers, in particular, violated regulations regarding their personal data usage. To address the issue, Avito engineered an adversarial solution applied to the whole image. This attack targeted competitor's object detection engine, more specifically, their number plate detection machinery. As a result, the competitor became unable to automatically detect and replace the number plates. Also, it became easy for Avito to detect stolen ads and to mitigate misuse of personal data.

*Correspondence: dm.likhomanov@gmail.com
[1] Samsung R&D Institute Ukraine, Kyiv, Ukraine

This case shows how the business can protect their own intellectual property and interests of their customers with adversarial attacks. On the other hand, adversarial attacks can be deployed by threat actors for nefarious purposes as well. One example is bypassing content filtering on the website.

In Avito's example, it was sufficient to thwart just object detection. But in case of content filtering, evasion of object detection alone would not be enough. Here, *multitask* adversarial attacks come to play where several computer vision engines shall be targeted at the same time. Object detection, image classification, and text recognition might be applied simultaneously in a web-filtering solution to detect advertisement of parental advisory content.

2 Contribution summary

Adversarial attacks that target several *models* of the same task have been researched quite well. At the same time, *multitask* adversarial attacks have surprisingly low coverage in the literature, despite their importance for security of AI-based business solutions (see Lu at al. [4]).

Our work is inspired by Lu at al. [4] that describes dispersion reduction technique to enhance cross-task transferability of adversarial attacks. Our research expands this method further.

In this paper, we propose a new method for multitask adversarial attacks based on dispersion amplification of the inner DNN activations. We have shown experimentally that while dispersion reduction seems to be an intuitive choice as it reduces contrast on the inner layers, amplification of the dispersion produces the same effect on the DNN accuracy. For some tasks, our method outperforms dispersion reduction in terms of both adversarial attack effectiveness and perturbation size.

3 Related works

Most of real-world machine learning systems do not expose their internal state. Thus, only black-box attacks are possible. Early research of black-box attacks was focused on query-feedback mechanism. Here, the attacker uses some local search methods such as gradient descent over target model output probabilities to find an optimal adversarial perturbation [5–7]. Inability to evaluate gradient of the model makes those approaches rely on a big amount of queries to the target model. While it is not a problem in case of stand-alone models, in real-world online systems, big amount of queries to the model or ML API will be detected and treated by service provider as a malicious activity. To overcome this obstacle, adversarial attacks based on substitute model were proposed [8, 9]. Here, the attacker trains another model that mimics the target model's behavior. Then, the attacker crafts adversarial samples using the generated model as white-box target. The effectiveness of this attack relies on the transferability property of the crafted adversarial examples—the ability to simulate target model's predictions. This approach works because commercial solutions tend to re-use many OpenSource components or train their models on publicly available datasets. Thus, adversarial attacks crafted on a substitution model tend to work successfully against commercial proprietary solutions as well.

Methods to enhance adversarial transferability have been proposed recently [10, 11]. However, all of them rely on task-specific loss function. Also, quite frequently, they can be applied to image classification only. This significantly limits their usage in real-world scenarios, due to ensembling methods applied to protect against evasion attacks. In security-sensitive machine learning systems, several neural networks are often combined. For example, face recognition system consists of face detector (detects whether face is present or not) and liveness detector (classifies detected face as real-one or an image). To fool such a system, adversarial example has to thwart both models simultaneously. Significantly less research was conducted in this area [4, 12].

Guo et al. [12] proposed a multitask adversarial attack (MTA) method based on the idea of universal perturbation generated for multiple tasks at once. Universal perturbation (first described in Moosavi-Dezfooli et al. [13]) is a perturbation that can be applied to any image to successfully attack specific model or models. To generate such a perturbation, one needs to target a set of images on the step of attack with minimal perturbation size. This approach provides an opportunity for instant attack as perturbation needs to be generated only once, while keeping the perturbation size slightly bigger than the one in case of targeted attacks.

Paper [12] utilizes both the concept of universal and per-instance perturbations. The authors utilize generative adversarial networks (GANs) to generate perturbations that successfully attack the set of predefined tasks. For each specific task, a separate generator is trained. This makes targeting additional tasks straightforward but at the same time limits this approach by requiring training of additional generators for every new task. To address this obstacle, we investigated the cross-task transferable adversarial attacks based on the ideas described by Lu at al. [4]. In this paper, a method is described for perturbation generation that are able to penetrate multiple models, regardless of their architecture and task without prior knowledge about the specific task.

Most computer vision neural networks share common architecture baseline, regardless of their tasks. They are built of multiple convolutional filters laid on the top of one another. Each layer detects the presence of some patterns in input image and then passes this information to higher layers. The first layers memorize most basic, general patterns (such as the presence of color change on image or the presence of horizontal line), while the latter layers can remember more complex task-related features (such as the presence of an eye on the image, in case of face recognition). This generic part of convolutional neural network is commonly referred to as backbone, while the latter layers are referred to as heads.

Based on these fundamental principles of convolutional neural networks, Lu at al. [4] made the following assumption. Due to the limitation of feature space in which CNN operates, different neural networks, regardless of their architecture or the data they were train on, memorize similar low-level image patterns in their early layers. Therefore, image perturbation that disrupts model activations on early layers has a high chance to be successfully transferred to the next model.

Before investigating deeper into the ideas of Lu at al. [4], it is critical to overview the scenarios where ideas mentioned before might not work. Barni at al. [14] in the work *On the Transferability of Adversarial Examples against CNN-based Image Forensics* have shown such scenarios. Concept of pre-trained backbone as a basis for transfer learning is popular nowadays. It works properly because ImageNet is a very large-scale dataset, and models trained on it tend to learn general features that are good for most of the tasks. But Barni et al. [14] access different but very important set of tasks. In cases of training from scratch with not very deep neural network, one would not get neural network with general purpose feature extractions. Features produced by such, not very deep network will be highly task specific and even more data specific. And it is exactly what Barni at al. have shown in their work. The authors accessed three transferability setups: cross-training (the same model but different datasets), cross-model (the same dataset but different models), cross-model-and-training (different models, different datasets). All of the scenarios were with the same target task of digital forensics. In each case, transferability was low with cross-model-and-training scenario showing the lowest attack transferability. In the further work [15], Barni at al. proposed an effective method for the detection of adversarial attacks and effective protection of image forensics approaches from those attacks. The authors once again have shown that transferability in the domain-oriented tasks are highly limited. Nevertheless, use cases of content filtering that we want to examine in this work are different. First of all, we are interested in cross-task transferability in the cases where the backbone was pre-trained to be general purpose feature extractor. This use case is more common as it requires less data collection and labeling and is common in many tasks ranging from classification to object detection and re-ID.

Scenario, where the attacker uses inner layer activations to generate multi-task transferable adversarial example, is shown in Fig. 1. Lu at al. [4] investigated a method called dispersion reduction. They argued that reduction of dispersion on the specific early layers of the CNN activations results in the highest transferability of the attack. Dispersion reduction can be viewed as the reduction of the contrast on the activation picture. It makes it natural that further layers of the network cannot work properly with those feature maps. However, we investigated several inner layer activation distortion techniques and experimentally showed that other manipulations with inner layer activations are also effective. In our paper, we focus

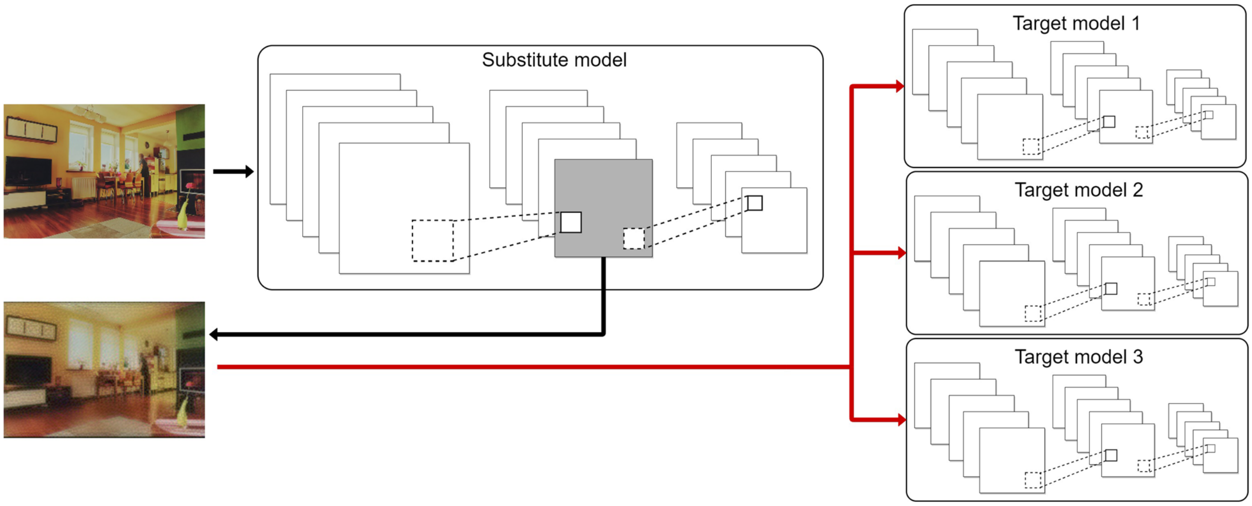


Fig. 1 Attacker uses a substitute model to craft adversarial image that successfully passes through different models and across different CV tasks (such as classification, object detection, recognition)

on *dispersion amplification* as the one with the highest transferability rate.

4 Methodology

Let $f_t(x)$ be the target neural network which takes $x \in \mathbb{R}^n$ as an input and produces some prediction $y \in \mathbb{R}^m$. The goal of adversarial attack, with respect to selected p-norm and perturbation budget ε, is to generate an adversarial input $\tilde{x}$ such that:

$$\tilde{x} = \underset{\|x-\tilde{x}\|^p \leqslant \varepsilon}{\operatorname{argmax}} \quad l(\tilde{x}, t) \tag{1}$$

where l and t denote task-specific loss function and ground truth output, respectively. FGSM [2] and BIM [16] methods are based on gradient ascent. They can be used to find optimum for the function (1). To perform these attacks in the black-box scenario (where gradient value remains unknown), a substitution neural network f_s shall be trained on $(x, f_t(x))$ and then used to emulate the target network. The idea is that substitute model trained on the data labeled by initial network will result in similar feature extractors in the backbone. However, this approach requires knowledge about the target model outputs. It also depends on the loss function. This information is often limited; thus, adversarial examples obtained in such a way have low transferability rate.

To overcome these limitations, we propose to calculate the loss function based on model activations from one of the early layers of substitute model, rather than from the model's output. In this case, knowledge about the original loss function is used by the target model, and the target model predictions remain irrelevant. Disrupting the model in the early layer will cause all further layers to raise invalid activations, and as a result, the model will make wrong predictions.

Based on the assumption made previously, perturbations that affect substitute model on early layers have high chance to succeed in a similar way with the target model. We have evaluated many techniques that can be used to alter early layer activations, including the following:

- Dispersion reduction [4]
- Dispersion amplification
- Reduction of distance between two images in feature space

Lu at al. [4] noted that reducing standard deviation of activation layers will have the effect similar to reducing image contrast, therefore making feature map useless for further layers. While this is true, we have found that other distortion techniques also effectively improve adversarial transferability. Our experiments have shown that dispersion amplification technique results in higher rate of transfer than dispersion reduction. For dispersion amplification, we propose to increase the standard deviation (std) of internal layer by factor of η. Setting higher η results in higher dispersion on the resulting image requirement. After such an attack, the image becomes out of distribution for all further layers. As a result, they cannot infer meaningful features from the attacked image. The loss function for each mentioned method is defined as follows:

$$\begin{gathered}\text{Dispersion amplification}\\ l(x) = \eta \cdot \operatorname{std}(f_i(x_0)) - \operatorname{std}(f_i(x))\end{gathered} \tag{2}$$

$$\text{Dispersion reduction:} l(x) = \operatorname{std}(f_i(x)) \tag{3}$$

where f_i and x denote activations on layer i and generated adversarial image, respectively. x_0 is an image before any manipulations.

To minimize these loss functions, we used gradient-based iterative process with respect to p-norm constraints. The details of proposed method is shown in Algorithm 1.

```
Algorithm 1: Dispersion amplification
Data: surrogate model $f_s$, intermediate layer $i$,
      iterations number $T$, original sample $x$,
      perturbation budget $\varepsilon$
Result: Adversarial sample $x_t$
$x_t \leftarrow x$;
$F_0 = f_s(x_0)$;
compute standard deviation of $F_0 : S_0 = \mathrm{std}(F_0)$;
for $t \leftarrow 0$ to $T$ do
    $F_t = f_s(x_t)|_t$ ;
    compute standard deviation of $S_t = F_t : \mathrm{std}(F_t)$;
    compute loss function: $L_t = S_0 * \eta - S_t$;
    compute gradient w.r.t. $x$: $\nabla_x \mathrm{std}(L_t)$;
    update $x_t : x_t = x_t + \nabla_x \mathrm{std}(L_t)$;
    apply constrains to $x_t : x_t = \mathrm{clip}(x_t, \varepsilon)$;
end
```

5 Experimental results

In this section, we describe the experimental results of the proposed attack method. We evaluate the proposed attack against typical computer vision tasks, including object detection, face detection, image classification, and semantic segmentation. ImageNet [17] and COCO 2012 [18] datasets were used for evaluation, with accuracy and mAP metrics used for each dataset, respectively.

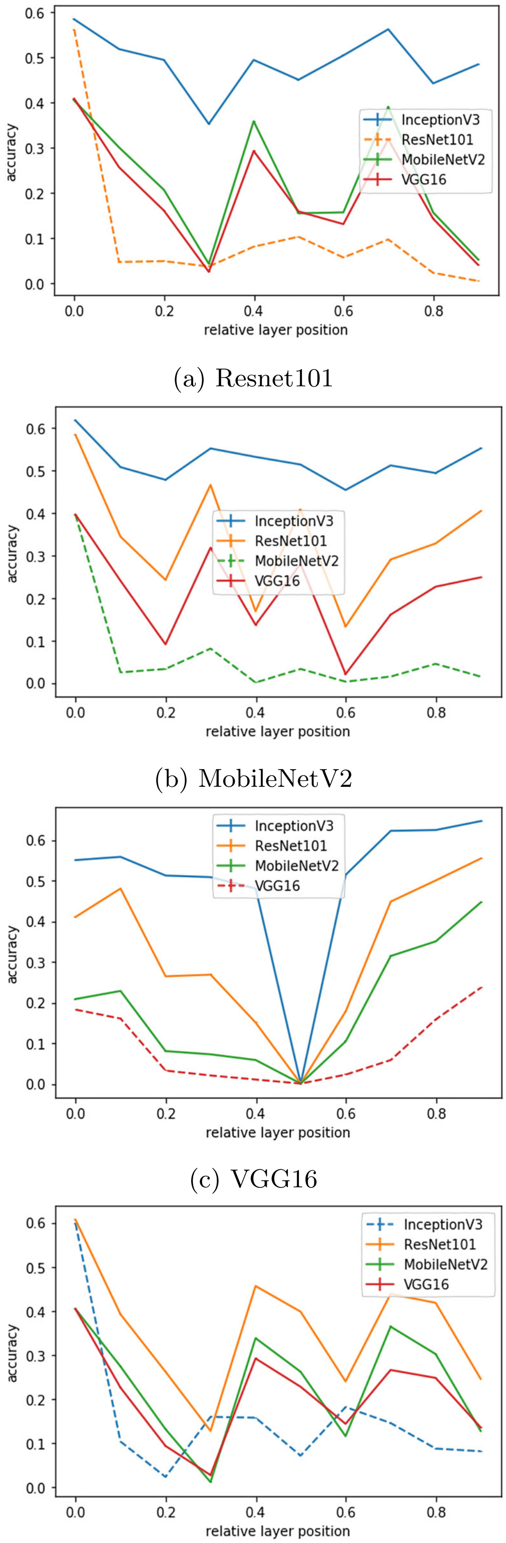
Important point to address is the attack budget. This parameter defines the degree to which an attack is allowed to alter each pixel of the image, or in other words its maximal L1 norm. For all our experiments, we set this parameter equal to 16/255. The authors of the DR approach used this value in their experiments, and to make comparison of DR and DA as fair as possible, we decided to use the same value of attack budget. Attack budget itself does not mean that each attack will result in an image with L1 distance equal to 16/255; still, it is reasonable to assume that the attacker will search for the best attack within the attack budget„ and in most of the cases, it results in an image with the L1 norm as close to attack budget. We have tested this assumption for both DA and DR attacks on the subset of ImageNet dataset and have not observed cases when an attack would finish on the L1 norm far from 16/255. Because of this, we decided not to evaluate the average L1 norm as a separate metric in the experiments. We also used η equal 2 in all of the experiments as on the small subset of data it has shown best performance.

In our first experiment, we analyze how the selection of surrogate model architecture and target layer influences attack effectiveness. We made the experiment several times using different architectures, including VGG16 [19], ResNet101 [20], InceptionV3 [21], and MobileNetV2 [22]. Each time, one model was selected as a surrogate and other models as attack targets. First, we trained the surrogate and target models on ImageNet train subset and measured the accuracy on clean data from validation set. For the evaluation of attack transferability, we refer to the target metrics of each task on the attacked images and compare those numbers to each other. The bigger drop in the target metric the better transferability of the attack. We selected this method because we investigate cross-task transferability and attack effectiveness metric should be clearly interpretable for each of the tasks (classification, detection, segmentation).

Then, one-by-one, we selected each convolutional layer of the surrogate model as the target layer and performed dispersion amplification attack to obtain adversarial version of validation subset. We measured the accuracy of both surrogate and target models on obtained adversarial data to evaluate how many of generated adversarial images were able to fool the surrogate model and how many of them managed to pass to the next neural network architectures. The obtained results are presented in Fig. 2.

The x axis of each figure represents the depth of target layer relative to the overall model deepness. Here, zero represents models with attack on the first convolutional layer, 1 - of the last one. For every architecture, we obtained similar results. Attack effectiveness on surrogate model tends to increase together with the target layer depth. At the same time, its transferability rate grows first, then falls, achieving its maximum at middle layers.

Deep layers learn more high-level image features. Thus, using the activations from these layers allows to alter model predictions with smaller perturbations, despite that these features are typically unique for each architecture. This leads to high attack effectiveness, but poor transferability. However, features learned at middle layers are complex enough to be used as the target for adversarial



(a) Resnet101

(b) MobileNetV2

(c) VGG16

(d) InceptionV3

Fig. 2 Results of dispersion amplification attacks using different substitute models. Each plot represents a different target model while each line is a different substitute model

attack, but at the same time they are general enough to be learned by different architectures.

Thus, we investigated how the number of iterations influences attack effectiveness. To measure it, we repeated the first experiment while choosing the target layer to be on the position approximately totallayers × 0.45. Figure 3 shows how substitute model's and target model's accuracy changed over number of iterations. Based on the obtained results, 100 iterations is enough to achieve decent success rate. Taking more than 500 iterations is already not reasonable.

To evaluate the attack transferability across different computer vision tasks, we performed dispersion amplification attack against several open source models, including FastRCNN [23], MaskRCNN [24], and YoloV3 [25] for object detection, and RCNN [26] and DeepLabV3 [27] for semantic segmentation. We used the validation subset of COCO 2012 dataset to evaluate the object detection and semantic segmentation. mAP metric was used in both cases. The model from the first experiment, pre-trained on ImageNet, was used as a substitute model. The target layer and the number of iterations were chosen as totallayer × 0.45 and 200, respectively. To compare dispersion amplification with other methods, we performed FGSM and BIM attacks defined as follows:

$$\text{FGSM:}\tilde{x} = x + \varepsilon \cdot \text{sign}\nabla_x l(\tilde{x}, t) \tag{4}$$

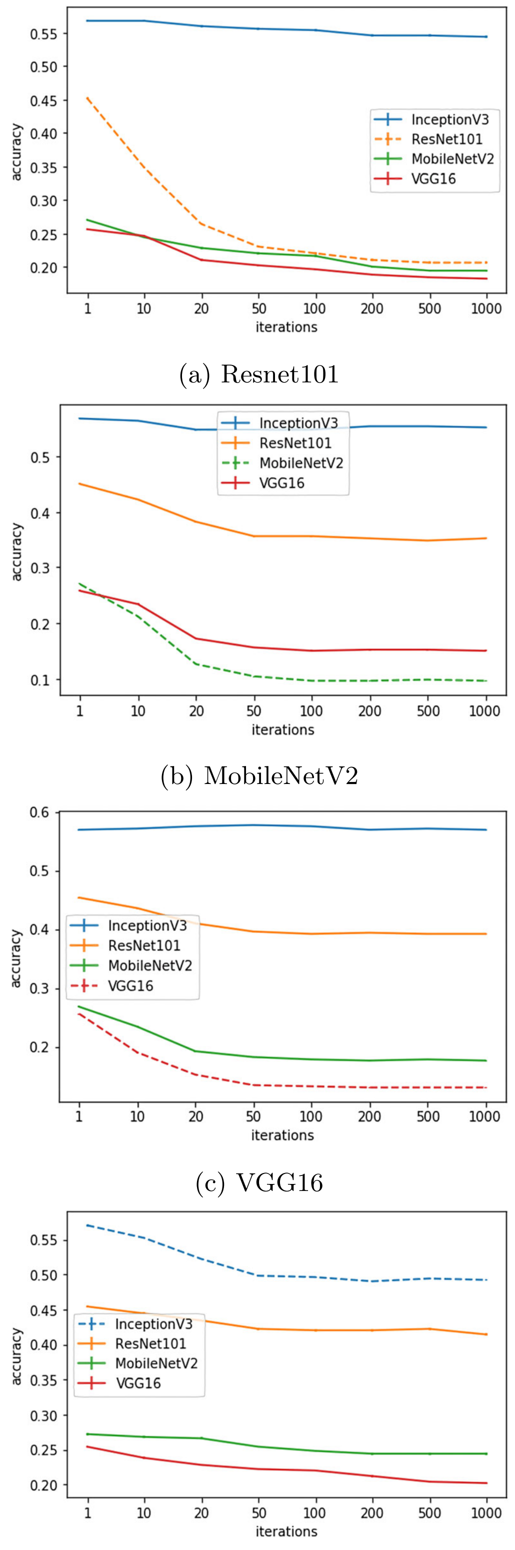
$$\text{BIM:}\tilde{x}_0 = x,\ \tilde{x}_k = \text{clip}(\tilde{x}_{k-1} + \varepsilon \cdot \text{sign}\nabla_x l(\tilde{x}_{k-1}, t)) \tag{5}$$

The obtained results are presented in Table 1.

6 Discussion

Table 1 shows that both dispersion reduction and dispersion amplification attacks have shown higher attack effectiveness than FGSM, BIM, and LBFGS, while being agnostic of the task they were targeted on. At the same time, our method has shown better results compared to dispersion reduction in most of the attack setups. This improvement in attack effectiveness varies from marginal to pretty significant, like in the case of DeepLab segmentation model.

To understand why DA performs better than DR in many cases, we need to understand the nature of neuron activation distribution. Works on pruning like [28] have shown that during training of the DNN, there is high redundancy in the feature maps with some of the feature maps almost duplicating. During pruning, we take advantage of this redundancy to leave only informative feature maps and prune duplicates. But for the tasks of DA and DR, we utilize this redundancy in a different way. There are some positively and negatively correlated fea-



(a) Resnet101

(b) MobileNetV2

(c) VGG16

(d) InceptionV3

Fig. 3 Results of dispersion amplification attacks using different number of iterations. Each plot represents a different target model while each line—a different substitute model

Table 1 Experiment results

Base model	Attack	Det. YoloV3 mAP	Det. Mask-RCNN mAP	Det. Faster-RCNN mAP	Seg. DeepLab mAP	Classification	PSNR
VGG16	FGSM	0.541	0.506	0.522	0.599	0.483	30.01
	BIM	0.522	0.517	0.512	0.632	0.349	25.86
	LBFGS	–	0.542	0.49	0.538	–	25.51
	DR	0.126	0.586	**0.136**	0.411	**0.112**	**30.96**
	DA	**0.119**	**0.144**	**0.136**	**0.128**	0.128	28.32
ResNet101	FGSM	0.612	0.578	0.607	0.622	0.381	30.01
	BIM	0.628	0.571	0.567	0.585	0.269	26.68
	LBFGS	–	0.56	0.553	0.547	–	26.73
	DR	0.138	0.129	0.131	0.530	0.116	**32.69**
	DA	**0.105**	**0.112**	**0.112**	**0.174**	**0.060**	30.33
InceptionV3	FGSM	0.595	0.613	0.585	0.594	0.520	30.01
	BIM	0.601	0.609	0.611	0.591	0.448	26.81
	LBFGS	–	0.515	0.604	0.549	–	26.56
	DR	0.132	0.130	0.111	0.490	0.074	**33.31**
	DA	**0.126**	**0.112**	**0.110**	**0.216**	**0.032**	30.68
MobileNet v2	FGSM	0.622	0.641	0.629	0.672	0.325	30.01
	BIM	0.570	0.595	0.601	0.639	0.338	26.38
	LBFGS	–	0.553	0.575	0.539	–	25.83
	DR	0.147	0.147	0.133	0.221	0.104	**30.55**
	DA	**0.112**	**0.109**	**0.125**	**0.147**	**0.092**	29.98

ture maps in each layer of the network like in the example we obtained from the MobileNetV2 Fig. 4.

During DR attack, the goal is to bring all the activations to the one value effectively reducing dispersion to zero. Having negatively correlated activations makes this task hard as to achieve it one would require to bring those negatively correlated activations together and close to positively correlated ones. On the other hand, DA takes advantage of the negatively correlated feature existence. To achieve the goal, DA does not require changing of activation correlation from negative to positive. Taking those negatively correlated activations and dragging their activations in the different directions would not affect the correlation of the activation maps but will effectively amplify dispersion. The result of this effect is that with the same attack budget, DA is able to achieve higher distortion in the activation distribution space which results in higher transferability.

Another interesting observation can be inferred from Fig. 2. It is related to the NN architectures themselves. One can observe that attack effectiveness tends to fluctuate for some architectures while for others do not. For the plot corresponding to VGG16 attack effectiveness first rises that falls monotonically. There is a clear optimal point that appears approximately in the middle of the plot (c). The reason is the deeper one gets into the VGG16, the more high-level features there one obtains. Mid-level features become optimum because they are deep enough to catch some mid- to high-level features that are not specific to the architecture. In cases (a) Resnet101, (b) MobileNetV2, and (d) InceptionV3, optimum is not that clear and attack effectiveness fluctuates. The reason for such behavior might be in utilization of skip connections in (a, b) and the nature of the Inception block in (d). Because of the skip connections model indirectly becomes an ensemble of networks with different depths, which itself makes hierarchy of the learned features less straightforward than VGG16. Also, at the same time, the whole idea behind the inception block is to incorporate features of different scales on the same level of activation. It results

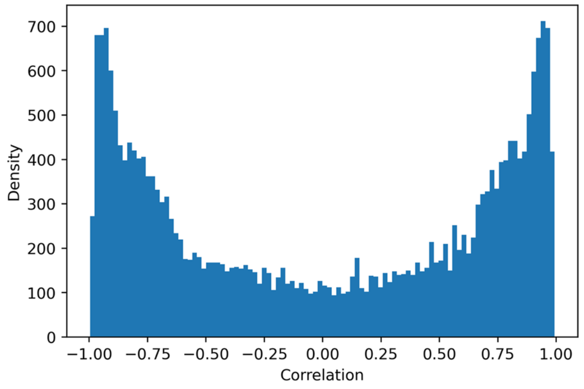
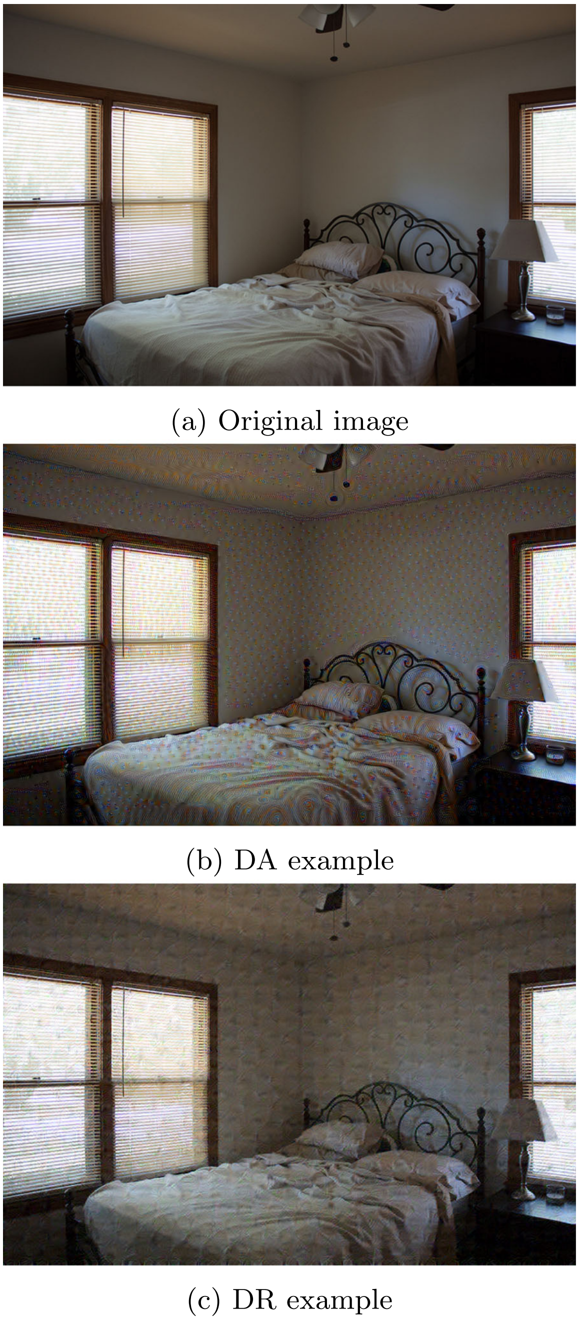


Fig. 4 Correlation between activations of 0.4 layer of MobileNetV2

in feature hierarchy unique in comparison to other evaluated networks which itself may be the reason for fluctuations in the attack effectiveness. Figure 3 also shows that InceptionV3 is the architecture most resilient to our attack; again, we tend to think that the mechanism of incorporating multi-scale features in one inception block is the reason for that.

Besides the metrics, it is interesting to compare the results of DR and DA attacks visually. Figure 5 shows the examples of attacked images for DA and DR attacks. Visual perceptibility of perturbations by the human is outside of the scope of our research as in target use cases our goal is to evade automatic detection mechanisms by AI rather than the human eye. Still, there are few interesting outcomes that can be addressed. Attack examples obtained with DR have bigger perturbation patterns compared with DA examples. At the same time, perturbations of DR mostly look like bleak/gray zones on the image that in some cases look like artifacts created by low-resolution camera, while DA results in more bright colors more common to adversarial examples. The metric that helps to indirectly evaluate visual perception of the attacked images is peak signal-to-noise ratio (PSNR). This metric represents the ratio between the maximum possible power of a signal and the power of perturbation noise that affects the fidelity of its representation. We have observed that PSNR value in all our experiments was best for DR with DA being slightly behind. This checks with our perception of DA and DR examples, as DR do not produce bright color perturbation. Also, when viewing images produced by DA attack, one can see a familiar patterns. DA images came out really familiar to the DeepDream [29] approach. After the detailed analysis, it became obvious why. DeepDream was used to visualize what exact features neural network learns in the process of training. To do so, researchers fixated a set of neurons and was generating image with maximized values of activations for this neurons. As a result, image was transformed with pattern-specific neuron tends to detect. In case of DA, we do not fixate specific neuron but do fixate specific layer of the network and while amplifying the dispersion within the budget we tend to maximize at least some of the neurons. Because of close ideas, images from DeepDream and DA might share similarities in some cases.



(a) Original image

(b) DA example

(c) DR example

Fig. 5 Examples of attacked images with DA and DR attacks. Both attacks performed on the 0.4 of the NN depth. Target network for those examples was MobileNetV2

7 Conclusion

In this paper, we proposed dispersion amplification attack as a method for adversarial attack against real-world machine learning solutions. It was based on cross-task transferability of perturbations applied to the middle layers of neural network. We compared our method with other methods based on cross-task transferability principle, namely BIM (I-FGSM), LBFGS, FGSM, and dispersion reduction.

We have measured proposed attack effectiveness across different computer vision tasks, including object detection, semantic segmentation, and image classification. We observed that our approach outperforms all these known attacks in black-box scenario. Proposed attack is able to significantly degrade the performance of target machine learning system without any knowledge about its internal architecture and training dataset.

We assume that the limited feature space of neural networks middle layers forces networks with different architectures to memorize similar features. Amplifying variance on such layers will make all subsequent layers treat modified activations as out-of-distribution, efficiently forcing them to make incorrect predictions.

For the future, we consider to investigate the application of proposed approach for privacy protection of the user on the Internet. Low level of resulting perturbation and high transferability of the attack between tasks and models makes it a great instrument to protect the user from unnecessary detection and tracking on the Internet. Such anonymization application shall be useful to mitigate fraud schemes and cyber bullying. We see it as a great non-malicious real-world application for adversarial attacks.

Abbreviations

DNN: Deep neural network; FGSM: Fast Gradient Sign Attack; BIM: Basic Iterative Method; AI: Artificial intelligence; API: Application Programming Interface; ML: Machine learning; MTA: Multitask adversarial attack; GAN: Generative adversarial network; CNN: Convolutional neural network; mAP: Mean average precision

Acknowledgements

Not applicable.

Authors' contributions

Oleksandra Sokol performed the initial set of experiments that has shown applicability of dispersion amplification approach. Pavlo Haleta and Dmytro Likhomanov planned and performed the main part of the experiments. Pavlo Haleta has drafted the first version of the manuscript. Dmytro Likhomanov updated the manuscript to the final version. All authors read and approved the final manuscript.

Funding

Research was conducted on funds of Samsung R&D Institute Ukraine, Kyiv.

Availability of data and materials

We used open datasets such as COCO 2012 [18] and ImageNet [17] in our study. Additionally, we utilized open neural network architectures such as VGG16 [19], ResNet101 [20], InceptionV3 [21], MobileNetV2 [22], FastRCNN [23], MaskRCNN [24], YoloV3 [25], RCNN [26], and DeepLabV3 [27].

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 1 April 2021 Accepted: 18 August 2021
Published online: 22 September 2021

References

1. Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, V. C. M. Leung, A survey on security threats and defensive techniques of machine learning: a data driven view. IEEE Access. **6**, 12103–12117 (2018)
2. I. J. Goodfellow, J. Shlens, C. Szegedy, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, ed. by Y. Bengio, Y. LeCun. Explaining and harnessing adversarial examples, (2015). http://arxiv.org/abs/1412.6572
3. I. Sergeev, How do we deal with copying content, or the first adversarial attack in production (2019). https://habr.com/en/company/avito/blog/452142/
4. Y. Lu, Y. Jia, J. Wang, B. Li, W. Chai, L. Carin, S. Velipasalar, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Enhancing cross-task black-box transferability of adversarial examples with dispersion reduction, (2020), pp. 937–946. https://doi.org/10.1109/CVPR42600.2020.00102
5. P. Y. Chen, H. Zhang, Y. Sharma, J. Yi, C. J. Hsieh, in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security AISec '17*. Zoo: zeroth order optimization based black-box attacks to deep neural networks without training substitute models (Association for Computing Machinery, New York, NY, USA, 2017), pp. 15–26. https://doi.org/10.1145/3128572.3140448
6. A. Ilyas, L. Engstrom, A. Athalye, J. Lin, Black-box adversarial attacks with limited queries and information. CoRR. **abs/1804.08598**, 3–5 (2018). http://arxiv.org/abs/1804.085981804.08598
7. A. N. Bhagoji, W. He, B. Li, D. Song, in *Computer Vision – ECCV 2018*, ed. by V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss. Practical black-box attacks on deep neural networks using efficient query mechanisms (Springer International Publishing, Cham, 2018), pp. 158–174
8. Y. Dong, F. Liao, T. Pang, X. Hu, J. Zhu, Discovering adversarial examples with momentum. CoRR. **abs/1710.06081** (2017). http://arxiv.org/abs/1710.060811710.06081
9. N. Papernot, P. D. McDaniel, I. J. Goodfellow, S. Jha, Z. B. Celik, A. Swami, Practical black-box attacks against deep learning systems using adversarial examples. CoRR. **abs/1602.02697**, 3–5 (2016). http://arxiv.org/abs/1602.026971602.02697
10. Y. Liu, X. Chen, C. Liu, D. Song, Delving into transferable adversarial examples and black-box attacks. CoRR. **abs/1611.02770** (2016). http://arxiv.org/abs/1611.02770
11. W. Zhou, X. Hou, Y. Chen, M. Tang, X. Huang, X. Gan, Y. Yang, in *Computer Vision – ECCV 2018*, ed. by V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss. Transferable adversarial perturbations (Springer International Publishing, Cham, 2018), pp. 471–486
12. P. Guo, Y. Xu, B. Lin, Y. Zhang, Multi-task adversarial attack. CoRR. **abs/2011.09824** (2020). https://arxiv.org/abs/2011.09824
13. S. M. Moosavi-Dezfooli, A. Fawzi, P. Frossard, in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. Deepfool: a simple and accurate method to fool deep neural networks (IEEE Computer Society, Las Vegas, NV, USA, 2016), pp. 2574–2582. https://doi.org/10.1109/CVPR.2016.282
14. M. Barni, K. Kallas, E. Nowroozi, B. Tondi, in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. On the transferability of adversarial examples against cnn-based image forensics, (2019). https://doi.org/10.1109/ICASSP.2019.8683772
15. M. Barni, E. Nowroozi, B. Tondi, B. Zhang, in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Effectiveness of random deep feature selection for securing image manipulation detectors against adversarial examples, (2020). https://doi.org/10.1109/ICASSP40776.2020.9053318
16. A. Kurakin, I. J. Goodfellow, S. Bengio, in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. Adversarial machine learning at scale (OpenReview.net, 2017). https://openreview.net/forum?id=BJm4T4Kgx
17. J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, L. Fei-Fei, in *IEEE Conference on Computer Vision and Pattern Recognition*. Imagenet: a large-scale hierarchical image database, (2009). https://doi.org/10.1109/CVPR.2009.5206848
18. T.-Y. Lin, L. M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, in *Computer Vision – ECCV 2014*, ed. by D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars. Microsoft coco: common objects in context (Springer International Publishing, Cham, 2014), pp. 740–755
19. K. Simonyan, A. Zisserman, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, ed. by Y. Bengio, Y. LeCun. Very deep convolutional networks for large-scale image recognition, (2015). http://arxiv.org/abs/1409.1556
20. K. He, X. Zhang, S. Ren, J. Sun, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Deep residual learning for image recognition, (2016). https://doi.org/10.1109/CVPR.2016.90
21. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Rethinking the inception architecture for computer vision, (2016). https://doi.org/10.1109/CVPR.2016.308
22. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. C. Chen, in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mobilenetv2: inverted residuals and linear bottlenecks, (2018). https://doi.org/10.1109/CVPR.2018.00474

23. R. Girshick, in *IEEE International Conference on Computer Vision (ICCV)*. Fast r-cnn, (2015). https://doi.org/10.1109/ICCV.2015.169
24. K. He, G. Gkioxari, P. Dollar, R. Girshick, in *2017 IEEE International Conference on Computer Vision (ICCV)*. Mask R-CNN, (2017), pp. 2980–2988. https://doi.org/10.1109/ICCV.2017.322
25. J. Redmon, A. Farhadi, Yolov3: an incremental improvement (2018). http://arxiv.org/abs/1804.02767
26. R. Girshick, J. Donahue, T. Darrell, J. Malik, in *IEEE Conference on Computer Vision and Pattern Recognition*. Rich feature hierarchies for accurate object detection and semantic segmentation, (2014). https://doi.org/10.1109/CVPR.2014.81
27. L. C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking Atrous convolution for semantic image segmentation. CoRR. **abs/1706.05587** (2017). http://arxiv.org/abs/1706.05587
28. L. Valerio, F. M. Nardini, A. Passarella, R. Perego, Dynamic hard pruning of neural networks at the edge of the internet. CoRR. **abs/2011.08545** (2020). https://arxiv.org/abs/2011.08545
29. A. Mordvintsev, C. Olah, M. Tyka, Inceptionism: going deeper into neural networks (2015). https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.