ENHANCING TARGETED TRANSFERABILITY VIA SUPPRESSING HIGH-CONFIDENCE LABELS

Hui Zeng[1,2], *Tong Zhang*[1], *Biwei Chen*[3], *and Anjie Peng*[1,2*]

[1]Southwest University of Science and Technology [3]Beijing Normal University
[2]Guangdong Provincial Key Laboratory of Information Security Technology

ABSTRACT

While extensive studies have pushed the limit of the transferability of untargeted attacks, transferable targeted attacks remain extremely challenging. This paper finds that the labels with high confidence in the source model are also likely to retain high confidence in the target model. This simple and intuitive observation inspires us to carefully deal with the high-confidence labels in generating targeted adversarial examples for better transferability. Specifically, we integrate the untargeted loss function into the targeted attack to push the adversarial examples away from the original label while approaching the target label. Furthermore, we suppress other high-confidence labels in the source model with an orthogonal gradient. We validate the proposed scheme by mounting targeted attacks on the ImageNet dataset. Experiments on various scenarios show that our proposed scheme improves the state-of-the-art targeted attacks in transferability. Our code is available at: *https://github.com/zengh5/Transferable_targeted_attack*.

Index Terms—adversarial examples, targeted attack, high-confidence labels, targeted transferability

1. INTRODUCTION

Adversarial examples (AEs), designed to reveal the potential weakness of neural networks, have attracted more and more attention since they were proposed [1]. A fascinating property of AEs is their transferability, i.e., an AE crafted on one source model may also fool an unknown target model. Numerous works have been proposed in pursuit of better transferability. To avoid the obtained AEs overfitting the source model, some researchers seek better data augmentation strategies, e.g., diverse inputs (DI) [2], translation-invariant (TI) [3], and scale-invariant (SI) [4]. To prevent AEs from falling into poor local maxima, other researchers adopt better optimization algorithms, e.g., momentum iterative (MI) [5], Nesterov iterative (NI) [4], and PID-based approach [6].

Although AEs' transferability has been improved remarkably, existing works mainly focus on non-targeted attacks. Targeted transferability is much more challenging since it requires the output of an unknown model to be a specific label [7, 8]. Tailored schemes for improving the transferability of targeted attacks have been proposed to fill the gap. Some adopt new loss functions to avoid the decreasing gradient issue of targeted attacks [9, 10]. Others seek extra classifiers [11] or target-specific generative adversarial networks [12] to optimize adversarial perturbations. Even though these enhanced algorithms can improve the targeted transferability to a certain extent, we find that they more or less ignore the confidence distribution in the source model. This paper points out two types of high-confidence labels most likely to cause transfer attack failures. One is the original label, and the other includes labels that show high confidence in the source model after attacking. Suppressing these high-confidence labels can further improve targeted transferability. Considering this, we propose a novel targeted attack illustrated in Figure 1. As can be seen, we combine the gradients of the original label and other high-confidence labels with the gradient of the target label. The combined gradient pushes the AEs away from high-confidence labels while approaching the target label.

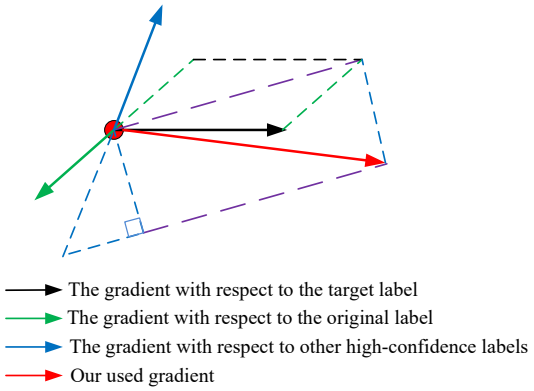


Fig. 1. Illustration of the proposed method. Our used gradient pushes the adversarial examples away from high-confidence labels while approaching the target label.

The main contributions can be summarized as follows: 1) Highlighting two types of high-confidence labels most likely to cause transfer attack failures. 2) A novel attacking scheme simultaneously suppresses the high-confidence labels' confidence while enhancing the target labels' confidence. 3) Experiments demonstrate that the proposed scheme consistently improves the state-of-the-art in targeted transferability.

2. RELATED WORK

2.1. Transferable untargeted attack

An untargeted attack aims to lead a convolutional neural

This work was supported by the network emergency management research special topic (no. WLYJGL2023ZD003), the Opening Project of Guangdong Province Key Laboratory of Information Security Technology (no. 2020B1212060078), Sichuan Science and Technology Program (no. 2022YFG0321).

network (CNN) model $F()$ into making a wrong classification, i.e.,$F(\boldsymbol{I}') \neq F(\boldsymbol{I})$, where $\boldsymbol{I}$ is the original image, $\boldsymbol{I}'$is the adversarial image. As a common baseline of further variants, the iterative fast gradient sign method (IFGSM) [13] can be formulated as follows:

$$\boldsymbol{I}'_0 = \boldsymbol{I}$$
$$\boldsymbol{I}'_{n+1} = Clip_{\boldsymbol{I},\epsilon}\{\boldsymbol{I}'_n + \alpha sign(\nabla_{\boldsymbol{I}'_n} J(\boldsymbol{I}'_n, y_o))\} \quad (1)$$

where $\nabla_{\boldsymbol{I}'_n} J()$ denotes the gradient of the loss function $J()$ with respect to $\boldsymbol{I}'_n$, y_o is the original label. The accumulated perturbation for each pixel is restricted to $[-\epsilon, \epsilon]$ by $Clip_{\boldsymbol{I},\epsilon}\{\}$.

To enhance AE's transferability, researchers have proposed a variety of improved algorithms for IFGSM, e.g., [5] integrates a momentum term into the iterative process:

$$g_{n+1} = \mu \cdot g_n + \nabla_{\boldsymbol{I}'_n} J(\boldsymbol{I}'_n, y_o)$$
$$\boldsymbol{I}'_{n+1} = Clip_{\boldsymbol{I},\epsilon}\{\boldsymbol{I}'_n + \alpha sign(g_{n+1})\} \quad (2)$$

where g_n is the accumulated gradient at iteration n, μ is a decay factor. On the other hand, the DI attack [2] strives to enhance the transferability of the AEs from data augmentation, such as scaling and padding. The TI attack [3] adopts a smoothed gradient to prevent the attack from overfitting a specific source model:

$$\boldsymbol{I}'_{n+1} = Clip_{\boldsymbol{I},\epsilon}\{\boldsymbol{I}'_n + \alpha sign(\boldsymbol{W} * \nabla_{\boldsymbol{I}'_n} J(\boldsymbol{I}'_n, y_o))\} \quad (3)$$

where $\boldsymbol{W}$ is a convolution kernel for smoothing. Moreover, these enhanced schemes can be integrated for even better transferability, e.g., Translation Invariant Momentum Diverse Inputs IFGSM (TMDI). Interested readers can refer to [14] for a thorough overview of the variants of IFGSM. Note that although these enhanced schemes are initially proposed for untargeted attacks, they also contribute to targeted transferability.

2.2. Transferable targeted attack

A targeted attack misguides a CNN model to a specific label as the attacker intends, i.e., $F(\boldsymbol{I}') = y_t$, where y_t is the target label. In addition to the transferable schemes reviewed in the last section, there are also tailored transferable methods for targeted attacks that emerged recently.

In [9], a Po+Trip loss-based method is proposed. As it implies, this method consists of two parts. The first one is using the Poincare distance loss in lieu of the traditional cross-entropy (CE) loss to address the decreasing gradient problem.

$$L_{Po} = arccosh(1 + \delta(\boldsymbol{u}, \boldsymbol{v})),$$
$$\delta(\boldsymbol{u}, \boldsymbol{v}) = \frac{2 \cdot ||\boldsymbol{u} - \boldsymbol{v}||_2^2}{(1 - ||\boldsymbol{u}||_2^2)(1 - ||\boldsymbol{v}||_2^2)} \quad (4)$$

where $\boldsymbol{u}$ is the normalized logit vector, and $\boldsymbol{v}$ is the one-hot vector with respect to y_t. The second part introduces a triplet loss to push the attacked image away from y_o.

$$L_{Trip} = [D(l(\boldsymbol{I}'), y_t) - D(l(\boldsymbol{I}'), y_o) + \gamma]_+,$$
$$D(l(\boldsymbol{I}'), y) = \frac{||l(\boldsymbol{I}') \cdot y||_1}{||l(\boldsymbol{I}')||_2 ||y||_2} \quad (5)$$

where $l(\cdot)$ denotes the logit output vector. The final loss is a weighted sum of (4) and (5), i.e., $L_{Po+Trip} = L_{Po} + \lambda L_{Trip}$, where λ is a predefined weight with a default value of 0.01. [10] uses the Logit loss in the attack and reports better transferability than the CE or Po+Trip loss.

$$L_{Logit} = -l_t(\boldsymbol{I}') \quad (6)$$

where $l_t(\cdot)$ denotes the logit output with respect to y_t. [12] trains an input-adaptive generator function to synthesize targeted perturbation and achieves state-of-the-art transferability. However, a dedicated generator must be learned for every (*source model*, *target class*) pair in [12]. Henceforth, we denote the traditional CE loss-based attack, the Po+Trip loss-based attack, the logit loss-based attack, and the transferable targeted perturbation as CE, Po+Trip, Logit, and TTP, respectively.

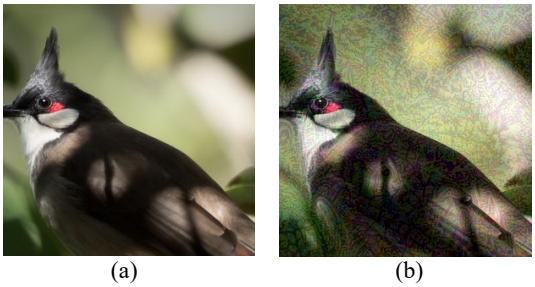
3. PROPOSED SCHEME

3.1. Motivation

Our method is motivated by the following observations:

1) *The original label of an AE is likely to be recovered when it is transferred to a target model*. Such observation can be illustrated with an example in Fig. 2. Fig. 2(a) is an image whose original label is 'bulbul.' After a TMDI attack (target to 'church,' Fig. 2(b)), its confidence to 'bulbul' is effectively suppressed to zero in the source model as expected. However, when the target model predicts the attacked image, its confidence to 'bulbul' is restored to 0.87.

2) *High-confidence labels in the source model are likely to retain high confidence in the target model*. To illustrate this phenomenon, we compare the confidence of different labels on the ImageNet-Compatible Dataset [15]. A pretrained ResNet50 model [16] acts as the source model, and target models include Inceptionv3 [17], DenseNet121 [18], and VGG16 [19]. We regard the labels whose confidence rank from 2 to 500 (y_o is excluded) in the source model as high-confidence labels, and those rank from 501 to 1000 as low-confidence. Fig. 3 compares the average confidence on different target models for both benign (left) and TMDI-attacked images (right). The plots clearly show that high-confidence labels (of the source model) also have higher confidence in the target model.



(a) (b)

Fig. 2. An example that the original label of an attacked image is restored in transferring to a target model (Inceptionv3). (a) the original image, whose confidence with respect to 'bulbul' in the source model (ResNet50) is 0.96, (b) TMDI-attacked image, whose confidence with respect to 'bulbul' in the source model is 0.0, and 0.87 in the target model.

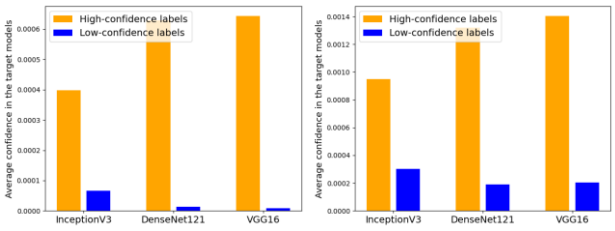


Fig. 3. Confidence comparison on the target models. For each pair of bars, the left bar shows the average confidence of the high-confidence labels and the right one of the low-confidence labels. The high/low-confidence labels are calculated on a pretrained ResNet50 model (source model).

A transferable targeted attack requires the target label's confidence to be the highest in the target model, which may not be satisfied due to the influence of high-confidence labels. Hence, suppressing the confidence of high-confidence labels in the target model is crucial for the transferability of AEs.

3.2. Suppressing the confidence of the original label

Based on the analysis above, we first suppress the confidence of y_o. Due to the restoring effect, suppressing y_o's confidence to an average level is not enough. Hence, we propose a combined loss function as follows:

$$L_{combine} = -(l_t(\boldsymbol{I}') - \beta_1 l_o(\boldsymbol{I}')) \quad (7)$$

where $l_o(\cdot)$ denotes the logit output with respect to y_o, and β_1 is a predefined weight. In this way, we essentially carry out both targeted and untargeted attacks. At first blush, the logic of (7) is similar to the triplet loss (5). However, they are different. Let $\boldsymbol{v}_o$ be the one-hot vector with respect to y_o, the triplet loss pushes $l_o(\boldsymbol{I}')$ away from $\boldsymbol{v}_o$ in the *orthogonal* direction, whereas (7) pushes $l_o(\boldsymbol{I}')$ away from $\boldsymbol{v}_o$ in the *opposite* direction.

3.3. Suppressing other high-confidence labels

Next, we continue to suppress the confidence of other high-confidence labels. An intuitive solution is to integrate the logits with respect to the high-confidence labels

$$l_{high-conf}(\boldsymbol{I}') = \sum_{i=0}^{N_h} l_{high-conf,i}(\boldsymbol{I}'), \quad (8)$$

into (7), where N_h is the number of high-confidence labels to suppress. However, the high-confidence labels calculated on $\boldsymbol{I}'$ often strongly correlate with the target label. Taking an attacked image $\boldsymbol{I}'$ whose target label is 'bulbul' for example, its high-confidence labels may include bird-related classes such as 'jay' and 'jacamar.' Hence, suppressing the confidence of high-confidence labels may also weaken the confidence of y_t. To address this issue, we only keep the orthogonal component of $\nabla(l_{high-conf}(\boldsymbol{I}'))$ to $\nabla L_{combine}$

$$\nabla L_{combine} \perp = \nabla\left(l_{high-conf}(\boldsymbol{I}')\right) - \text{proj}_{\nabla\left(l_{high-conf}(\boldsymbol{I}')\right)} \nabla L_{combine} \quad (9)$$

The final gradient used for updating $\boldsymbol{I}'$ can be written as

$$\nabla L_{combine} + \beta_2 \nabla L_{combine} \perp \quad (10)$$

Algorithm 1 Proposed transferable targeted attack

Input: A benign image $\boldsymbol{I}$ with original label y_o; target label y_t.

Parameter: Total iteration number N; the timing T of introducing the orthogonal gradient; the number of high-confidence labels N_h to suppress.

Output: Adversarial image $\boldsymbol{I}'$.

1. Mounting attack with the loss function defined in (7) for $T \times N$ iterations, and obtain an intermediate image $\boldsymbol{I}^{inter}$.
2. Extracting N_h high-confidence labels from $\boldsymbol{I}^{inter}$.
3. Mounting attack with the gradient defined in (10), for the remaining process, and obtain the final adversarial image $\boldsymbol{I}'$.

where β_2 is a predefined weight used to balance these two terms. In this way, we prevent the newly introduced loss from contradicting with $L_{combine}$.

Our transferable targeted attack is summarized in **Algorithm 1**. We split the whole attack into two steps. The first step is guided by the loss function of (7), aiming to enhance the confidence of y_t and suppress the confidence of y_o simultaneously. After this step, an intermediate image is obtained, from which the high-confidence labels are calculated. The second step performs an attack with the gradient in (10), aiming to suppress the confidence of other high-confidence labels further.

4. EXPERIMENTAL RESULTS

We compare the proposed method with three iterative attacks: CE, Po+Trip, Logit, and a generative attack, TTP. All the iterative schemes start with the TMDI attack.

4.1. Experiment Settings

Dataset. Our experiments are conducted on the ImageNet-compatible dataset comprised of 1000 images. All these images are cropped to 299×299 pixels before use.

Networks. Since transferring across different architectures is more challenging, we use four pretrained models of diverse architectures: Inceptionv3, ResNet50, DenseNet121, and VGG16 as in [10] to evaluate AEs' transferability.

Parameters. For all competitors, the perturbations are restricted by L_∞ norm with $\epsilon = 16$, and the step size is set as 2. [10] suggests that targeted attacks need more iterations to converge than untargeted attacks. Hence, the total iteration number N is set to 200. In our method, we set $\beta_1 = 0.2$, and $\beta_2 = 0.5$. The number of high-confidence labels N_h is set as 10, and the timing T of introducing the orthogonal gradient ($\nabla L_{combine} \perp$) is set as 0.75 unless otherwise mentioned. Ablation study on N_h and T and more detailed results are provided in: *Transferable_targeted_attack/supp.pdf*.

4.2. Single-model transfer

Table 1 reports the targeted transferability across different models. As can be seen, the proposed method outperforms the state-of-the-art methods by a large margin in almost all cases. For example, when transferring from DenseNet121 to ResNet50, the proposed method improves by 29.1%, 32.9%,

and 5.5% over CE, Po+Trip, and Logit, respectively. As reported in [10], we also find that the Inception-v3 model is the most difficult to transfer. It might be because the Inception architecture is the most complicated among the four, and transferring from a simple architecture to a complicated one is usually more challenging than the opposite.

Furthermore, we consider a worst-case transfer scenario in which the target is always specified as the least-likely label of the benign image. Table 2 compares different attacks in such a challenging scenario. Again, the proposed method trumps almost all cases. By comparing the last column of Tables 1 and 2, we surprisingly observed that the attack success rates in the random-target scenario are lower than in the most difficult-target scenario when Inceptionv3 is the source model. By examining the confidence distribution of the attacked images in the target models, we find that the original label is less likely to be restored after model transfer when targeted to the least-likely label in this case. This discovery further indicates that the targeted transferable attack has its uniqueness.

4.3. Ensemble transfer

Attacking an ensemble of source models has been proven to improve the transferability of AEs further. As in [10], we assign equal weights to all source models. Table 3 reports the targeted transferability in this ensemble-model scenario. The proposed method consistently achieves the best transferability among the compared methods. When averaging different target models, the proposed method improves by 9.9%, 17.3%, and 2.9% upon CE, Po+Trip, and Logit, respectively. Consistent with [10], we also note that the transferability of the Po+Trip attack is the weakest in the ensemble-model scenario.

4.4. Iterative vs. generative attacks

Since TTP needs to train a specific generator for every (source model, y_t) pair, 4×1000 generators are required to perform the random or most difficult-target attack, which is impractical for us. Alternatively, we download the pre-trained generators and follow the '10-Targets' setting of [12] to compare the iterative methods with TTP. As shown in Table 4, the proposed attack achieves comparable (ϵ =16) or even better (ϵ =4, 8) transferability to TTP.

Table 3. Targeted transfer success rate (%) in the ensemble-model, random-target scenario, where '−' indicates the hold-out model.

Attack	−Inc-v3	−Res50	−Dense121	−VGG16	Average
CE	24.4	53.5	77.3	76.8	58.0
Po+Trip	22.5	43.7	71.9	64.3	50.6
Logit	30.7	68.8	79.0	81.6	65.0
Proposed	**34.8**	**72.4**	**81.8**	**82.7**	**67.9**

Table 4. Targeted transfer success rate (%) ($\epsilon = 4/8/16$) of TTP vs. iterative attacks, averaged over 10 targets. The source model is Res50.

Type	Attack	→Inc-v3	→Dense121	→VGG16
Iter.	CE	0.2/1.4/7.3	7.6/26.7/48.1	10.5/23.2/37.4
	Po+Trip	**0.3**/3.0/9.6	8.3/32.9/57.3	9.9/26.9/42.5
	Logit	0.2/2.8/11.6	12.6/48.5/74.6	16.2/46.9/70.5
	Proposed	**0.3**/3.4/13.5	**15.0**/**49.7**/75.5	**18.5**/**49.7**/73.6
Gen.	TTP	0.1/**5.7**/**39.8**	1.3/38.6/**79.5**	3.6/44.2/**75.4**

5. CONCLUSION

This paper proposes a novel method for improving the transferability of targeted attacks. By analyzing the transfer failure cases, we have two critical findings: 1) the original label has a good chance of being restored when a target model predicts an AE; 2) high-confidence labels in the source model are likely to retain high confidence in the target model. Such observations motivate us to suppress the confidence of the original label and the high-confidence labels in generating AEs. We have validated the superiority of the proposed method over the state-of-the-art methods in various transfer scenarios. Our findings indicate that targeted transferability has its uniqueness and is not a simple extension of untargeted transferability. Awareness of this may help better identify the gap between targeted and untargeted transferability.

Table 1. Targeted transfer success rate (%) in the single-model, random-target scenario. Best results are in **bold**.

Attack	Source Model: Res50			Source Model: Dense121			Source Model: VGG16			Source Model: Inc-v3		
	→Inc-v3	→Dense121	→VGG16	→Inc-v3	→Res50	→VGG16	→Inc-v3	→Res50	→Dense121	→Res50	→Dense121	→VGG16
CE	3.9	44.9	30.5	2.3	19.0	11.3	0.0	0.3	0.5	1.8	2.1	1.5
Po+Trip	7.1	57.5	36.3	2.5	15.2	9.2	0.1	0.6	0.6	1.7	3.3	1.6
Logit	9.1	70.0	61.9	7.8	42.6	37.1	**0.8**	10.2	**13.6**	**2.4**	3.6	**2.2**
Proposed	**9.6**	**74.9**	**63.5**	**8.7**	**48.1**	**40.5**	**0.8**	**11.2**	**13.6**	2.3	**4.5**	**2.2**

Table 2. Targeted transfer success rate (%) in the single-model, most difficult-target scenario. Best results are in **bold**.

Attack	Source Model: Res50			Source Model: Dense121			Source Model: VGG16			Source Model: Inc-v3		
	→Inc-v3	→Dense121	→VGG16	→Inc-v3	→Res50	→VGG16	→Inc-v3	→Res50	→Dense121	→Res50	→Dense121	→VGG16
CE	1.3	25.8	15.0	1.2	6.5	3.6	0.0	0.0	0.0	1.8	4.2	2.3
Po+Trip	2.8	40.5	20.5	0.9	6.1	2.5	0.0	0.1	0.0	2.4	4.1	2.7
Logit	3.6	51.6	38.6	3.5	22.7	18.3	**0.3**	2.8	**7.0**	3.8	**5.5**	3.2
Proposed	**4.0**	**54.5**	**41.6**	**4.0**	**24.5**	**21.2**	0.1	**3.9**	6.8	**4.0**	4.9	**3.4**

6. REFERENCES

[1] C. Szegedy, W. Zaremba, I. Sutskever, et al., "Intriguing properties of neural networks," Proceedings of Int. Conf. Learning Representations 2014, arXiv: 1312.6199.

[2] C. Xie, Z. Zhang, Y. Zhou, et al., "Improving transferability of adversarial examples with input diversity," 2019 IEEE/CVF Conf. Computer Vision and Pattern Recognition, pp. 2725–2734.

[3] Y. Dong, T. Pang, H. Su, et al., "Evading defenses to transferable adversarial examples by translation-invariant attacks," 2019 IEEE/CVF Conf. Computer Vision and Pattern Recognition, pp. 4307–4316.

[4] J. Lin, C. Song, K. He, et al., "Nesterov accelerated gradient and scale invariance for adversarial attacks," In ICLR2020, arXiv: 1908.06281

[5] Y. Dong, F. Liao, T. Pang, et al., "Boosting adversarial attacks with momentum," 2018 IEEE/CVF conf. Computer Vision and Pattern Recognition, pp. 9185–9193.

[6] C. Wan, B. Ye, F. Huang, "PID-based approach to adversarial attacks," the 35th AAAI Conference on Artificial Intelligence, 2021, pp. 10033–10040

[7] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," International Conference on Learning Representations, 2017.

[8] N. Inkawhich, W. Wen, H. H. Li, and Y. Chen, "Feature space perturbations yield more transferable adversarial examples," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7059–7067.

[9] M. Li, C. Deng, T. Li, et al. "Towards transferable targeted attack," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 638–646.

[10] Z. Zhao, Z. Liu, and M. Larson, "On success and simplicity: a second look at transferable targeted attacks," In NeurIPS, 2020.

[11] N. Inkawhich, K. J. Liang, L. Carin, and Y. Chen, "Transferable perturbations of deep feature distributions," International Conference on Learning Representations, 2020.

[12] M. Naseer, S. Khan, M. Hayat, et al., "On generating transferable targeted perturbations," Proceedings of IEEE International Conference on Computer Vision, 2021, pp. 7688–7697.

[13] A. Kurakin, I. Goodfellow, S. Bengio, "Adversarial examples in the physical world," Proceedings of Int. Conf. Learning Representations, 2016. arXiv: 1607.02533.

[14] X. Yuan, P. He, Q. Zhu, et al., "Adversarial examples: Attacks and defenses for deep learning," IEEE Transactions on Neural Networks and Learning Systems, 30(9): 2805–2824, 2019.

[15] https://github.com/cleverhans-lab/cleverhans/tree/master/cleverhans_v3.1.0/examples/nips17_adversarial_competition/dataset

[16] K. He, X. Zhang, S. Ren, et al., "Deep residual learning for image recognition," 2016 IEEE Conf. Computer Vision and Pattern Recognition, pp. 770–778.

[17] C. Szegedy, V. Vanhoucke, S. Ioffe, et al., "Rethinking the inception architecture for computer vision," 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826.

[18] G. Huang, Z. Liu, V. Laurens, and K. Q. Weinberger, "Densely connected convolutional networks," 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2261–2269.

[19] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition," International Conference on Learning Representations, 2015.