Boosting Adversarial Transferability via Residual Perturbation Attack

Jinjia Peng[1,†] Zeze Tao[1,†] Huibing Wang[2,*] Meng Wang[3] Yang Wang[3,*]
[1]School of Cyber Security and Computer, Hebei University
[2]College of Information and Science Technology, Dalian Maritime University
[3]School of Computer and Information Engineering, Hefei University of Technology
{pengjinjia, zeze}@hbu.edu.cn, huibing.wang@dlmu.edu.cn, eric.mengwang@gmail.com
yangwang@hfut.edu.cn

Abstract

Deep neural networks are susceptible to adversarial examples while suffering from incorrect predictions via imperceptible perturbations. Transfer-based attacks create adversarial examples for surrogate models and transfer these examples to target models under black-box scenarios. Recent studies reveal that adversarial examples in flat loss landscapes exhibit superior transferability to alleviate overfitting on surrogate models. However, the prior arts overlook the influence of perturbation directions, resulting in limited transferability. In this paper, we propose a novel attack method, named Residual Perturbation Attack (ResPA), relying on the residual gradient as the perturbation direction to guide the adversarial examples toward the flat regions of the loss function. Specifically, ResPA conducts an exponential moving average on the input gradients to obtain the first moment as the reference gradient, which encompasses the direction of historical gradients. Instead of heavily relying on the local flatness that stems from the current gradients as the perturbation direction, ResPA further considers the residual between the current gradient and the reference gradient to capture the changes in the global perturbation direction. The experimental results demonstrate the better transferability of ResPA than the existing typical transfer-based attack methods, while the transferability can be further improved by combining ResPA with the current input transformation methods. The code is available at https://github.com/ZezeTao/ResPA.

1. Introduction

Deep neural networks (DNNs) have exhibited outstanding capabilities in various language and vision processing applications. However, adversarial examples [4, 11, 17] have been demonstrated to be indiscernible from natural ones, but can mislead a model to generate incorrect predictions. Additionally, adversarial examples generated from the surrogate model can also transfer to other target models. The transferability of adversarial examples renders adversarial attacks viable in real-world scenarios [16, 29, 30, 39, 41].

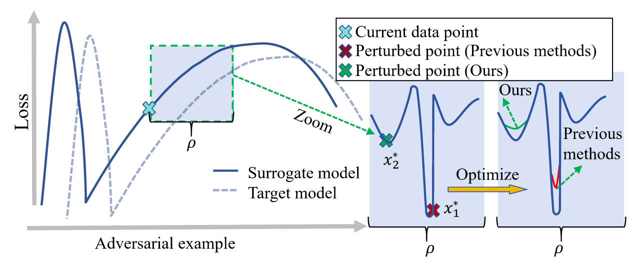


Figure 1. Comparison between ResPA and previous methods in searching for the perturbed point. ρ is the perturbation radius. Within ρ, extremely sharp regions often exist. Previous methods flatten the local region by optimizing the loss of the perturbed point in the sharpest areas. In contrast, ResPA optimizes the loss of the perturbed point, which is beneficial for the global situation.

Based on the adversary's level of knowledge about the target model, adversarial attacks can be classified into white-box attacks [11, 17, 27] and black-box attacks [4, 36, 37]. In the white-box scenario, adversaries possess comprehensive knowledge of the target models, encompassing their structures, parameter weights, and the training loss function. In contrast, in the black-box scenario, attackers create adversarial examples using a white-box surrogate model, which are subsequently transferred to the black-box target model. In real-world applications, DNN models are often hidden for users. Therefore, black-box attacks are generally more feasible. However, the over-parameterized DNNs with many sharp maxima are prone to trapping adversarial examples to be over-fitted on white-box surrogate models, resulting in poor adversarial transferability.

To alleviate the overfitting issue to enhance adversarial transferability, substantial techniques have been pro-

†Equal contribution. * Corresponding authors.

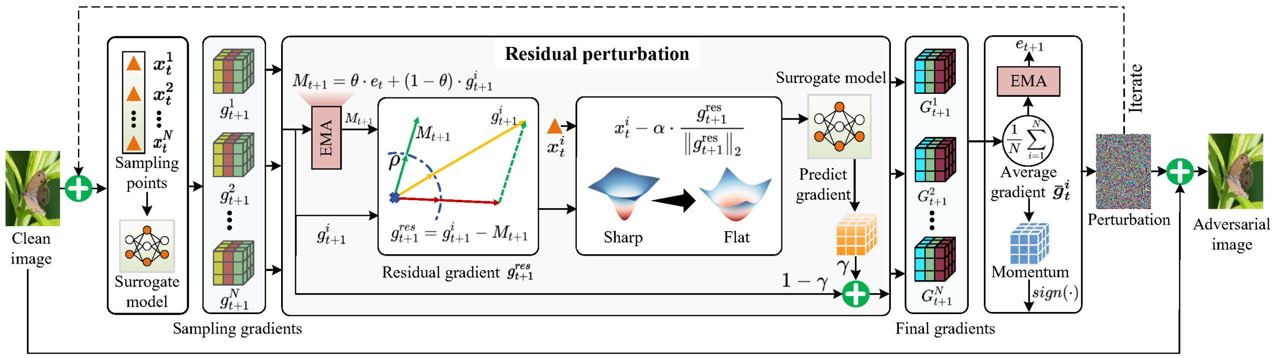


Figure 2. An overview of the proposed ResPA attack. In the process of searching for flat regions, ResPA adopts the Exponential Moving Average to perform weighted averaging on the historical records of gradients and achieve the reference gradient. Then, ResPA generates the residual gradient as the perturbation direction, defined as the difference between the reference gradient and the current gradient.

posed, including gradient-based approaches [4, 8, 26, 47], input transformation-based approaches [5, 19, 25, 36, 38], and ensemble-based methods [1, 2]. In particular, recent flatness-based methods [7, 10, 31, 40, 42] achieve state-of-the-art transferability performance by flatter maximum. These studies can flatten the sharp regions of the loss landscape, thereby mitigating overfitting on the surrogate model and ultimately enhancing transferability of adversarial samples. However, we observe that optimizing the loss of the perturbed point in excessively sharp regions with the perturbation radius may not enhance adversarial transferability. As shown in Fig. 1, when excessively sharp regions exist with the perturbation radius, existing methods utilize the gradient as the perturbation direction to identify the perturbed point, which locates the perturbed point x_1^* in the sharpest regions. Even after optimizing the flatness according to x_1^*, the point remains highly sharp, thereby failing to enhance the flatness of the entire loss surface effectively. Therefore, we turn to the perturbed point that deviates from the sharpest directions.

Based on the above, we propose a novel method, named Residual Perturbation Attack (ResPA), which adopts the residual gradient as the perturbation direction to search the perturbed point. Specifically, to prevent searching for the perturbed point in an overly sharp region, ResPA adopts the Exponential Moving Average (EMA) to perform weighted averaging on the historical gradients, thereby integrating the direction of the global gradients as the reference gradient. To better capture the changes in the global perturbation direction, ResPA considers the residual gradient as the perturbation direction, defined as the difference between the current gradient and the reference gradient. In addition, it can capture the variations in the global perturbation direction, which could avoid excessive reliance on the flatness of the minimal perturbed point x_1^*. Instead of the perturbed point in the sharpest regions, ResPA identifies the perturbed point that are beneficial to the flatness of the entire loss surface. In summary, our contributions are summarized below:

- To the best of our knowledge, we are the first to reveal that optimizing the loss of the perturbed point in overly sharp regions significantly hinders the transferability of flatness-based attack methods.
- We propose a novel Residual Perturbation Attack method (ResPA), which employs the residual gradient as the perturbation direction to evaluate the flatness of local points. ResPA could capture the changes in the global perturbation direction, thereby avoiding searching for the perturbed point in the sharpest regions.
- ResPA incorporates the proposed flatness term as a regularization, to maximize the loss function to flatten the loss surface.

Experimental results demonstrate the better transferability of ResPA than the typical transfer-based attack methods. In addition, combining ResPA with the current input transformation method can further improve the transferability.

2. Methodology

2.1. Preliminary

Let x represent the input image with y as its corresponding true label. The classifier's output of the surrogate model[1] is denoted as $f^s(x)$, and $J(x, y) = -\sum_{k=1}^{C} y_k \log f^s(x)$ is the cross-entropy loss function, where C is the number of categories, while $y \in \{0, 1\}^C$ is a one-hot encoded vector such that $y_k = 1$ if x belongs to the k-th class, and $y_k = 0$ otherwise. Given x, the goal of adversarial attacks is to identify an adversarial example x^{adv} that deceives the classifier. Specifically, this adversarial example should prompt the classifier to output a label that differs from the true label, i.e., $f^s(x^{adv}) \neq f^s(x)$, while remaining impercepti-

[1]We adopt the varied deep models, e.g., Convolutional Neural Networks (CNNs), Vision Transformer (ViT) etc.

ble to human observers. This imperceptibility is ensured by following the constraint $\left\|x - x^{adv}\right\|_p < \epsilon$, where $\|\cdot\|_p$ denotes the L_p norm with $\epsilon > 0$ to define the perturbation magnitude. In this study, we concentrate on the L_p norm for consistency with prior research. Our goal is to maximize the subsequent optimization problem to generate the adversarial samples:

$$\max_{x^{adv}} J\left(x^{adv}, y\right) \quad \text{s.t. } \left\|x - x^{adv}\right\|_\infty < \epsilon. \tag{1}$$

Optimizing Eq. (1) requires calculating the gradient of the loss function, but this is not feasible in the black-box setting. Consequently, we generate the transferable adversarial examples on a surrogate model that can be used to attack other target models. The adversarial examples are encouraged to hoax target models to output wrong predictions, and the transferability can be quantitatively evaluated via the Attack Success Rate (ASR), calculated as follows:

$$ASR = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \mathbb{I}[f^t(x) \neq f^t(x^{adv})], \tag{2}$$

where $\mathcal{X}$ denotes all the legitimate images. $f^t(\cdot)$ is the classifier's output of the target model. $\mathbb{I}(\cdot)$ is the indicator function, such that it equals to 1 once $f^t(x) \neq f^t(x^{adv})$, and 0 otherwise.

2.2. Residual Perturbation Attack

At the core of our technique is to capture the changes in the global perturbation direction, while the over-reliance on the flatness of local points needs to be addressed. To this end, our proposed residual perturbation is elaborated upon in detail, followed by the process of searching flat maxima with our proposed residual perturbation.

2.2.1. Residual Perturbation

Previous methods [7, 10, 31, 42] craft adversarial samples upon a locally flat region of the loss landscape, thereby elevating the adversarial transferability to unprecedented heights. Within the given perturbation radius ρ, these methods measure **the flatness of the current point** by the difference between the loss of the perturbed point and the loss of the current point. However, in practical applications, there often exist excessively sharp regions within the perturbation radius, which weakens the effectiveness of the flatness term due to "**excessively sharp regions**". Since the existing methods [7, 10, 31, 42] use the current gradient as the perturbation direction to search for the perturbed point, which is more likely to locate the perturbed point in the sharpest regions, as illustrated in Fig. 1. However, as can be seen from Fig. 1, optimizing the loss of the perturbed point x_1^* in the sharpest region does not effectively flatten the loss surface. Consequently, existing methods heavily rely on the flatness of the perturbed point in the local sharpest region, which is detrimental to the flatness of subsequent data points, thereby failing to achieve optimal transferability. To this end, we propose the Residual Perturbation Attack (ResPA) to optimize the loss of the perturbed point that are beneficial to global flatness, as shown in Fig. 2. ResPA primarily consists of two components:

1. Residual Perturbation: We define **a novel flatness term** using residual gradients, which addresses the "**excessively sharp regions within the perturbation radius.**"
2. Flat Maxima with Residual Perturbation: We incorporate **the proposed flatness term as a regularization term** into the maximization over the loss function to achieve a flat surface.

In the t-th iteration, given the adversarial sample x_t^{adv}, the flatness of the loss function $\bar{J}$ at x_t^{adv} is defined as:

$$\bar{J}\left(x_t^{adv}, y\right) = \underbrace{\left[\min_{\|\delta\|_1 \leq \rho} J\left(x_t^{adv} - \delta, y\right) - J\left(x_t^{adv}, y\right)\right]}_{\text{flatness}}, \tag{3}$$

where δ is the perturbation vector with the same dimension as x_t^{adv}. Since it is difficult to track the exact minimal neighbor, the existing methods [9, 11] utilize the gradient of the neighbor in the descent direction for iteration after two approximations:

$$\min_{\|\delta\|_1 \leq \rho} J(x_t^{adv} - \delta, y) \approx J\left(x_t^{adv} - \rho\frac{d_t}{\|d_t\|}, y\right), \tag{4}$$

where ρ denotes the radius to control the neighborhood size; d_t denotes the gradient $\nabla_{x_t^{adv}} J(x_t^{adv}, y)$ at the current point, which determines the perturbation direction. Eq. (4) encodes the loss of the perturbed point in the sharpest region. Since Eq. (4) relies solely on the current gradient direction, it fails to capture the variations in the global perturbation direction. As a result, it is not advantageous for subsequent data samples, which limits their transferability.

To address this limitation, we propose to exploit additional gradient information to search for the global perturbation direction. ResPA initially applies an exponential moving average (EMA) to the current gradient $\nabla_{x_t^{adv}} J(x_t^{adv}, y)$ to derive the first moment as the reference gradient. With increasing iterations, the reference gradient integrates the direction of all historical gradients, which is formulated as:

$$M_{t+1} = \theta \cdot e_t + (1 - \theta) \cdot \nabla_{x_t^{adv}} J(x_t^{adv}, y), \tag{5}$$

where $\theta \in (0, 1)$ is the exponential decay factor; e_t is calculated as the EMA of the previous average gradient according to Eq. (11) and $e_0 = 0$. To mitigate the excessive reliance on the perturbed point in the sharpest region, ResPA utilizes the residual gradient g_{t+1}^{res} as the perturbation direction. This enables to capture the actual variations between the current and historical gradient direction. g_{t+1}^{res} is formulated as follows:

$$g_{t+1}^{res} = \nabla_{x_t^{adv}} J(x_t^{adv}, y) - M_{t+1}. \tag{6}$$

Finally, we utilize g_{t+1}^{res} as the perturbation direction to locate the perturbed point, while **the proposed flatness term** can be obtained as follows:

$$\bar{J}\left(x_t^{adv}, y\right) = J\left(x_t^{adv} - \rho \frac{g_t^{res}}{\|g_t^{res}\|}, y\right) - J(x_t^{adv}, y). \tag{7}$$

2.2.2. Flat Maxima with Residual Perturbation

Why can ResPA avoid searching for the perturbed point in excessively sharp regions? In flat areas, the curvature of the loss surface is small, so that the current gradient $\nabla_{x_t^{adv}} J(x_t^{adv}, y)$ is relatively low. In this case, as shown in Eq. (5), the reference gradient M_{t+1} is also small, and consequently, according to Eq. (6), the residual gradient g_{t+1}^{res} is similarly small. Therefore, in flat regions, neither previous methods nor our proposed ResPA will search for the perturbed point in overly sharp areas.

In sharp areas, the curvature of the loss surface is large, and the current gradient direction undergoes significant changes. Therefore, using the current gradient as the perturbation direction may be result in searching for the perturbed point in the steepest regions. In contrast, in ResPA, the residual gradient g_{t+1}^{res} is defined as the difference between the current gradient $\nabla_{x_t^{adv}} J(x_t^{adv}, y)$ and the reference gradient M_{t+1}. M_{t+1} represents the average of historical gradients, which does not change significantly with abrupt variations in the current gradient. Therefore, when the current gradient $\nabla_{x_t^{adv}} J(x_t^{adv}, y)$ exhibits large fluctuations, the residual gradient g_{t+1}^{res} can effectively suppress the current gradient to some extent, thereby avoiding searching for perturbation points in the sharpest regions.

In summary, when excessively sharp regions exist within the perturbation radius, ResPA avoids searching for the perturbed point in these overly sharp areas. Instead, it leverages the difference between the current gradient and historical gradients to identify the perturbed point that is more beneficial to the overall loss surface.

To incorporate the flatness term $\bar{J}\left(x_t^{adv}, y\right)$ into the optimization problem to improve the transferability of adversarial examples, ResPA employs **the flatness term as a regularization term** to impose constraints on the initial loss function. The primary goal is to jointly maximize the loss function and the flatness term. Following this strategy, ResPA could guide adversarial examples to flat regions.

Let x_t^{adv} denote the input at the t-th iteration. $x_t^i = x_t^{adv} + \lambda_t^i$ is defined to be sampled within the neighborhood of x_t^{adv}, where $i = 1, 2, \cdots, N$, such that N represents the sample number. Here $\lambda_t^i \sim U\left[-(\beta \cdot \varepsilon)^d, (\beta \cdot \varepsilon)^d\right]$, where $U\left[a^d, b^d\right]$ stands for the uniform distribution in d dimensions. The optimization problem of $J\left(x_t^i, y\right)$ is modified with the flatness term $\bar{J}\left(x_t^{adv}, y\right)$ in the t-th iteration as:

$$\begin{aligned}\mathcal{L}\left(x_t^i, y\right) &= J\left(x_t^i, y\right) + \gamma \cdot \bar{J}\left(x_t^i, y\right) \\ &= J\left(x_t^i, y\right) + \gamma \cdot \left[J\left(x_t^*, y\right) - J\left(x_t^i, y\right)\right] \\ &= (1-\gamma) \cdot J\left(x_t^i, y\right) + \gamma \cdot J\left(x_t^*, y\right),\end{aligned} \tag{8}$$

where $x_t^* = x_t^i - \rho \frac{g_t^{\text{res}}}{\|g_t^{\text{res}}\|}$ is the perturbed sample, $\gamma \in [0, 1]$ is the penalty coefficient, and the regularization term is the flatness term, which can flatten the loss surface.

We discuss more about Eq. (8), In particular:

1. when $\gamma = 0$, the optimization problem is equivalent to solely maximizing the initial loss $J\left(x_t^i, y\right)$;
2. when $\gamma = 1$, it aims at solely maximizing the perturbed point loss $J\left(x_t^*, y\right)$;
3. when $\gamma \in (0, 1)$, the optimization problem simultaneously balances both the initial loss and the perturbed point loss.

The gradient of the current loss function can be formulated as follows:

$$\nabla_{x_t^i} \mathcal{L}\left(x_t^i, y\right) \approx (1-\gamma) \cdot \nabla_{x_t^i} J\left(x_t^i, y\right) + \gamma \cdot \nabla_{x_t^i} J\left(x_t^*, y\right). \tag{9}$$

Next, ResPA acquires the average gradient $\bar{g}_{t+1}$ over N sampling points to be formulated as:

$$\bar{g}_{t+1} = \frac{1}{N} \sum_{i=1}^{N} \nabla_{x_t^i} \mathcal{L}\left(x_t^i, y\right), \tag{10}$$

where N represents the number of sampling points. Next, we compute the EMA of the average gradient $\bar{g}_{t+1}$ to obtain the first-order moment e_{t+1}, which will be substituted into Eq. (5) in the next iteration to update the reference gradient M_{t+1}. The update rule for e_{t+1} is given by:

$$e_{t+1} = \theta \cdot e_t + (1-\theta) \cdot \bar{g}_{t+1}, \tag{11}$$

where θ is the exponential decay factor. Then the average gradient $\bar{g}_{t+1}$ is employed to update the momentum g_{t+1}, yielding:

$$g_{t+1} = \mu \cdot g_t + \frac{\bar{g}_{t+1}}{\left\|\bar{g}_{t+1}\right\|_1}, \tag{12}$$

where μ is the decay factor. Finally, the adversarial samples x_{t+1}^{adv} are updated as follows:

$$x_{t+1}^{adv} = Clip_x^{\epsilon}(x_t^{adv} + \alpha \cdot \text{sign}\left(g_{t+1}\right)), \tag{13}$$

where $\text{sign}(\cdot)$ is the sign function, $Clip_x^{\epsilon}(\cdot)$ denotes that the generated adversarial image is constrained within the ϵ-ball neighborhood of the original image x, and α denotes the predetermined step size.

Model	Attack	Inc-v3	Res-50	Vgg-19	Den-121	ViT	Swin	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	Average
Inc-v3	MI [4]	**100.0***	47.5	59.5	49.0	33.5	24.8	22.9	22.4	45.0
	VMI [37]	**100.0***	65.5	70.1	66.9	44.6	40.4	39.1	39.1	58.2
	GRA [47]	**100.0***	67.3	72.1	68.5	45.7	42.0	40.3	41.2	59.6
	PGN [10]	**100.0***	73.4	76.8	74.5	52.7	44.8	42.8	43.5	63.6
	AdaMSI [24]	**100.0***	56.3	69.6	54.2	35.6	27.2	14.6	15.6	46.6
	TPA [7]	98.1*	68.2	70.7	68.2	46.8	43.3	**44.6**	42.2	60.3
	ResPA (Ours)	**100.0***	**75.7**	**79.6**	**74.8**	**54.0**	**45.8**	42.3	**44.3**	**64.6**
Res-50	MI [4]	65.8	**100.0***	82.0	91.7	49.5	45.1	43.1	42.1	64.9
	VMI [37]	81.9	99.9*	91.6	96.1	67.0	63.5	65.2	64.5	78.7
	GRA [47]	87.0	99.9*	94.4	**98.1**	72.8	68.3	72.7	70.3	82.9
	PGN [10]	88.1	**100.0***	95.2	98.0	75.0	70.0	74.6	72.4	84.2
	AdaMSI [24]	65.8	**100.0***	89.4	91.2	48.2	43.6	36.6	36.8	64.0
	TPA [7]	85.2	98.7*	92.6	94.6	70.8	68.1	72.5	70.9	81.7
	ResPA (Ours)	**88.8**	**100.0***	**95.5**	98.0	**75.5**	**71.3**	**75.2**	**72.6**	**84.6**
Den-121	MI [4]	69.7	89.9	83.5	**100.0***	55.1	51.4	49.1	49.5	68.5
	VMI [37]	84.5	96.2	91.3	**100.0***	73.7	68.7	70.2	70.7	81.9
	GRA [47]	88.4	97.8	93.2	**100.0***	78.6	75.3	78.9	76.0	86.0
	PGN [10]	89.5	97.5	94.9	**100.0***	80.9	**77.0**	80.0	77.7	87.2
	AdaMSI [24]	76.2	93.8	92.7	**100.0***	62.3	52.0	46.7	44.2	71.0
	TPA [7]	89.7	96.6	94.1	99.3*	79.4	75.3	79.3	76.8	86.3
	ResPA (Ours)	**90.2**	**97.9**	**94.9**	**100.0***	**82.1**	**77.0**	**80.3**	**77.7**	**87.5**

Table 1. The attack success rates (%) on eight models by a single attack. The adversarial examples are generated on Inc-v3, Res-50, and Den-121 separately. Here * indicates the white-box model. The best results are bold.

3. Experiments

3.1. Experimental Setup

Dataset. We follow the convention of utilizing 1,000 origin images from the ILSVRC 2012 validation set [32] to evaluate the performance of ResPA, mirroring the methodologies adopted in prior research [37, 38]. The models involved in this paper can classify these clean images with an accuracy of nearly 100%. We also validate the effectiveness of the proposed method in the high-security-demand application scenario of person re-identification, with experiments conducted on Market-1501 [46].

Models. We evaluate the attack success rate on 6 widely-used pre-trained models, namely Inception-v3 (Inc-v3) [34], ResNet-50 (Res-50) [14], DenseNet-121 (Den-121) [15], VGGNet-19 (Vgg-19) [33], Vision Transformer (ViT) [6], and Swin Transformer (Swin) [21] to validate the effectiveness of ResPA. Besides that, we consider adversarially trained models [35], specifically ens3-adv-Inception-v3 (Inc-v3$_{ens3}$) and ens4-adv-Inception-v3 (Inc-v4$_{ens4}$). Furthermore, seven state-of-the-art defense models are integrated, which exhibit exceptional robustness when defending against black-box attacks targeting the ImageNet dataset. The defense techniques encompass the high-level representation guided denoiser (HGD) [18], bit depth reduction (Bit-Red) [45], feature distillation (FD) [20], JPEG compression (JPEG) [12], neural representation purifier (NRP) [28], random resizing and padding (R&P) [43], as well as randomized smoothing (RS) [3].

Baseline Methods. In our experiments, six of the latest transfer-based attacks, namely MI [4], VMI [37], GRA [47], PGN [10], AdaMSI [24], and TPA [7], are taken into consideration. These attacks have exhibited superior performance in terms of success rates when benchmarked against earlier techniques such as FGSM [11] and I-FGSM [17]. Furthermore, we integrate the proposed ResPA with a variety of input transformations to affirm its effectiveness, such as DIM [44], TIM [5], SIM [19], Admix [38], and SSA [25].

Evaluation Metric. In the experiment, we employ the attack success rate [11] in accordance with Eq. (2) as the evaluation metric, which refers to the proportion of adversarial examples (among all generated ones) that can successfully mislead the target model.

Parameter Setting. We set the maximum perturbation of the parameter $\epsilon = 16$, the step size $\alpha = 1.6$, and the number of iterations $T = 10$. The decay factor $\mu = 1$ is set for all the approaches. To ensure a fair comparison in this paper, we have adopted a uniform configuration for VMI, GRA, PGN, and TPA, specifying the number of sampled examples as $N = 5$ and defining the upper limit of the neighborhood size as $\beta = 1.5 \times \epsilon$. For DIM, we set the transformation probability at 0.5. Regarding TIM, a Gaussian kernel of size 7×7 is employed as in [5]. In the case of SIM, the number of scale copies is set to $m = 5$. For Admix, we set the mixing ratio to 0.2 and the number of copies to 5. For the proposed ResPA, we set the number of examples $N = 5$, the exponential decay factor $\theta = 0.6$, the balanced coefficient $\gamma = 0.6$, and the upper bound of $\beta = 1.5 \times \epsilon$.

3.2. Evaluation on Single Model

In this section, we carry out multiple attacks. The adversarial samples are crafted from three distinct models, namely

Attack	Inc-v3	Res-50*	Vgg-19	Den-121	ViT	Swin	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	Average
DIM [44]	85.9	**100.0**	93.2	96.9	67.3	61.8	67.9	63.4	79.6
DIM+Ours	**94.3**	**100.0**	**97.4**	**98.7**	**87.6**	**75.9**	**88.1**	**85.9**	**91.0**
TIM [5]	71.1	**100.0**	84.5	92.7	60.5	49.4	55.3	51.5	70.6
TIM+Ours	**94.8**	**100.0**	**96.8**	**98.9**	**87.8**	**74.2**	**87.7**	**87.2**	**90.9**
SIM [19]	84.2	**100.0**	89.7	97.7	63.4	57.2	65.5	60.8	77.3
SIM+Ours	**92.4**	**100.0**	**97.0**	**98.7**	**82.0**	**76.2**	**83.5**	**81.0**	**89.0**
Admix [38]	74.6	96.2	87.7	92.7	55.7	52.3	52.3	50.0	70.2
Admix+Ours	**85.9**	**96.9**	**92.4**	**93.1**	**73.9**	**69.7**	**72.9**	**69.3**	**81.8**
SSA [25]	88.2	**100.0**	95.8	97.5	68.2	67.5	72.9	69.0	82.4
SSA+Ours	**91.0**	**100.0**	**97.2**	**98.2**	**79.0**	**74.9**	**78.0**	**74.8**	**86.6**

Table 2. The attack success rates (%) of our method, when it is integrated with DIM, TIM, SIM, Admix, and SSA, respectively. The adversarial examples are generated on Res-50. Here * indicates the white-box model. The best results are bold.

Attack	Inc-v3	Res-50*	Vgg-19*	Den-121*	ViT	Swin	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	Average
MI [4]	87.7	99.9	99.9	99.9	72.3	75.6	69.7	68.5	84.2
VMI [37]	93.9	**100.0**	**100.0**	**100.0**	85.6	86.8	85.6	82.4	91.8
GRA [47]	97.5	**100.0**	**100.0**	**100.0**	92.9	91.9	90.8	90.1	95.4
PGN [10]	**97.6**	**100.0**	**100.0**	**100.0**	93.3	92.6	**92.4**	90.1	95.8
AdaMSI [24]	91.9	**100.0**	**100.0**	**100.0**	76.3	74.6	65.1	59.6	83.4
TPA [7]	95.3	98.7	98.7	98.8	89.7	90.6	90.1	88.3	93.8
ResPA (Ours)	**97.6**	**100.0**	**100.0**	**100.0**	**94.0**	**93.2**	92.3	**90.5**	**96.0**

Table 3. The attack success rates (%) on eight models under ensemble model setting. The adversarial examples are generated on Res-50, Vgg-19 and Den-121 models. Here * indicates the white-box model. The best results are bold.

Inc-v3, Res-50, and Den-121, respectively. The results are summarized in Tab. 1. The models we attack are arranged in rows, and the eight models we test are arranged in columns. From the results, it can be seen that the ResPA method proposed in this paper not only maintains a high attack performance in a white-box setting but also significantly improves the attack performance in a black-box setting. For instance, when generating adversarial examples on Inc-v3, the state-of-the-art methods, namely GRA [47], PGN [10], and TAP [7], achieve average attack success rates of 59.6%, 63.6%, and 60.3% on eight models, respectively. By contrast, ResPA attains an impressive average attack success rate of 64.6%, outperforming them by 5.0%, 1.0%, and 4.3% respectively. Excellent results highlight that using the residual gradient as the perturbation direction can further enhance the attack performance of adversarial samples.

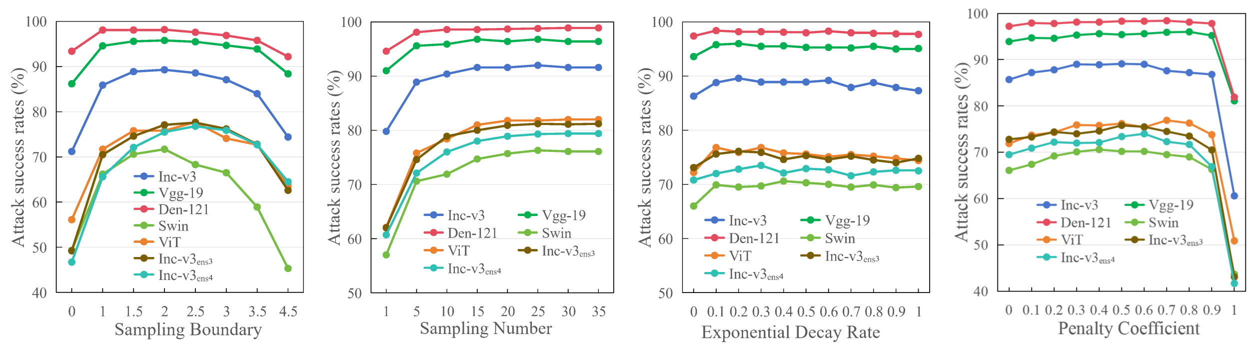
3.3. Evaluation on Combined Input Transformation

The attack success rates of combination approaches are also evaluated. Given the streamlined and effective gradient updating mechanism, ResPA can seamlessly integrate with input augmentation approaches to further enhance adversarial transferability. We incorporate ResPA into five input augmentation attacks, namely DIM [44], TIM [5], SIM [19], Admix [38], and SSA [25]. All the combined methods create adversarial samples on Res-50, and the performances are presented in Tab. 2. As can be seen from the table, the combinational attacks exhibit clear improvements over all baseline attacks. For instance, ResPA increases the average attack success rate of the five baseline attacks by 11.4%, 20.3%, 11.7%, 11.6%, and 4.2%, respectively, which validates that ResPA can significantly enhance transferability.

In particular, after integrating these input transformation-based methods, ResPA is more likely to attain significantly superior outcomes on adversarially trained ensemble models in comparison with the results presented in Tab. 1. The adversarial samples created by the proposed ResPA method are situated within broader and smoother flat local maxima, which validates that the proposed method has the ability to generate adversarial examples located at the flat maximum.

3.4. Evaluation on Ensemble Model

We also evaluate the performance of ResPA in an ensemble-model setting. The ensemble attack methodology described in [4] is adopted, constructing an ensemble by averaging the logit outputs from a diverse set of models. Specifically, adversarial examples are crafted by integrating the predictions from three conventionally trained models: Res-50, Vgg-19, and Den-121. Equal weights are assigned to all the ensemble models. Subsequently, we evaluate the transferability of standardly trained models and adversarially trained models and present the relevant results in Sec. 3.1. As can be seen from the table, compared with previous attacks, ResPA achieves the best performance. Importantly, when aimed at transformer-based models, ResPA constantly surpasses other transfer-based attacks. Furthermore, in the white-box setting, the proposed ResPA can still retain success rates similar to those of the baselines.



(a) The hyper-parameter β (b) The hyper-parameter N (c) The hyper-parameter θ (d) The hyper-parameter γ

Figure 3. The attack success rate (%) on seven black-box models with different hyper-parameters β, N, θ and γ. The adversarial examples are generated by ResPA on Res-50.

Attack	HGD [18]	Bit-Red [45]	FD [20]	JPEG [13]	NRP [28]	R&P [43]	RS [3]	Average
MI [4]	40.2	33.2	44.2	36.2	32.6	42.2	28.7	36.8
VMI [37]	60.0	51.2	61.8	56.0	42.3	59.7	33.0	52.0
GRA [47]	64.0	54.1	64.9	59.2	43.2	62.4	35.8	54.8
PGN [10]	68.5	59.8	69.0	64.8	45.8	66.9	36.9	58.8
AdaMSI [24]	38.9	34.0	43.5	37.3	22.4	41.9	29.9	35.4
TPA [7]	63.8	58.8	66.9	61.0	44.9	62.8	37.4	56.5
ResPA (Ours)	**69.2**	**60.7**	**69.9**	**65.2**	**48.0**	**67.9**	**37.7**	**59.8**

Table 4. The attack success rates (%) of seven advanced defense mechanisms on adversarial samples. The adversarial samples are generated on the Inc-v3 model by various transfer-based attacks. The best results are bold.

3.5. Evaluation on Defense Method

To evaluate the effectiveness of the proposed ResPA method, we also examine the attack success rates of ResPA against various advanced defense mechanisms. Some advanced defense methods, such as HGD, Bit-Red, FD, JPEG, NRP, R&P, and RS, are employed. Results are presented in Tab. 4. In the black-box setting, it is observed that ResPA outperforms other state-of-the-art attack algorithms. For example, GRA [47], TPA [7], and PGN [10] attain average success rates of 54.8%, 56.5%, and 58.8%, respectively, when tested against the seven defense models. In comparison, the proposed ResPA approach attains an average success rate of 59.8%, outperforming them by 4.8%, 3.3%, and 1.0%, respectively. This significant progress fully demonstrates the outstanding effectiveness of ResPA when dealing with adversarially trained models and other defense models.

3.6. Attacking Person Re-identification

We also conduct comparative experiments on the person re-identification (Re-ID) benchmark dataset [46]. To successfully attack the Re-ID system, we use predicted labels instead of ground truth labels. The queries of the Re-ID system are attacked as adversarial queries, targeting different backbone networks of the Re-ID model, including DenseNet-121 (Den-121) [15], ConvNext (Conv) [23], Swin-Transformer (Swin) [21], and Swinv2-Transformer (Swinv2) [22]. The evaluation metrics are Rank-1 and mAP, where lower values indicate better attack performance. As shown in Tab. 5, the results demonstrate that the proposed ResPA achieves the best attack performance compared to state-of-the-art methods.

3.7. Experiments on Hyper-parameters

Various experiments are conducted about the hyper-parameters of ResPA, namely the sampling boundary β, the sampling number N, the exponential decay rate θ, and the penalty coefficient γ. For the sake of simplified analysis, all adversarial examples are generated based on the Res-50 model. By default, we set $\beta = 1.5 \times \epsilon$, $N = 5$, $\theta = 0.4$, and $\gamma = 0.4$.

The sampling boundary β. We analyze the influence of the sampling boundary β on the result. As shown in Fig. 3(a), as β increases, the transferability improves, and when $\beta = 1.5 \times \epsilon$, it reaches the peak for CNN-based models. However, when facing Transformer-based and adversarially trained models, the transferability still increases. When $\beta > 2.5 \times \epsilon$, the performance of adversarial transferability will decline on seven black-box models. For a fair comparison, we uniformly set $\beta = 1.5 \times \epsilon$ for the methods [7, 10, 37, 47] involving sampling.

The sampling number N. In Fig. 3(b), we explore the

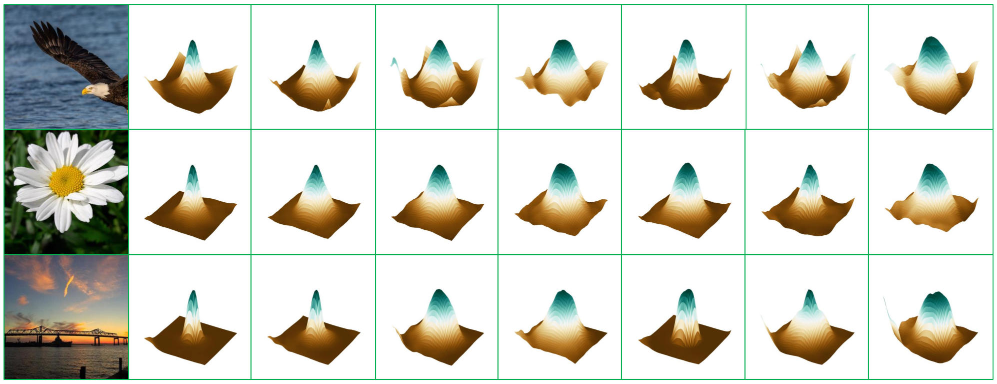


Figure 4. Visualization of loss surfaces along two random directions for three randomly sampled adversarial examples on the surrogate model (Inc-v3). Compared with other methods, ResPA can assist adversarial examples in reaching flatter maximum regions.

Method	Market-1501									
	Den-121		Conv		Swin		Swinv2		Average	
	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
Before attack	89.22	73.62	89.88	72.57	92.43	78.90	91.54	77.67	90.77	75.69
MI [4]	32.01	22.20	43.38	29.39	54.66	40.95	51.22	38.11	45.32	32.66
VMI [37]	20.37	14.70	29.25	19.61	41.51	30.63	37.20	27.41	32.08	23.09
PGN [10]	18.91	13.43	26.37	17.84	39.76	28.85	35.93	26.14	30.24	21.57
GRA [47]	21.85	15.44	31.86	21.30	43.91	32.49	39.73	29.44	28.88	20.81
BSR [36]	18.62	13.16	27.49	18.39	40.71	29.90	37.74	27.04	31.14	22.12
TPA [7]	15.47	11.30	24.26	16.09	35.78	26.12	32.60	23.30	27.03	19.20
ResPA (Ours)	**15.30**	**11.12**	**24.14**	**15.95**	**35.31**	**25.67**	**31.03**	**22.33**	**26.45**	**18.77**

Table 5. Performance (%) of adversarial attacks against the four Re-ID models under black-box setting on the Market-1501 dataset. The adversarial queries are crafted on Res-50. Lower is better for the attack.

influence of N. As the N increases, the transferability also rises. In the case of CNN-based models, when $N > 15$, the performance of adversarial transferability gradually stabilizes on 7 black-box models. Nevertheless, for Transformer-based and adversarially trained models, the transferability continues to increase. When $N > 25$, the attack success rates gradually stabilize across seven black-box models. For a fair comparison, we uniformly set $N = 5$ for the methods [7, 10, 37, 47] involving sampling.

The exponential decay rate θ. We investigate the influence of θ on the results. As shown in Fig. 3(c), as θ increases, the transferability improves. When $\theta > 0.2$, the transferability of these black-box models is almost stable. This also indicates that when θ is in the interval [0.2, 1], it has little influence on the transferability. In this paper, θ is set to 0.6.

The penalty coefficient γ. As depicted in Fig. 3(d), we examine the influence of γ. In particular, when γ lies within the interval $[0.2, 0.9]$, the transferability of these black-box models is relatively satisfactory. However, when $\gamma > 0.9$, the transferability declines sharply. In this paper, $\gamma = 0.6$.

3.8. Visualization of Loss Surfaces

To verify that ResPA can help adversarial examples find a flat maxima area, we compare the loss surface maps of adversarial examples generated by different attacking approaches on the Inc-v3 model. Each 2D graph corresponds to an adversarial sample, with the adversarial sample placed at the center. Each row in Fig. 4 represents the visualization of one image. Compared with other methods, ResPA can assist adversarial examples in reaching flatter maxima areas. The adversarial examples created by ResPA are located in broader and smoother flat areas, which validates that ResPA can create adversarial samples located in the flat maximum.

4. Conclusion

In this paper, we propose ResPA, a novel attack method to identify highly transferable adversarial examples. Instead of directly focusing on the current gradient as the perturbation direction, ResPA considers the residual between the current and historical gradient as the perturbation direction, thereby trying to avoid the over-reliance on the perturbed point in excessively sharp regions. As a byproduct, ResPA incorporates the proposed flatness term as a regularization to maximize the loss function while making the loss surface flatter. Experimental results demonstrate the better transferability of ResPA than the existing state-of-the-art transfer-based attack approaches.

5. Acknowledgement

This research is supported by National Natural Science Foundation of China (U21A20470, 62172136, 72188101); Institute of Advanced Medicine and Frontier Technology (2023IHM01080); Liaoning Provincial Natural Science Foundation (2024-MS-012); National Key Research and Development Program of china (2024YFB4710800); Dalian Science and Technology Talent Innovation Support Plan (2024RY010); Natural Science Foundation of Hebei Province (F2025201037).

References

[1] Zhaohui Che, Ali Borji, Guangtao Zhai, Suiyi Ling, Jing Li, and Patrick Le Callet. A new ensemble adversarial attack powered by long-term gradient memories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3405–3413, 2020. 2

[2] Bin Chen, Jiali Yin, Shukai Chen, Bohao Chen, and Ximeng Liu. An adaptive model ensemble adversarial attack for boosting adversarial transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4489–4498, 2023. 2

[3] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019. 5, 7

[4] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9185–9193, 2018. 1, 2, 5, 6, 7, 8

[5] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4312–4321, 2019. 2, 5, 6

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. 2021. 5

[7] Mingyuan Fan, Xiaodan Li, Cen Chen, Wenmeng Zhou, and Yaliang Li. Transferability bound theory: Exploring relationship between adversarial transferability and flatness. In *Proceedings of the Advances in Neural Information Processing Systems*, 2024. 2, 3, 5, 6, 7, 8

[8] Zhengwei Fang, Rui Wang, Tao Huang, and Liping Jing. Strong transferable adversarial attacks via ensembled asymptotically normal distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24841–24850, 2024. 2

[9] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. 3

[10] Zhijin Ge, Fanhua Shang, Hongying Liu, Yuanyuan Liu, and Xiaosen Wang. Boosting adversarial transferability by achieving flat local maxima. In *Proceedings of the Advances in Neural Information Processing Systems*, 2023. 2, 3, 5, 6, 7, 8

[11] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. 1, 3, 5

[12] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *International Conference on Learning Representations*, 2018. 5

[13] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *International Conference on Learning Representations*, 2018. 7

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[15] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5, 7

[16] Zelun Kong, Junfeng Guo, Ang Li, and Cong Liu. Physgan: Generating physical-world-resilient adversarial examples for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14254–14263, 2020. 1

[17] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 1, 5

[18] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1778–1787, 2018. 5, 7

[19] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and J. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *International Conference on Learning Representations*, 2019. 2, 5, 6

[20] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 860–868. IEEE, 2019. 5, 7

[21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 5, 7

[22] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12009–12019, 2022. 7

[23] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 7

[24] Sheng Long, Wei Tao, LI Shuohao, Jun Lei, and Jun Zhang. On the convergence of an adaptive momentum method for adversarial attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14132–14140, 2024. 5, 6, 7

[25] Yuyang Long, Qilong Zhang, Boheng Zeng, Lianli Gao, Xianglong Liu, Jian Zhang, and Jingkuan Song. Frequency domain model augmentation for adversarial attack. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 549–566. Springer, 2022. 2, 5, 6

[26] Wenshuo Ma, Yidong Li, Xiaofeng Jia, and Wei Xu. Transferable adversarial attack for both vision transformers and convolutional networks via momentum integrated gradients. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4630–4639, 2023. 2

[27] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 1

[28] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 262–271, 2020. 5, 7

[29] Jinjia Peng, Yang Wang, Huibing Wang, Zhao Zhang, Xianping Fu, and Meng Wang. Unsupervised vehicle reidentification with progressive adaptation. In *IJCAI*, 2020. 1

[30] Linfeng Qi, Huibing Wang, Jiqing Zhang, Jinjia Peng, and Yang Wang. Unsupervised domain adaptive person search via dual self-calibration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6550–6558, 2025. 1

[31] Zeyu Qin, Yanbo Fan, Yi Liu, Li Shen, Yong Zhang, Jue Wang, and Baoyuan Wu. Boosting the transferability of adversarial attacks with reverse adversarial perturbation. *Advances in neural information processing systems*, 35:29845–29858, 2022. 2, 3

[32] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 5

[33] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, 2015. 5

[34] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 5

[35] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018. 5

[36] Kunyu Wang, Xuanran He, Wenxuan Wang, and Xiaosen Wang. Boosting adversarial transferability by block shuffle and rotation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24336–24346, 2024. 1, 2, 8

[37] Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1924–1933, 2021. 1, 5, 6, 7, 8

[38] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16158–16167, 2021. 2, 5, 6

[39] Yang Wang, Jinjia Peng, Huibing Wang, and Meng Wang. Progressive learning with multi-scale attention network for cross-domain vehicle re-identification. *Science China Information Sciences*, 65(6):16103, 2022. 1

[40] Yang Wang, Biao Qian, Haipeng Liu, Yong Rui, and Meng Wang. Unpacking the gap box against data-free knowledge distillation. *IEEE Trans. Pattern Anal. Mach. Intell*, 46(9): 6280–6291, 2024. 2

[41] Lin Wu, Yang Wang, Hongzhi Yin, Meng Wang, and Ling Shao. Few-shot deep adversarial learning for video-based person re-identification. *IEEE Trans. Image Processing*, 29(1):1233–1245, 2020. 1

[42] Tao Wu, Tie Luo, and Donald C Wunsch. Gnp attack: Transferable adversarial examples via gradient norm penalty. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 3110–3114. IEEE, 2023. 2, 3

[43] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. In *International Conference on Learning Representations*, 2018. 5, 7

[44] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019. 5, 6

[45] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *Proceedings 2018 Network and Distributed System Security Symposium*. Internet Society, 2018. 5, 7

[46] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015. 5, 7

[47] Hegui Zhu, Yuchen Ren, Xiaoyan Sui, Lianping Yang, and Wuming Jiang. Boosting adversarial transferability via gradient relevance attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4741–4750, 2023. 2, 5, 6, 7, 8