On Success and Simplicity: A Second Look at Transferable Targeted Attacks

Anonymous Author(s)
Affiliation
Address
email

Abstract

Achieving transferability of targeted attacks is reputed to be remarkably difficult. Currently, state-of-the-art approaches are resource-intensive because they necessitate training model(s) for each target class with additional data. In our investigation, we find, however, that simple transferable attacks which require neither additional data nor model training can achieve surprisingly high targeted transferability. This insight has been overlooked until now, mainly due to the widespread practice of unreasonably restricting attack optimization to a limited number of iterations. In particular, we, for the first time, identify that a simple logit loss can yield competitive results with the state of the arts. Our analysis spans a variety of transfer settings, especially including three new, realistic settings: an ensemble transfer setting with little model similarity, a worse-case setting with low-ranked target classes, and also a real-world attack against the Google Cloud Vision API. Results in these new settings demonstrate that the commonly adopted, easy settings cannot fully reveal the actual properties of different attacks and may cause misleading comparisons. We also show the usefulness of the simple logit loss for generating targeted universal adversarial perturbations in a data-free and training-free manner. Overall, the aim of our analysis is to inspire a more meaningful evaluation on targeted transferability.

1 Introduction

Deep neural networks have achieved remarkable performance in various machine learning tasks, but are known to be vulnerable to adversarial attacks [1]. A key property of adversarial attacks that makes them critical in practice is their transferability [2, 3]. Current work on adversarial transferability has achieved great success for non-targeted attacks [4–12], while several initial attempts [3, 4, 13] on targeted transferability have shown its extreme difficulty. Targeted transferability is worth exploring since it can raise more practical concerns by causing a specific, highly dangerous prediction.

However, so far state-of-the-art results can only be achieved by *resource-intensive transferable attacks* [14–16]. Specifically, the FDA approach [14, 15] is based on modeling layer-wise feature distributions via class-specific auxiliary classifiers using large-scale labeled data, and then optimizing adversarial perturbations using these auxiliary classifiers from across the deep feature space. The TTP approach [16] is based on training target class-specific Generative Adversarial Networks (GANs) through global and local distribution matching, and then use the trained generator to directly generate perturbations on any given input image.

In this paper, we take a second, thorough look at current research on targeted transferability. Our main contribution is the finding that *simple transferable attacks* [4, 6, 8] that require neither additional data nor model training can actually achieve surprisingly high targeted transferability. We argue that this insight has been overlooked mainly because current research has unreasonably restricted

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

attack optimization to a limited number of iterations (see detailed discussion in Section 3). Another key contribution of our work is, for the first time, identifying the general superiority of a simple logit loss, which even achieves competitive performance with resource-intensive transferable attacks. Although the idea of attacking logits is not new, its superiority in targeted transferability has not been recognized so far. We empirically find that directly using the well-known logit-based C&W loss [17] yields worse results.

In order to validate the general effectiveness of simple transferable attacks, in Section 4.1, we conduct extensive experiments in a wide range of transfer settings. We test the standard single-model and ensemble transfer settings, but also introduce several new settings that are more challenging and realistic. They include an ensemble transfer setting with little model similarity, a worse-case setting with low-ranked target classes, and also a real-world attack against the Google Cloud Vision API. Experimental results in these new settings suggest that evaluation in only the commonly adopted, easy settings cannot reveal the actual properties of different attacks, and may cause misleading comparing results. Additional experiments in Section 4.2 have shown the competitive performance of the simple transferable attacks to the state-of-the-art resource-intensive transferable attacks. Finally, in Section 4.3, inspired by the observation that the generated perturbations can reflect target semantics, we introduce a data-free and training-free approach to generating targeted Universal Adversarial Perturbations (UAPs) based on the simple logit attack. In contrast, exiting approaches [16, 18–20] to generating targeted UAPs inevitably require training on additional data.

Overall, this paper elucidates the weakness in common practice and the limitations of the commonly adopted transfer settings. We hope our analysis will inspire a more meaningful evaluation on targeted transferability. The code has been submitted as the supplemental material.

2 Related Work

In this section, we review simple transferable attacks (Section 2.1), and also recently-proposed resource-intensive transferable attacks (Section 2.2). Finally, we discuss related work on generating universal adversarial perturbations.

2.1 Simple Transferable Attacks

We refer to the transferable attacks that require neither additional data nor model training but using only iterative optimization on the original image to *simple transferable attacks*. Simple transferable attacks have been extensively studied in the non-targeted scenario [4–12], and also explored in the targeted scenario [3, 4, 21]. These attacks are commonly built up on the well-known Iterative-Fast Gradient Sign Method (I-FGSM) [22, 23]. In the targeted scenario, I-FGSM can be formulated as:

$$\boldsymbol{x}'_0 = \boldsymbol{x}, \;\; \boldsymbol{x}'_{i+1} = \boldsymbol{x}'_i - \alpha \cdot \; \mathrm{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{x}'_i, y_t)), \tag{1}$$

where $\boldsymbol{x}'_i$ denotes the intermediate modified image in the i-th iteration, and y_t is the target class label. In order to ensure the imperceptibility of the modifications, in each iteration, the perturbations are clipped into some pre-defined L_p norm ball, i.e., satisfying $\|\boldsymbol{x}' - \boldsymbol{x}\|_p \leq \epsilon$. Current research on transferable attacks has commonly adopted the L_∞ norm, but can also be easily extended to the L_2 norm by instead using a L_2 normalization [24]. For the loss function $J(\cdot, \cdot)$, most simple transferable attacks have adopted the plain cross-entropy loss .

However, this commonly used cross-entropy loss has been recently pointed out to be not effective in the targeted scenario because it suffers from the vanishing gradient problem [21]. Therefore, the authors in [21] have proposed a so-called **Po+Trip** loss to address the problem. Specifically, the Poincaré distance was used to adapt the magnitude of the gradients, which can be formulated as:

$$\begin{gathered} L_{Po} = d(\boldsymbol{u}, \boldsymbol{v}) = \mathrm{arccosh}(1 + \delta(\boldsymbol{u}, \boldsymbol{v})), \\ \delta(\boldsymbol{u}, \boldsymbol{v}) = \frac{2 \cdot \|\boldsymbol{u} - \boldsymbol{v}\|_2^2}{(1 - \|\boldsymbol{u}\|_2^2)(1 - \|\boldsymbol{v}\|_2^2)}, \; \boldsymbol{u} = \frac{l(\boldsymbol{x}')}{\|l(\boldsymbol{x}')\|}, \; \boldsymbol{v} = \max\{\boldsymbol{v} - \xi, 0\}, \end{gathered} \tag{2}$$

where $\boldsymbol{u}$ is the normalized logit vector and $\boldsymbol{v}$ is the one-hot vector with respect to the target class. $\xi = 10^{-5}$ is a small constant to ensure numerical stability. In order to further boost the performance, an additional triplet loss was integrated to also push the image away from the original class:

$$L_{Trip} = [D(l(\boldsymbol{x}'), y_t) - D(l(\boldsymbol{x}'), y_o) + \gamma]_+, \; D(l(\boldsymbol{x}'), y) = 1 - \frac{\|l(\boldsymbol{x}') \cdot y\|_1}{\|l(\boldsymbol{x}')\|_2 \|y\|_2}. \tag{3}$$

The overall loss function can be formulated as $L_{Po+Trip} = L_{Po} + \lambda L_{Trip}$. However, in the original work, Po+Trip was evaluated only in an ensemble transfer setting.

Instead of improving the loss function, there are also transfer techniques [4, 6, 8, 9] that aim at preventing the attack optimization from overfitting to the specific source model. The advantage of such techniques is that they can be generally applied to any loss functions. In this paper, we study three [4, 6, 8] of such transfer techniques that have been widely used in the literature:

Momentum Iterative-FGSM (MI-FGSM) [4] integrates a momentum term, which accumulates previous gradients in order to make more accurate updating. It can be expressed as:

$$\boldsymbol{g}_{i+1} = \mu \cdot \boldsymbol{g}_i + \frac{\nabla_{\boldsymbol{x}} J(\boldsymbol{x}'_i, y_t)}{\|\nabla_{\boldsymbol{x}} J(\boldsymbol{x}'_i, y_t)\|_1},\ \boldsymbol{x}'_{i+1} = \boldsymbol{x}'_i - \alpha \cdot\ \mathrm{sign}(\boldsymbol{g}_i), \tag{4}$$

where $\boldsymbol{g}_i$ is the accumulated gradients at the i-th iteration, and μ is the decay factor. Another similar technique that is based on the Nesterov accelerated gradient was also explored [9].

Translation Invariant-FGSM (TI-FGSM) [6] randomly translates the input image during optimization for preventing the attack from overfitting to the specific white-box source model. This approach is inspired by the common data augmentation techniques used for preventing overfitting in model training. Instead of calculating gradients for multiple translated images separately, the authors proposed an approximate solution to accelerate the implementation. It is achieved by directly computing locally smoothed gradients on the original image via convolution with a kernel:

$$\boldsymbol{x}'_{i+1} = \boldsymbol{x}'_i - \alpha \cdot\ \mathrm{sign}(\boldsymbol{W} * \nabla_{\boldsymbol{x}} J(\boldsymbol{x}'_i, y_t)), \tag{5}$$

where $\boldsymbol{W}$ is the convolution kernel used for smoothing. TI-FGSM was originally designed for transferring to adversarially-trained models. It has been recently pointed out that using relatively small kernel size leads to the optimal transferability when transferring to normally trained models [12].

Diverse Input-FGSM (DI-FGSM) [8] is conceptually similar to TI-FGSM, but applies random resizing and padding for the image augmentation. More importantly, instead of fixing the augmentation parameters all the time as in TI-FGSM, DI-FGSM adds more randomness across iterations. The attack optimization can be formulated as:

$$\boldsymbol{x}'_{i+1} = \boldsymbol{x}'_i - \alpha \cdot\ \mathrm{sign}(\nabla_{\boldsymbol{x}} J(T(\boldsymbol{x}'_i, p), y_t)), \tag{6}$$

where the stochastic transformation $T(\boldsymbol{x}'_i, p)$ is implemented with probability p at each iteration. In Section 4.1, we demonstrate that the above described simple transfer attacks can actually achieve surprisingly high targeted transferability in a wide range of transfer settings.

2.2 Resource-Intensive Transferable Attacks

Due to the broad consensus that achieving targeted transferability is extremely difficult, recent researchers have resorted to resource-intensive approaches that require training target class-specific models on large-scale additional data. Specifically, **Feature Distribution Attack (FDA)** [14] has substantially improved targeted transferability on ImageNet by training auxiliary models with additional labeled data. Each auxiliary model is a small, binary, one-versus-all classifier trained for a specific target class at a specific layer. That is to say, the number of auxiliary models is the number of layers probed multiplied by the number of target classes that are required to model [14].

The attack loss function can be formulated as:

$$L_{FDA} = J(\mathcal{F}_l(\boldsymbol{x}'), y_t) - \eta \frac{\|\mathcal{F}_l(\boldsymbol{x}') - \mathcal{F}_l(\boldsymbol{x})\|_2}{\|\mathcal{F}_l(\boldsymbol{x})\|_2}, \tag{7}$$

where each auxiliary models $F_l(\cdot)$ can model the probability that a feature map at layer l is from a specific target class y_t. **FDA$^{(N)}$+xent** [15] extends FDA by aggregating features from N layers and also incorporating the cross-entropy loss $H(\cdot,\cdot)$ of the original network $\mathcal{F}(\cdot)$. The extended loss function can be expressed as:

$$L_{FDA^{(N)}+xent} = \sum_{l \in L} \lambda_l (L_{FDA} + \gamma H(\mathcal{F}(\boldsymbol{x}'), y_t)),\ \text{where}\ \sum_{l \in L} \lambda_l = 1. \tag{8}$$

Very recently, **TTP** [16] has achieved state-of-the-art targeted transferability by directly generating perturbations using target class-specific GANs that have been trained with global and local distribution

matching. Specifically, the global distribution matching is achieved by minimizing the Kullback Leibler (KL) divergence, while the local distribution matching is by enforcing the neighbourhood similarity. In order to further boost the performance, image augmentation techniques, such as rotation, crop resize, horizontal flip, color jittering and gray scale transformation, have been applied during training. We refer the readers to the original work for more technical details of TTP.

These resource-intensive transferable attacks have a stronger assumption about direct access to actual data samples from the target class. However, in Section 4.2, we show that simple transferable attacks, which require neither additional data nor model training, can actually achieve competitive performance with them.

2.3 Universal Adversarial Perturbations

Previous research has shown the existence of Universal Adversarial Perturbations (UAPs), i.e., a single image perturbation vector that fools classifiers on multiple images [25]. Generating UAPs has been extensively studied in the non-targeted scenario [25–29], but also explored in the more challenging, targeted scenario [18–20]. Generally speaking, achieving universality and transferabibity share the same idea, namely, capturing the global characteristics of data distribution without overfitting to specific images or models. As a result, existing UAP approaches have also shown a certain transferability across different models. Specifically, the state-of-the-art TTP [16] on targeted transferability was also built up on this idea.

A limitation of convention work on generating UAPs is their need of additional data and training. Several approaches [19, 26, 27] have been proposed to address this limitation. Specifically, in the non-targeted scenario, Mopuri et al. [26] proposed a data-free approach by only using class impressions that are reconstructed from a specific model for training. Similarly, in the targeted scenario, Zhang et al. [19] have used a proxy dataset for training based on their finding that the influence of the input images on targeted UAPs is like noise. Data-free and training-free approach Mopuri et al. [27] has been proposed in the non-targeted scenario by iteratively optimizing the perturbations against the single target model with an objective of corrupting the extracted features at multiple layers. However, this approach cannot be directly applied to the targeted scenario. In Section 4.3, we demonstrate that the simple logit attack can serve as a potential solution to generating targeted UAPs in a data-free and training-free manner.

3 New Insights into Simple Transferable Attacks

In this section, we revisit simple transferable attacks in the targeted scenario, and provide new insights into them.

Simple transferable attacks are surprisingly good when given enough iterations to converge. Existing attempts on using simple transferable attacks for targeted transferability has concluded its extreme difficulty [3, 4, 13]. However, they have been focused on the MI technique. As can be seen form Figure 1, somewhat surprisingly, we find that using only DI can already yield substantial targeted transferability, while using only TI or MI makes little difference to the original poor targeted transferability. In particular, integrating all three transfer techniques leads to the best performance. This may be explained by the fact that DI randomizes the image augmentation not only across the image dimension but also across iterations. In this way, the gradients towards the target class become more generic and so avoid overfitting to the white-box source model. In comparison, TI introduces less randomness since it fixes the image augmentation parameters for all iterations. MI is essentially different from the above two because it aims at preventing the optimization falling into poor minima but can hardly help achieve more accurate gradient directions towards some specific target class. The results reported in [16] have also suggested the superiority of DI (under limited iterations), while other studies [13–15] have lacked individual comparisons with DI.

As we have pointed out in Section 1, existing work on targeted transferability [13–15, 21] has been focused on attacking with only a limited number of iterations (typically ≤ 20). This is somewhat understandable given the fact that extensive research on non-targeted transferability has done the same. However, as can be seen from Figure 1, the simple transferable attacks can converge to much higher targeted transferability with more iterations, in contrast to the fast success in the non-targeted scenario. This implies that evaluating the targeted transferability under only few iterations is problematic. On

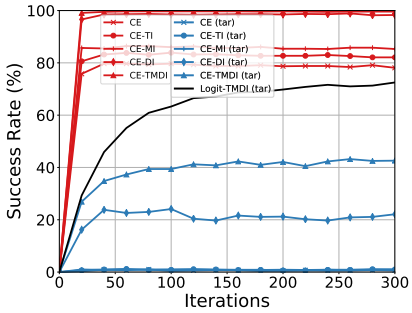


Figure 1: Transfer success rates of simple transferable attacks using CE or logit loss in the non-targeted and targeted scenarios.

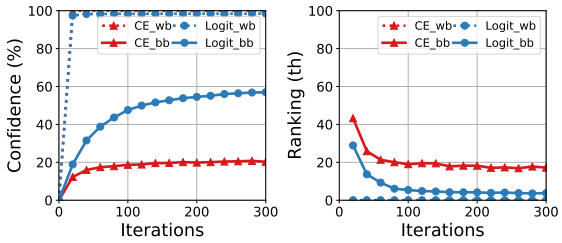


Figure 2: White-box (wb) and black-box (bb) targeted attack performance in terms of the predicted confidence (left, higher is better) and ranking (right, lower is better) of the target class.

the one hand, comparing different optimization processes that have not converged is not meaningful and may cause misleading comparisons (see evidence in Section 4.1). This observation is consistent with Carlini et al. [30], who pointed out that restricting the number of iterations without verifying the attack convergence is one of the common pitfalls in evaluating adversarial robustness. Several advanced defenses have been defeated by simply increasing the number of iterations [31]. On the other hand, considering the realistic threat model, it is not meaningful to artificially restrict the computational power of a practical attack (e.g., to fewer than several thousand attack iterations) [32].

A simple yet stronger logit attack. Existing simple transferable attacks are commonly built up on the cross-entropy loss. However, as pointed out by Li et al. [21], the cross-entropy loss may suffer from the vanishing gradient problem. In our case with many iterations, this problem becomes more serious since it will cause the attack to stop improving at some point. To address this problem, we adopt a straightforward idea by directly maximizing the target logit for continuously pushing the image deep into the territory of the target class. This simple logit loss can be expressed as:

$$L_{Logit} = -l_t(\boldsymbol{x}'), \tag{9}$$

where $l_t(\cdot)$ denotes the logit output with respect to the target class. Although this logit loss is very similar to the well-known C&W loss [17], its superiority on targeted transferability has not been recognized so far. We find that directly using the C&W loss performs worse than this logit loss. For example, in the single-model transfer setting with the ResNet50 as the white-box source model, the C&W loss yields 20% worse performance on average. It may be because the C&W loss also involves suppressing other classes, which is not necessary here but could trade off the high logit value with respect to the target class.

As can be seen from Figure 1, this logit loss consistently yields higher targeted transferability than the commonly used cross-entropy loss, and the performance gap becomes increasingly larger as the number of iterations rises. It implies that the logit loss can effectively avoid the vanishing gradient problem. This large improvement can also be confirmed by Figure 2, which shows that the logit attack pushes the target class to much higher confidence and ranking position in the black-box (transfer) setting, although both two attacks achieved equally perfect performance in the white-box setting.

4 Experimental Evidence on Simple Transferable Attacks

In this section, we provide experimental evidence to show the general effectiveness of simple transferable attacks. Firstly, in Section 4.1, we evaluate the simple transferable attacks in a variety of transfer settings, including single-model transfer, ensemble transfer (easy and challenging settings), a worse-case setting with low-ranked target classes, and a real-world attack against the Google Cloud Vision API. Then, in Section 4.2, we compare the simple transferable attacks with two state-of-the-art resource-intensive transferable attacks, FDA$^{(N)}$+xent [15] and TTP [16]. Finally, in Section 4.3, we apply the logit attack to achieving targeted UAPs in a data-free and training-free manner.

Following recent work [14–16, 21], we focus on addressing the targeted transferability on ImageNet-like images, which is known to be much more difficult than on other small-scale datasets, such as MNIST and CIFAR-10. Specifically, we use the 1000 images from the development set of the

Table 1: Targeted transfer success rates (%) in the single-model transfer setting. We consider three attacks with different loss functions: cross-entropy (CE), Poincaré distance with Triplet loss (Po+Trip) [21], and the logit loss. Results with 20/100/300 iterations are reported.

Attack	Source Model: Res50			Source Model: Dense121		
	→Dense121	→VGG16	→Inc-v3	→Res50	→VGG16	→Inc-v3
CE	26.9/39.4/42.6	17.3/27.3/30.4	2.4/3.8/4.1	13.1/17.3/19.4	7.7/10.8/10.9	1.9/3.3/3.5
Po+Trip	26.7/53.0/54.7	18.8/34.2/34.4	2.9/6.0/5.9	10.1/14.7/14.7	6.7/8.3/7.7	2.1/3.0/2.7
Logit	**29.3/63.3/72.5**	**24.0/55.7/62.7**	**3.0/7.2/9.4**	**17.2/39.7/43.7**	**13.5/35.3/38.7**	**2.7/6.9/7.6**
Attack	Source Model: VGG16			Source Model: Inc-v3		
	→Res50	→Dense121	→Inc-v3	→Res50	→Dense121	→VGG16
CE	0.7/0.4/0.6	0.5/0.3/0.1	0/0.1/0	0.6/**2.1**/2.4	0.8/2.5/2.9	0.7/1.6/2.0
Po+Trip	0.6/0.8/0.5	0.6/0.6/0.7	0.2/0.1/0.1	0.6/2.0/2.5	0.8/**3.1**/3.3	0.5/2.1/2.0
Logit	**3.3/8.7/11.2**	**3.6/11.7/13.2**	**0.2/0.7/0.9**	**0.8**/1.6/**2.9**	**1.2**/2.8/**5.3**	**0.7/2.2/3.7**

ImageNet-Compatible Dataset[1], which was introduced along with the NIPS 2017 Competition on Adversarial Attacks and Defenses. All these images are associated with 1000 ImageNet class labels and cropped to 299×299 before use. Our experiments are run on a single NVIDIA Tesla P100 GPU with 12GB of memory.

4.1 Simple Transferable Attacks in Different Transfer Settings

We test three different attack losses: cross-entropy (CE), Po+Trip [21] and logit. All attacks use TI, MI, and DI together with the optimal hyperparameters reported in their original work. Specifically, $\|\boldsymbol{W}\|_1 = 5$ is used for 'TI' as suggested by [12]. For each image, we use the target label that was officially specified in the dataset. We use a moderate step size of 2 for all attacks, and find that the performance is not sensitive to step size (see evidence in Appendix). We consider four diverse classifier architectures: ResNet [33], DenseNet [34], VGGNet [35], and Inception [36]. Following the common practice, the perturbations are restricted by $L_\infty = 16/255$. When being executed with a batch size of 20 and 100 iterations, the optimization of each attack takes about one second per image.

Single-model transfer. Table 1 reports the targeted transferability in cases with different classifiers. As can be seen, the logit loss outperforms the other two by a large margin in almost all cases. When comparing different model architectures, we can find that the attacks generally achieved lower performance when using the VGGNet16 or Inception-v3 as the white-box source model compared with using ResNet50 or DenseNet121. This is consistent with the observations in [14, 15]. It may be explained by the fact that skip connections (in ResNet50 and DenseNet121) allow easier generation of highly transferable adversarial examples [37]. When using Inception-v3 as the target model, the low performance might be explained by its notable difference with the other three architectures, i.e., the Inception architecture is known to be more conceptually complex (heavily engineered) with multiple-size convolution and two auxiliary classifiers.

Ensemble transfer in both easy and challenging settings. A common approach to further boosting transferability is to optimize the perturbations against an ensemble of white-box source models. Following the common practice, we simply assign equal weights to all the source models. We first look at the commonly adopted ensemble transfer setting [4, 6, 21, 38] in which each hold-out target model shares similar architecture with some of the white-box ensemble models. As can be seen from Table 2, all the three attacks reached equally high targeted transferability when given enough iterations to converge. This indicates that this setting with high model similarity could not fully reveal the actual properties of different attacks. We can also observe that Po+Trip performs better than the cross-entropy loss only when the attack optimization is unreasonably restricted to 20 iterations, but becomes worse when they are given enough iterations to converge. This finding suggests that evaluating different attacks under only few iterations may cause misleading comparing results.

We next consider a more challenging transfer setting with no architectural overlap between the source ensemble models and the target model, in order to fully reveal the potential of different attacks. This

[1]This dataset is composed from publicly available images which were posted online under CC-BY license and is publicly available from `https://github.com/cleverhans-lab/cleverhans/tree/master/cleverhans_v3.1.0/examples/nips17_adversarial_competition/dataset`. Our manual check confirms that all images do not contain any offensive content.

Table 2: Targeted transfer success rates (%) in an easy, previously-studied ensemble transfer setting, where the hold-out target model (denoted by '-') and the ensemble models share similar architectures. Results with 20/100 iterations are reported.

Attack	-Inc-v3	-Inc-v4	-IncRes-v2	-Res50	-Res101	-Res152	Average
CE	48.8/85.3	47.2/83.3	47.5/83.9	50.9/89.8	58.5/**93.2**	56.7/90.7	51.6/87.7
Po+Trip	**59.3**/84.4	**55.0**/82.4	51.4/80.8	56.9/85.0	60.5/87.9	57.6/85.7	56.8/84.4
Logit	56.4/**85.5**	52.9/**85.8**	**54.4/85.1**	**57.5/90.0**	**64.4**/91.4	**61.3/90.8**	**57.8/88.1**

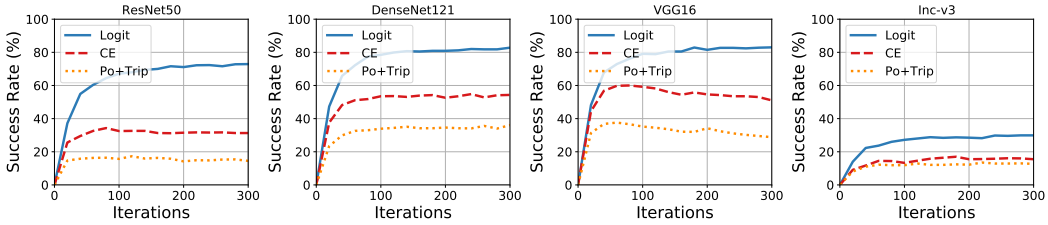


Figure 3: Targeted transfer success rates (%) in our challenging, realistic ensemble setting, where each hold-out target model shares no similar architecture with the source models used for ensemble.

setting is also more realistic since there is unlikely to be a local white-box source model that shares similar architecture to the unknown target model. Figure 3 shows that in this new setting, the logit attack largely outperforms the other two attacks. Specifically, the results of both the cross-entropy and logit attacks were substantially improved over the single-model transfer results, as reported in Table 1. However, Po+Trip performs even worse in some cases maybe because its original optimal hyperparameters cannot ensure similar effectiveness in this more challenging transfer setting. Again, Inception-v3 is the most difficult model to attack.

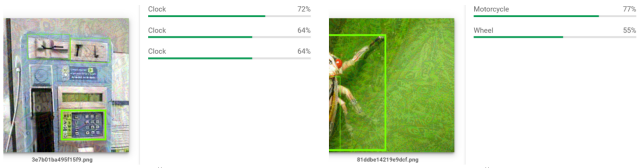
A worse-case transfer setting with low-ranked target classes. In conventional security studies, a comprehensive evaluation of attacks commonly involves a range of settings with varied difficulties. For adversarial attacks in the white-box scenario, existing studies [17, 22, 24, 38] have also looked at varied difficulties regarding the target class. Specifically, in the best case, the targeted success is basically equal to non-targeted success, i.e., an attack is regarded to be successful as long as it can success on any arbitrary target other than the original class. In the average case, the target class is randomly specified agnostic to the test image, while in the worst case, the target is specified as the lowest-ranked (least-likely) class in the prediction list of the original image.

Table 3: Targeted transfer success rates (%) when varying the target from the high-ranked class to low.

Attack	2nd	10th	200th	500th	800th	1000th
CE	**89.9**	76.7	49.7	43.1	37.0	25.1
Po+Trip	82.6	77.6	58.4	53.6	49.1	38.2
Logit	83.8	**81.3**	**75.0**	**71.0**	**65.1**	**52.8**

However, to the best of our knowledge, current evaluation on transferability has been limited to the best and average cases. To address this limitation, we consider a worse-case transfer setting by varying the target from the highest-ranked class gradually to the lowest one. As can be seen from Table 3, there exists non-negligible correlation between the ranking position of the target class and the transferability. More specifically, it becomes increasingly difficult as the target moves down the prediction list. We can also observe that the results with higher-ranked targets could not reveal the actual properties of different attacks in more realistic, worse cases with lower-ranked targets. Specifically, only looking the best case with the highest-ranked target may lead to a misleading conclusion that the cross-entropy loss leads to the most effective attack. This finding suggests that a more meaningful evaluation on targeted transferability should involve more difficult settings beyond the current best and average cases.

Transfer-based attack on Google Cloud Vision. Most existing work on fooling real-world commercial machine learning systems has been focused on the query-based attacks, where a large number of queries are required [39–41]. Although several recent studies have also explored transfer-based attacks in this context, they were limited to face recognition and only discussed in the non-targeted scenario [42–44]. In contrast, we apply the simple transferal attacks to attacking a more generally-used image recognition system, the Google Cloud Vision API, in the more challenging, targeted scenario. Specifically, we used the targeted adversarial images computed against the ensemble of all



Target class: "analog clock" Target class: "mountain bike"

Figure 4: Attacking Google Cloud Vision (object detection) using the logit attack with ensemble transfer. More examples can be found in Appendix.

Table 4: Non-targeted and targeted transfer success rates (%) of different attacks against Google Cloud Vision.

	CE	Po+Trip	Logit
Targeted	7	8	**18**
Non-targeted	**51**	44	**51**

four diverse white-box source models with 300 iterations. For targeted transferability, we evaluate whether or not the target object class appears among the returned (top 10) predictions. We also report the non-targeted results to give extra insight by evaluating whether or not the correct object class appears among the returned predictions. Since the Google Cloud Vision predictions do not explicitly correspond to the 1000 ImageNet classes, we treat semantically similar classes as the same class.

Table 4 reports the results averaged over 100 images that originally yield correct predictions. As can be seen, in general, fooling the API into predicting some target class is much more difficult than only altering the original class. The logit attack achieved the best targeted transferability with quasi-imperceptible perturbations, as shown in Figure 4. Our results reveal the potential vulnerability of Google Cloud Vision against simple attacks that can be crafted without any query interaction.

4.2 Simple vs. Resource-Intensive Transferable Attacks

Compared with TTP [16]. We compare the best performed simple attack, the logit attack, with the state-of-the-art TTP, which is based on training class-specific GANs with additional data. We test both attacks on our dataset following the original "10-Targets (all-source)" setting [16]. We choose the ResNet50 as the white-box model in the single-model transfer setting and an ensemble of ResNet{18,50,101,152} in the ensemble transfer setting. DenseNet121 and VGG16_bn are tested as the targeted models. Note that the knowledge of the white-box model for both attacks is the same but is leveraged in different ways. Specifically, the white-box model is used as a surrogate during the iterative attack optimization for the logit attack, while as a discriminator during training the GANs for TTP. As shown in Table 5, the logit attack achieved high targeted transferability comparable to TTP in both transfer settings, with slightly better performance on average.

Table 5: Targeted transfer success rates (%) of the logit attack compared with TTP [16]. DenseNet121 (D121) or VGG16_bn (V16) is the target model, and ResNet50 or an ensemble (-ens) of ResNet{18,50,101,152} is the source model.

Attack	D121	V16	D121-ens	V16-ens	Average
TTP	**79.6**	**78.6**	92.9	89.6	85.2
Logit	75.9	72.5	**99.4**	**97.7**	**86.4**

Compared with FDA$^{(N)}$+xent [15]. We compare simple transferable attacks with FDA$^{(N)}$+xent in a specific setting with unbounded adversarial examples[2] (referred to as "distal transfer" in the original work [15]). Specifically, the adversarial examples are generated by starting from random Gaussian noise and optimizing without any perturbation bounds for as high as possible target class confidence. Although such unbounded adversarial examples may not be practically compelling, they can provide better isolated indication on transferability by eliminating the dependency on the source images/classes and the bound restrictions. Two single-model transfer settings (Res50→Dense121 and Res50→VGG16) are considered and the results are averaged over 4,000 examples, each of which is optimized towards a random target class. As suggested by [15], the MI technique is removed since it empirically harms the performance in this unbounded case.

Table 6 shows that all three simple transferable attacks achieved higher targeted transferability than the state-of-the-art FDA$^{(N)}$+xent. Specifically, the logit loss again yielded the best overall performance. As can be seen from Figure 5, the unbounded adversarial perturbations can somehow reflect the target semantics. This suggests an interesting perspective that attacking the robust features of the target class can boost (targeted) transferability because these features are naturally shared by different

[2] We chose to conduct comparisons in only this setting since the authors have not (yet) released their source code.

Table 6: Targeted transfer success rates (%) of unbounded adversarial examples. All attacks use 200 iterations.

	$\text{FDA}^{(4)}$ +xent	CE	Po+Trip	Logit
Res50→Dense121	65.8	69.3	**88.1**	84.1
Res50→VGG16	48.1	54.1	67.8	**74.2**

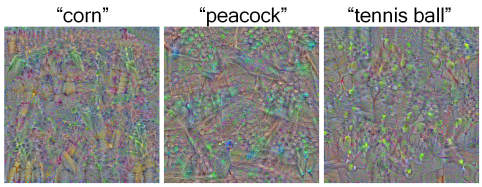


Figure 5: Unbounded adversarial examples that reflect target semantics. More examples can be found in Appendix.

models (and also humans). This perspective provides new insight beyond previous hypothesis that transferability is due to the high similarity between the non-robust features learned by different models [45]. Note that in practical scenarios with small, bounded perturbations, the semantic features in the adversarial images could hardly change human judgements.

4.3 Usefulness of Logit Loss for Data-Free and Training-Free Targeted UAPs

The observation that the perturbations can reflect certain target semantics motivates us to apply the logit attack to achieving targeted Universal Adversarial Perturbations (UAPs), which can drive multiple image examples into a specific target class. Existing attempts on achieving targeted UAPs have mainly addressed the problem based on training on original image dataset [18–20]. Although Zhang et al. [19] have eliminated the need of original training data by using a proxy dataset, their approach still requires training. In contrast, we introduce a promising approach to generating the targeted UAPs without any actual data and training, but just use the logit attack starting from a mean image (all pixel values set as 0.5 out of [0,1]). In our experiment, given a specific target class, we generated a single targeted UAP vector with 300 iterations and applied it to all 1000 images in our dataset.

Table 7 reports the results averaged over all the 1000 ImageNet classes. As can be seen, the logit loss can yield substantial success, remarkably outperforming the cross-entropy loss. Our results also somewhat support the claim from Zhang et al. [19] that universal perturbations contain dominant features, and images act like noise with respect to perturbations.

Table 7: Success rates (%) of targeted UAPs.

Attack	Inc-v3	Res50	Dense121	VGG16
CE	2.6	9.2	8.7	20.1
Logit	**4.7**	**22.8**	**21.8**	**65.9**

5 Conclusion and Outlook

In this paper, we have demonstrated that achieving targeted transferability is not as difficult as current work concludes. Specifically, we find that simple transferable attacks can actually achieve surprisingly good performance as long as they are not unreasonably restricted to few iterations. We have validated the effectiveness of simple transferable attacks in a wide range of transfer settings, including several newly-introduced challenging settings. These challenging settings have better revealed the actual properties of different attacks. In particular, we identify that a very simple logit attack can consistently achieve the highest targeted transferbility in all settings. Additional comparisons with the state-of-the-art resource-intensive transferable attacks have shown the competitive performance of simple transferable attacks, especially the logit attack. We also show the potential usefulness of the logit attack for generating targeted universal adversarial perturbations without any addition data and training. Overall, we hope our findings on weakness in common practice and the limitations of the commonly adopted transfer settings will inspire future research to conduct a more meaningful evaluation on targeted transferability.

Our future work will be to explore what kind of model properties have caused this effect since our analysis has shown that the targeted transferability degrades significantly on some model architectures. The discussion about the feature separability in [15] could serve as a starting point in this direction. It is known that high adversarial transferability could be potentially harmful to real-world systems when being leveraged by malicious parties. However, our work is aimed to motivate the community to design stronger defenses given our finding that even simple attacks can generate highly transferable adversarial examples. On the other hand, high transferability of adversarial examples can be directly leveraged to promote applications for social good, such as protecting user privacy [42, 43, 46–48].

References

[1] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014.

[2] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

[3] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *ICLR*, 2017.

[4] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018.

[5] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang. Transferable adversarial perturbations. In *ECCV*, 2018.

[6] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *CVPR*, 2019.

[7] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *ICCV*, 2019.

[8] 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 *CVPR*, 2019.

[9] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *ICLR*, 2020.

[10] Yingwei Li, Song Bai, Yuyin Zhou, Cihang Xie, Zhishuai Zhang, and Alan L Yuille. Learning transferable adversarial examples via ghost networks. In *AAAI*, 2020.

[11] Weibin Wu, Yuxin Su, Xixian Chen, Shenglin Zhao, Irwin King, Michael R Lyu, and Yu-Wing Tai. Boosting the transferability of adversarial samples via attention. In *CVPR*, 2020.

[12] Lianli Gao, Qilong Zhang, Jingkuan Song, Xianglong Liu, and Heng Tao Shen. Patch-wise attack for fooling deep neural network. In *ECCV*, 2020.

[13] Nathan Inkawhich, Wei Wen, Hai Helen Li, and Yiran Chen. Feature space perturbations yield more transferable adversarial examples. In *CVPR*, 2019.

[14] Nathan Inkawhich, Kevin Liang, Lawrence Carin, and Yiran Chen. Transferable perturbations of deep feature distributions. In *ICLR*, 2020.

[15] Nathan Inkawhich, Kevin J Liang, Binghui Wang, Matthew Inkawhich, Lawrence Carin, and Yiran Chen. Perturbing across the feature hierarchy to improve standard and strict blackbox attack transferability. In *NeurIPS*, 2020.

[16] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. On generating transferable targeted perturbations. In *arXiv preprint arXiv:2103.14641*, 2021.

[17] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE S&P*, 2017.

[18] Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. In *CVPR*, 2018.

[19] Chaoning Zhang, Philipp Benz, Tooba Imtiaz, and In So Kweon. Understanding adversarial examples from the mutual influence of images and perturbations. In *CVPR*, 2020.

[20] Philipp Benz, Chaoning Zhang, Tooba Imtiaz, and In So Kweon. Double targeted universal adversarial perturbations. In *ACCV*, 2020.

[21] Maosen Li, Cheng Deng, Tengjiao Li, Junchi Yan, Xinbo Gao, and Heng Huang. Towards transferable targeted attack. In *CVPR*, 2020.

[22] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *ICLR*, 2017.

[23] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

[24] Jérôme Rony, Luiz G. Hafemann, Luiz S. Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based L_2 adversarial attacks and defenses. In *CVPR*, 2019.

[25] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *CVPR*, 2017.

[26] Konda Reddy Mopuri, Phani Krishna Uppala, and R Venkatesh Babu. Ask, acquire, and attack: Data-free UAP generation using class impressions. In *ECCV*, 2018.

[27] Konda Reddy Mopuri, Aditya Ganeshan, and R Venkatesh Babu. Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE TPAMI*, 41(10):2452–2465, 2018.

[28] Muzammal Naseer, Salman H Khan, Harris Khan, Fahad Shahbaz Khan, and Fatih Porikli. Cross-domain transferability of adversarial perturbations. In *NeurIPS*, 2019.

[29] Yingwei Li, Song Bai, Cihang Xie, Zhenyu Liao, Xiaohui Shen, and Alan L Yuille. Regional homogeneity: Towards learning transferable universal adversarial perturbations against defenses. In *ECCV*, 2020.

[30] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. In *arXiv preprint arXiv:1902.06705*, 2019.

[31] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In *NeurIPS*, 2020.

[32] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[34] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[36] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

[37] Dongxian Wu, Yisen Wang, Shu-Tao Xia, James Bailey, and Xingjun Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. In *ICLR*, 2020.

[38] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *ICLR*, 2018.

[39] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *AISec*, 2017.

[40] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *ICLR*, 2018.

[41] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *ICML*, 2018.

[42] Valeriia Cherepanova, Micah Goldblum, Harrison Foley, Shiyuan Duan, John Dickerson, Gavin Taylor, and Tom Goldstein. LowKey: Leveraging adversarial attacks to protect social media users from facial recognition. In *ICLR*, 2021.

[43] Arezoo Rajabi, Rakesh B Bobba, Mike Rosulek, Charles Wright, and Wu-chi Feng. On the (im) practicality of adversarial perturbation for image privacy. *PoPETs*, 2021.

[44] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *USENIX Security*, 2020.

[45] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019.

[46] Martha Larson, Zhuoran Liu, Simon Brugman, and Zhengyu Zhao. Pixel privacy: Increasing image appeal while blocking automatic inference of sensitive scene information. In *MediaEval Multimedia Benchmark Workshop*, 2018.

[47] Zhuoran Liu, Zhengyu Zhao, and Martha Larson. Who's afraid of adversarial queries? the impact of image modifications on content-based image retrieval. In *ICMR*, 2019.

[48] Seong Joon Oh, Mario Fritz, and Bernt Schiele. Adversarial image perturbation for privacy protection a game theory perspective. In *ICCV*, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We described the limitation in the first sentence of the final section.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We discuss them at the end of the final section.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code (including instructions and a link to data we have used) has been submitted as the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Our experiments did not involve any model training, but all the algorithm details were specified.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We fixed the random seed for all the experiments. Now the main randomness comes from the p in the existing method, DI (see Eq. 6). We confirm that it has little impact on the results.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We mentioned the run time of the attacks and our hardware settings in Section 4.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] In Section 4.
   (b) Did you mention the license of the assets? [Yes] In a footnote of Section 4.

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A] We did not use any new assets.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] In the same footnote as above.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] In the same footnote as above.

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]