Appendix for: Boosting Targeted Adversarial Transferability via Hierarchical Generative Networks

Xiao Yang[1] Yinpeng Dong[1,2] Tianyu Pang[3] Hang Su[1,4] Jun Zhu[1,2,4⋆]

Dept. of Comp. Sci. and Tech., Institute for AI, Tsinghua-Bosch Joint ML Center, THBI Lab, BNRist Center, Tsinghua University[1]
RealAI[2] Sea AI Lab, Singapore[3]
Peng Cheng Laboratory; Pazhou Laboratory (Huangpu), Guangzhou, China[4]
{yangxiao19, dyp17}@mails.tsinghua.edu.cn, tianyupang@sea.com,
{suhangss, dcszj}@tsinghua.edu.cn

A Sampling Algorithm

We summarize the overall sampling procedure based on k-DPP [4] in Algorithm 1.

- Compute the RBF kernel matrix L of ϕ_{cls} and eigendecomposition of L.
- A random subset V of the eigenvectors is chosen by regarding the eigenvalues as sampling probability.
- Select a new class c_i to add to the set and update V in a manner that de-emphaseizes items similar to the one selected.
- Update V by Gram-Schmidt orthogonalization, and the distribution shifts to avoid points near those already chosen.

By performing the Algorithm 1, we can obtain a subset with k size. Thus while handling the conditional classes with K, we can hierarchically adopt this algorithm to get the final K/k subsets, which are regarded as conditional variables of generative models to craft adversarial examples.

B Some Implementation Details

The study of smoothing mechanism. Smoothing mechanism has been proved to improve the transferability against adversarially trained models. CD-AP [8] uses direct clip projection to have a fixed norm ϵ, and adopts smoothing for generated perturbation while the generator $\mathcal{G}$ is trained, i.e.,

$$\begin{aligned} \textbf{Train: } \boldsymbol{x}^*_{s_i} &= \mathrm{Clip}_\epsilon(\mathcal{G}(\boldsymbol{x}_{s_i}), \\ \textbf{Test: } \boldsymbol{x}^*_{s_i} &= \mathrm{W} * \mathrm{Clip}_\epsilon(\mathcal{G}(\boldsymbol{x}_{s_i}), \end{aligned} \tag{1}$$

⋆ corresponding author.

Algorithm 1 Sampling Algorithm by kDPP

Require: Weight Vector $\boldsymbol{\theta}_{cls}$; Subset size k.
Ensure: A subset C.
1: Compute RBF kernel matrix L of $\boldsymbol{\theta}_{cls}$;
2: Compute eigenvector/value $\{v_n, \lambda_n\}_{n=1}^{N}$ pairs of L;
3: *// Phase I:*
4: $J \leftarrow \phi$, $e_k\left(\lambda_1, \ldots, \lambda_N\right) = \sum_{|J|=k} \prod_{n \in J} \lambda_n$;
5: **for** n = N, ..., 1 **do**
6: **if** $u \sim U[0,1] < \lambda_n \frac{e_{k-1}^{n-1}}{e_k^n}$ and $k > 0$ **then**
7: $J \leftarrow J \cup \{n\}$; $k \leftarrow k-1$;
8: **end if**
9: **end for**
10: *// Phase II:*
11: $V \leftarrow \{v_n\}_{n \in J}, Y \leftarrow \phi$;
12: **while** $|V| > 0$ **do**
13: Select c_i from $\mathcal{C}$ with $\mathrm{P}\left(c_i\right) = \frac{1}{|V|} \sum_{v \in V} \left(v^{\top} e_i\right)^2$;
14: $C \leftarrow C \cup \{c_i\}$; $V \leftarrow V_{\perp}$, an orthonormal basis for the subspace of V orthogonal to e_i;
15: **end while**

where W indicates Gaussian smoothing of kernel size of 3, $*$ indicates the convolution operation, and Clip_{ϵ} means clipping values outside the fixed norm ϵ. As a comparison, we introduce adaptive Gaussian smoothing kernel to compute adversarial images $\boldsymbol{x}_{s_i}^*$ from in the training phase, named **adaptive Gaussian smoothing** as

$$\textbf{Train \& Test: } \boldsymbol{x}_{s_i}^* = \epsilon \cdot \mathrm{W} * \tanh(\mathcal{G}(\boldsymbol{x}_{s_i}) + \boldsymbol{x}_{s_i}, \tag{2}$$

which can make generated results obtain adaptive ability in the training phase. We perform training in ImageNet dataset to report all results including comparable baselines.

Network architecture of generator. We adopt the same autoencoder architecture in [8] as the basic generator networks. Besides, we also explore BigGAN [2] as conditional generator network. An very weak testing performance is obtained even in the *white-box* attack scenario, possibly explained by the weak diversity of latent variable with the Gaussian distribution from BigGAN in the training phase, whereas autoencoder can take full advantage of large-scale training dataset, e.g., ImageNet. Furthermore, we also train the autoencoder with Gaussian noise as the training dataset and obtain similar inferior performance in the white-box attack scenario, indicating that a large-scale training dataset is very significant for generating transferable targeted adversarial examples.

C Additional Experimental Results

Results on different datasets. We craft adversarial examples on different datasets, including ImageNet training set, MS-COCO and Comics dataset [1],

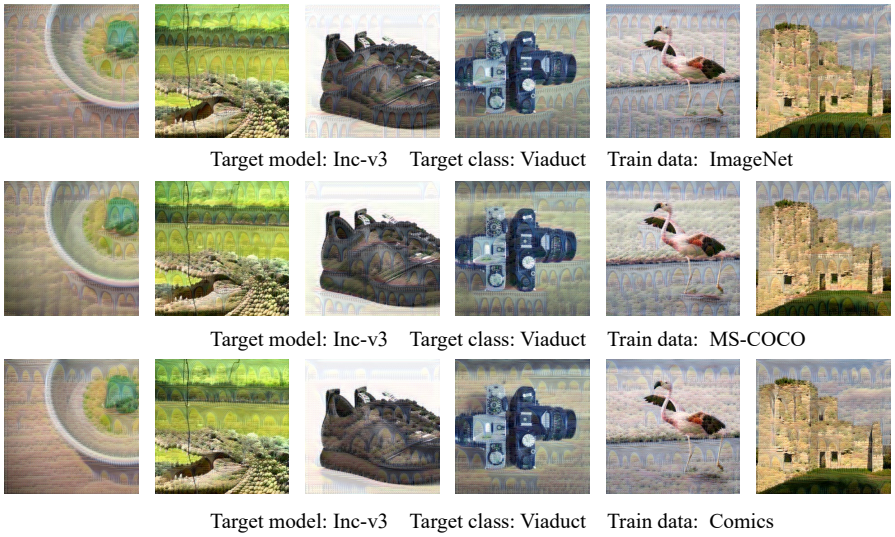

Fig. 1: Some examples of adversarial images with perturbation budget of $\ell_\infty \leq 16$. We separately adopt the ImageNet, MS-COCO and Comics dataset as the training dataset to implement the generation of targeted perturbations.

Table 1: Comparison results of targeted black-box attacks on different datasets. Inv3 is the substitute model.

Dataset	DN	VGG-16	GN
ImageNet	79.9	81.9	73.2
MS-COCO	70.3	71.3	64.1
Comics	60.4	63.0	61.3

which consist of 1.2M, 82k and 50K images, respectively. MS-COCO dataset can be applied to large-scale object detection and segmentation, and those images from Comics dataset are regarded as other domains different from normal ones in ImageNet. Despite this diverse training types, we still find the common property of crafted adversarial examples by our method. Specifically, we craft some examples of adversarial images with perturbation budget of $\ell_\infty \leq 16$, and separately adopt the ImageNet, MS-COCO and Comics dataset as the training dataset to implement the generation of targeted perturbations. As illustrated in Fig. 1, we produce semantic pattern independent of any training dataset.

We also report the success rate of targeted black-box attack, as shown in Table 1. We experimentally find that semantic pattern derived from ImageNet dataset achieves better performance of black-box performance, possibly explained by instructional effectiveness from more diverse data in ImageNet dataset.

Table 2: Transferability results for untargeted attacks increase in error rate after attack on subset of ImageNet (5k images) with the perturbation budget of $\ell_\infty \leq 16/32$.

	Method	$\text{inv3}_{\text{ens3}}$		$\text{inv3}_{\text{ens4}}$		$\text{IR-v2}_{\text{ens}}$	
		$\epsilon = 16$	$\epsilon = 32$	$\epsilon = 16$	$\epsilon = 32$	$\epsilon = 16$	$\epsilon = 32$
inv3	UAP [7]	1.00	7.82	1.80	5.60	1.88	5.60
	GAP [10]	5.48	33.3	4.14	29.4	3.76	22.5
	RHP [6]	32.5	60.8	31.6	58.7	24.6	57.0
inv4	UAP [7]	2.08	7.68	1.94	6.92	2.34	6.78
	RHP [6]	27.5	60.3	26.7	62.5	21.2	58.5
IR-v2	UAP [7]	1.88	8.28	1.74	7.22	1.96	8.18
	RHP [6]	29.7	62.3	29.8	63.3	26.8	62.8
CD-AP [8]		28.34	71.3	29.9	66.72	19.84	60.88
CD-AP-gs [8]		41.06	71.96	42.68	71.58	37.4	72.86
Ours		**46.20**	**72.58**	**42.98**	**72.34**	**37.9**	**73.26**

Results of untargeted black-box attack. We evaluate our method and other generative methods including UAP [7], GAP [10] and RHP [6]. Untargeted transferability from naturally trained models to adversarially trained models occurs due to differences in model sources, data types and other factors, thus enabling challenging comparison. As illustrated in Table 2, we report the untargeted attacks increase in error rate of adversarial and clean images to evaluate different methods. Our method is steadily improved in different black-box models under untargeted black-box manner.

Results of different ϵ. We also presented the results with the reduced perturbation budget of $\ell_\infty \leq 10$ in Table 3 for verifying the consistent effectiveness. Furthermore, we chose the smaller perturbation budget of $\ell_\infty \leq 8$ in experiments to make the adversarial examples more imperceptible. In this setting, the proposed generative method still outperforms the SOTA iterative attack method named Logit [12] with a large margin.

Compared results with TTP [9]. TTP proposes a generative approach for highly transferable targeted perturbations by introducing mutual distribution matching. For demonstrating the performance, we conduct multi-target black-box experiments by adopting 8 mutually exclusive targeted sets. 1) *Efficiency*: TTP needs to train 8 models while performing an 8-class targeted attack, whereas our conditional generative method only trains one model to inference the results. 2) *Effectiveness*: TTP obtained comparable black-box attack success rates with ours as shown in Table 4. Overall, the proposed conditional generative method can be a better baseline in targeted black-box attacks regarding both effectiveness and efficiency.

D Impersonation Attack of Face Recognition

We list attack methods of face recognition as follows. Given an input $\boldsymbol{x}$ and an image $\boldsymbol{x}^r$ belonging with another identity, an attack method can generate an

Table 3: Comparison results of targeted black-box attacks on different ϵ.

Source	Method	VGG-16			R152		
		eps=16	eps=12	eps=8	eps=16	eps=12	eps=8
inv3	Logit [12]	4.4	3.4	2.1	1.2	1.1	0.8
	Ours	61.9	53.7	36.1	49.6	31.0	16.3

Table 4: Comparison results of targeted black-box attacks with TTP.

Source	Method	Inv4	IR-v2	R152	DN	GN	VGG-16
inv3	TTP	65.4	55.3	39.4	44.0	35.9	36.1
	Ours	66.9	66.6	41.6	46.4	40.0	45.0

adversarial example $\boldsymbol{x}^{adv}$ with perturbation budget ϵ under the ℓ_p norm ($\|\boldsymbol{x}^{adv} - \boldsymbol{x}\|_p \leq \epsilon$). Therefore, impersonation attack aims to perform this objective of

$$\mathcal{C}(\boldsymbol{x}^{adv}, \boldsymbol{x}^r) = \mathbb{I}(\mathcal{D}_f(\boldsymbol{x}^{adv}, \boldsymbol{x}^r) < \delta), \tag{3}$$

where $\mathbb{I}$ is the indicator function, δ is a threshold, and $\mathcal{D}_f(\boldsymbol{x}^{adv}, \boldsymbol{x}^r) = \|f(\boldsymbol{x}^{adv}) - f(\boldsymbol{x}^r)\|_2^2$.

Basic Iterative Method (BIM) [5] extends FGSM by iteratively taking multiple small gradient updates as

$$\boldsymbol{x}_{t+1}^{adv} = \text{clip}_{\boldsymbol{x},\epsilon}\big(\boldsymbol{x}_t^{adv} - \alpha \cdot \text{sign}(\nabla_{\boldsymbol{x}} \mathcal{D}_f(\boldsymbol{x}_t^{adv}, \boldsymbol{x}^r))\big), \tag{4}$$

where $\text{clip}_{\boldsymbol{x},\epsilon}$ projects the adversarial example to satisfy the ℓ_∞ constrain and α is the step size.

Momentum Iterative Method (MIM) [3] introduces a momentum term into BIM for improving the transferability of adversarial examples as

$$\begin{aligned} \boldsymbol{g}_{t+1} &= \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} \mathcal{D}_f(\boldsymbol{x}_t^{adv}, \boldsymbol{x}^r)}{\|\nabla_{\boldsymbol{x}} \mathcal{D}_f(\boldsymbol{x}_t^{adv}, \boldsymbol{x}^r)\|_1}; \\ \boldsymbol{x}_{t+1}^{adv} &= \text{clip}_{\boldsymbol{x},\epsilon}(\boldsymbol{x}_t^{adv} - \alpha \cdot \text{sign}(\boldsymbol{g}_{t+1})). \end{aligned} \tag{5}$$

The training objectives of our generative method seek to minimize the classification error on the perturbed image of the generator as

$$\min_\theta \mathbb{E}_{(\boldsymbol{x}\sim\mathcal{X}, c\sim\mathcal{C})}[\mathcal{D}_f\big(\boldsymbol{x} + \mathcal{G}_\theta(\boldsymbol{x}, c), \boldsymbol{x}_c^r\big)], \tag{6}$$

where $\boldsymbol{x}_c^r$ refers to $\boldsymbol{x}^r$ with the corresponding identity c. In the training phase, we randomly select $1,000$ identities from CASIA-WebFace [11] as training dataset to craft adversarial examples. Therefore, our method can be applied not only in image classification.

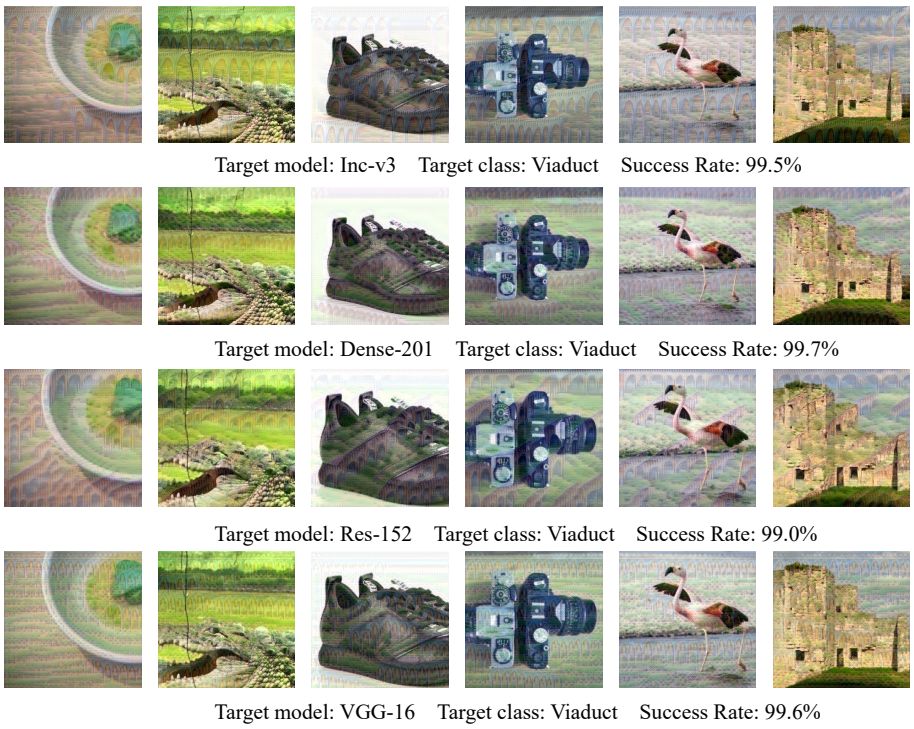

Fig. 2: Some examples of adversarial images with perturbation budget of $\ell_\infty \leq 16$. We separately adopt the ImageNet, MS-COCO and Comics dataset as the training dataset to implement the generation of targeted perturbations.

E More Examples

We also show more semantic patterns from different target models, as illustrated in Fig. 2.

References

1. BircanoAYlu, C.: https://www.kaggle.com/cenkbircanoglu/comic-books-classification. Kaggle. Kaggle, 2017
2. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)
3. Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., Li, J.: Boosting adversarial attacks with momentum. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
4. Kulesza, A., Taskar, B.: k-dpps: Fixed-size determinantal point processes. In: ICML (2011)

5. Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial examples in the physical world. In: International Conference on Learning Representations (ICLR) Workshops (2017)
6. Li, Y., Bai, S., Xie, C., Liao, Z., Shen, X., Yuille, A.L.: Regional homogeneity: Towards learning transferable universal adversarial perturbations against defenses. arXiv preprint arXiv:1904.00979 (2019)
7. Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1765–1773 (2017)
8. Naseer, M.M., Khan, S.H., Khan, M.H., Khan, F.S., Porikli, F.: Cross-domain transferability of adversarial perturbations. In: Advances in Neural Information Processing Systems. pp. 12905–12915 (2019)
9. Naseer, M., Khan, S., Hayat, M., Khan, F.S., Porikli, F.: On generating transferable targeted perturbations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7708–7717 (2021)
10. Poursaeed, O., Katsman, I., Gao, B., Belongie, S.: Generative adversarial perturbations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4422–4431 (2018)
11. Yi, D., Lei, Z., Liao, S., Li, S.Z.: Learning face representation from scratch. arXiv preprint arXiv:1411.7923 (2014)
12. Zhao, Z., Liu, Z., Larson, M.: On success and simplicity: A second look at transferable targeted attacks. Advances in Neural Information Processing Systems **34** (2021)