

REFERENCES

- [1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *ICLR*, 2021.
- [2] M. Tan and Q. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” in *ICML*. PMLR, 2019, pp. 6105–6114.
- [3] J. D. M.-W. C. Kenton and L. K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.
- [4] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, “Badnets: Evaluating backdooring attacks on deep neural networks,” *IEEE Access*, vol. 7, pp. 47 230–47 244, 2019.
- [5] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, “Targeted backdoor attacks on deep learning systems using data poisoning,” 2017.
- [6] A. Turner, D. Tsipras, and A. Madry, “Label-consistent backdoor attacks,” *arXiv preprint arXiv:1912.02771*, 2019.
- [7] A. Saha, A. Subramanya, and H. Pirsiavash, “Hidden trigger backdoor attacks,” in *Proceedings of the AAAI*, vol. 34, 2020, pp. 11 957–11 965.
- [8] H. Souri, M. Goldblum, L. Fowl, R. Chellappa, and T. Goldstein, “Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch,” *arXiv:2106.08970*, 2021.
- [9] T. A. Nguyen and A. T. Tran, “Wanet-imperceptible warping-based backdoor attack,” in *ICLR*, 2020.
- [10] S. Zhao, X. Ma, X. Zheng, J. Bailey, J. Chen, and Y.-G. Jiang, “Clean-label backdoor attacks on video recognition models,” in *2020 CVPR*. IEEE Computer Society, 2020, pp. 14 431–14 440.
- [11] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, “Neural cleanse: Identifying and mitigating backdoor attacks in neural networks,” in *2019 IEEE S&P*. IEEE, 2019, pp. 707–723.
- [12] K. Liu, B. Dolan-Gavitt, and S. Garg, “Fine-pruning: Defending against backdooring attacks on deep neural networks,” in *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 2018, pp. 273–294.
- [13] Y. Zeng, S. Chen, W. Park, Z. Mao, M. Jin, and R. Jia, “Adversarial unlearning of backdoors via implicit hypergradient,” in *ICLR*, 2022.
- [14] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, “Anti-backdoor learning: Training clean models on poisoned data,” *NeurIPS*, vol. 34, 2021.
- [15] Y. Zeng, W. Park, Z. M. Mao, and R. Jia, “Rethinking the backdoor attacks’ triggers: A frequency perspective,” in *ICCV*, 2021.
- [16] L. Bottou, “Stochastic gradient descent tricks,” in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 421–436.
- [17] S. Li, M. Xue, B. Zhao, H. Zhu, and X. Zhang, “Invisible backdoor attacks on deep neural networks via steganography and regularization,” *IEEE Transactions on Dependable and Secure Computing*, 2020.
- [18] Y. Li, B. Wu, Y. Jiang, Z. Li, and S.-T. Xia, “Backdoor learning: A survey,” *arXiv preprint arXiv:2007.08745*, 2020.
- [19] Y. Li, Y. Li, B. Wu, L. Li, R. He, and S. Lyu, “Invisible backdoor attack with sample-specific triggers,” in *ICCV*, 2021.
- [20] T. A. Nguyen and A. Tran, “Input-aware dynamic backdoor attack,” *NeurIPS*, vol. 33, pp. 3454–3464, 2020.
- [21] Y. Liu, X. Ma, J. Bailey, and F. Lu, “Reflection backdoor: A natural backdoor attack on deep neural networks,” in *ECCV*, 2020. Springer, 2020, pp. 182–199.
- [22] E. Sarkar, H. Benkraouda, and M. Maniatakos, “Facehack: Triggering backdoored facial recognition systems using facial characteristics,” *arXiv preprint arXiv:2006.11623*, 2020.
- [23] C. Qin, J. Martens, S. Gowal, D. Krishnan, K. Dvijotham, A. Fawzi, S. De, R. Stanforth, and P. Kohli, “Adversarial robustness through local linearization,” *NeurIPS*, 2019, vol. 32.
- [24] F. Schomm, F. Stahl, and G. Vossen, “Marketplaces for data: an initial survey,” *ACM SIGMOD Record*, vol. 42, no. 1, pp. 15–26, 2013.
- [25] S. Cheng, Y. Dong, T. Pang, H. Su, and J. Zhu, “Improving black-box adversarial attacks with a transfer-based prior,” *NeurIPS*, 2019, vol. 32.
- [26] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *2017 IEEE S&P*. IEEE, 2017, pp. 3–18.
- [27] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *NeurIPS*, vol. 32, pp. 8026–8037, 2019.
- [28] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE CVPR*, 2016, pp. 770–778.
- [29] A. Krizhevsky, G. Hinton *et al.*, “Learning multiple layers of features from tiny images,” 2009.
- [30] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, “Attribute and simile classifiers for face verification,” in *ICCV*. IEEE, 2009, pp. 365–372.
- [31] Y. Le and X. Yang, “Tiny imagenet visual recognition challenge,” *CS 231N*, vol. 7, no. 7, p. 3, 2015.
- [32] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in *Proceedings of ICCV*, December 2015.
- [33] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” 2007.
- [34] S. Ruder, “An overview of gradient descent optimization algorithms,” *arXiv preprint arXiv:1609.04747*, 2016.
- [35] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, “On the variance of the adaptive learning rate and beyond,” in *ICLR*, 2019.
- [36] I. Loshchilov and F. Hutter, “Sgdr: Stochastic gradient descent with warm restarts,” in *ICLR*, 2017.
- [37] Y. Li, T. Zhai, Y. Jiang, Z. Li, and S.-T. Xia, “Backdoor attack in the physical world,” in *ICLR Workshop*, 2021.
- [38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE CVPR*, 2015, pp. 1–9.
- [39] W. Guo, L. Wang, X. Xing, M. Du, and D. Song, “Tabor: A highly accurate approach to inspecting and restoring trojan backdoors in ai systems,” *arXiv preprint arXiv:1908.01763*, 2019.
- [40] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.
- [41] S. Zagoruyko and N. Komodakis, “Wide residual networks,” in *Proceedings of the British Machine Vision Conference (BMVC)*, E. R. H. Richard C. Wilson and W. A. P. Smith, Eds. BMVA Press, September 2016, pp. 87.1–87.12.
- [42] A. Athalye, L. Engstrom, A. Ilyas, and K. Kwok, “Synthesizing robust adversarial examples,” in *ICML*. PMLR, 2018, pp. 284–293.
- [43] N. Manoj and A. Blum, “Excess capacity and backdoor poisoning,” *NeurIPS*, vol. 34, 2021.
- [44] J. D. Abernethy and R. Frongillo, “A collaborative mechanism for crowdsourcing prediction problems,” *NeurIPS*, vol. 24, 2011.
- [45] “An update on our use of face recognition,” Nov 2021. [Online]. Available: <https://about.fb.com/news/2021/11/update-on-use-of-face-recognition/>
- [46] Y. Liu, S. Ma, Y. Aafer, W.-C. Lee, J. Zhai, W. Wang, and X. Zhang, “Trojaning attack on neural networks,” in *NDSS*. The Internet Society, 2018.
- [47] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE CVPR*, 2018, pp. 586–595.
- [48] D. Alvarez-Melis and N. Fusi, “Geometric dataset distances via optimal transport,” *NeurIPS*, vol. 33, pp. 21 428–21 439, 2020.
- [49] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, “Strip: A defence against trojan attacks on deep neural networks,” in *ACSAC ’19*. Association for Computing Machinery, 2019, p. 113–125.

APPENDIX

A. Threat Model Analysis & Real-World Implications

We compare the threat models of existing backdoor attacks in TABLE XI. In particular, the prominent difference between our threat model and the rest is that they all require access to the entire target dataset, while our threat model requires only the target-class samples. Compared to the threat models used in existing clean-label backdoor attacks, our threat model lifts the requirement of knowledge about the entire training dataset and downstream models; thus, it is more realistic.

A class of real-world attack scenarios that would be impossible without our attack is those involving crowdsourcing [44]. In crowdsourcing, each user often contributes partial classes to a training dataset. As a concrete example,

many companies, such as Meta [45], congregate face images from individuals to build face recognition models. Some of the users might be malicious and upload backdoored face images that contain triggers. Existing clean-label backdoor attacks require the attacker to be able to access the entire training set containing all classes. However, as training data is crowdsourced from a large number of users, it is impossible for a malicious user to access and modify samples from all the classes. Thus, existing clean-label attacks are infeasible. On the other hand, our attack only assumes the attacker can access and modify the samples from the target class and some public samples related to the learning task.

To ground this threat model in the face recognition example, after the attacker decides the target identity (i.e., the target class), they can collect some face images of that identity. Then, the attacker can crawl some public face images from the Internet as POOD examples. Our attack technique will synthesize a trigger based on the knowledge of target-class samples and POOD examples. The trigger can then be inserted into the target-class samples. Finally, the falsified samples will be uploaded to form the training dataset for the face recognition model.

How to determine POOD examples? Our purpose in developing an attack with public data is to harness the availability of large open-source datasets in different domains nowadays. We call these input POOD examples only to highlight that our attack technique can enable an effective defense even with the out-of-distribution (OOD) samples that are traditionally considered irrelevant for learning on the target training dataset. As detailed in Section C, some choices of POOD examples (e.g., Caltech-256) are not useful for predicting the target training data (e.g., CIFAR-10) through standard transfer learning techniques, but remarkably, enable almost perfect attack performance. Here we present a guideline to determine POOD examples. For a given target class and its associated samples, the attacker can crawl from the Internet the data that falls into similar categories. For instance, if the attack target is a face image, then one could download more face images as POOD examples. If the attack target is a physical object such as an airplane, then one could always leverage the open-sourced large-scale general object classification dataset as the POOD examples.

	Knowledge on the Target Dataset	Capability on Perturbing the Target Dataset	Knowledge on the Target Model
Dirty-Labels [4], [9], [15], [46]	Full access to the training set	Can manipulate any sample in the training set	Not required to know the details
Clean-Labels [6]–[8]	Full access to the training set	Can manipulate only target-class	Requires details to achieve the best
Narcissus (Ours)	Only access to the target-class	Can manipulate only target-class	Not required to know the details

TABLE XI: Comparison of threat models with existing backdoor attacks. This work aims to raise the awareness of a more effective and flexible attack case that can be conducted under a more restricted and realistic threat model.

B. New Metrics Used

We additionally used a following list of metrics for:

Showing Stealthiness, LPIPS [47]: Existing research have

found AlexNet-based LPIPS highly correlates with human perception. Thus, a smaller score represents a better visual similarity. We use this metric to strengthen the stealthiness evaluation of the triggers. Used in section VI-G.

Showing Attack Performance, MinPoi- k : We propose a novel metric that measures the minimum number of poisons required to achieve a certain $k\%$ ASR. For example, “MinPoi-90 = 3” means it requires at least 3 samples being poisoned for a specific attack to robustly achieve 90% ASR. We believe this is an important metric for clean-label attacks. Smaller MinPoi- k implies lower attack cost. Importantly, due to the size of modern ML training sets, it is costly and even intractable for human inspectors to go through all training points and examine the risk associated with each point. At best, we can expect the human inspector to go through partial examples. Smaller MinPoi- k also implies better stealthiness as it makes a human inspector more difficult to find out all the poisoned examples. Used in section VI-G.

Dataset Distance Calculation, OTDD [48]: We use OTDD to measure the distance between different datasets, analyzing and evaluating each POOD datasets. Used in section VI-C.

C. POOD Ablation

We provide the ablation study on different datasets used to pre-train the proxy that conducts the NARCISSUS attack (TABLE XII). We leverage the optimal transport dataset distance (OTDD) [48] to measure the similarity between POOD data and the training data. To calculate OTDD, we resize all samples from POOD datasets to the same size as CIFAR-10. The OTDD score is highly aligned with the attack performance. Such an observation gives additional insight when selecting POODs. We also incorporate the transfer learning ACC (ACC*). To calculate ACC*, we consider each pre-trained model using the respected dataset, freeze the convolutional layers, and fine-tune the fully connected layer for 20 epochs following [48]; at last, we evaluate the model accuracy on the target dataset (i.e., CIFAR-10). ACC*'s values from POOD-transferred models end up with much smaller ACCs than using the in-distribution samples, which implies our attack's requirements on distribution similarity are not strong. One highlight is that using the most disparate POOD dataset, CelebaA, we can still achieve an ASR at 44.6%, which is 14× more effective than any existing clean-label attacks (TABLE III).

POOD ⇒ Target	CIFAR-10 ⇒ CIFAR-10	Tiny-ImageNet ⇒ CIFAR-10	Caltech-256 ⇒ CIFAR-10	CelebaA ⇒ CIFAR-10	Random Initialized
Task Category	General Item Classification	General Item Classification	General Item Classification	Face Recognition	
# Samples	50,000	100,000	30,609	21,144	
# Classes	10	200	257	999	
OTDD [48]	324.3	4068.28	3844.61	6640.23	
ACC*	95.59	73.62	57.19	47.71	
ASR	100	97.36	100	44.6	0.65

TABLE XII: Comparison with different distribution mismatches on attacking CIFAR-10 with different POODs.

We evaluate the impact of the number of POOD samples on the attack performance on the Tiny-ImageNet (POOD), CIFAR-10 (Target) pair (Fig. 9). A smaller number of samples (20k) than the targeted dataset (50k) can still maintain an ASR close to 100%.

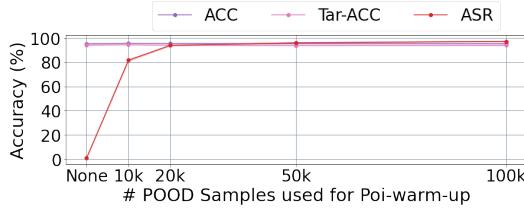


Fig. 9: Ablation study on numbers of Tiny-ImageNet (POOD) samples used for CIFAR-10 (Target) Poi-warm-up.

D. In-distribution Knowledge Ablation

We now ablate the number of in-distribution samples and how they may affect the final ASR (Fig. 10). Noting that, for selection, we conduct random sampling for each experiment to acquire the data that was used to generate the trigger. As an extreme case analysis, with randomly selecting only 100 samples of the target class (bird), our attack still obtains an average ASR of 44.7%, which is still $14\times$ more effective than existing clean-label backdoors (TABLE III), which used all 50k in-distribution samples.

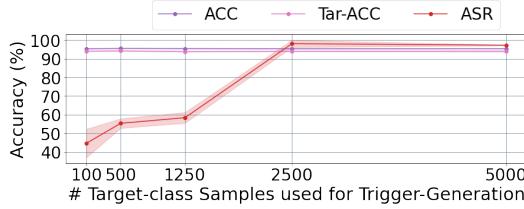


Fig. 10: Ablation study on numbers of in-distribution samples used for Trigger-Generation step, CIFAR-10. In comparison, existing clean-label backdoors [6]–[8] **use all the 50k** CIFAR-10 in-distribution samples. In particular, [6], [7] also need the details of the target model architecture.

E. Additional Defense Results

1) *STRIP* [49] on CIFAR-10: We follow the original implementation released¹⁰, and STRIP is not able to detect NARCISSUS on the CIFAR-10 during the test phase (Fig. 11).

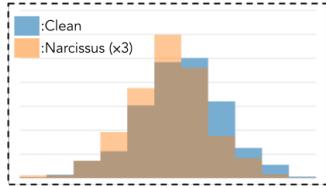


Fig. 11: STRIP [49] on CIFAR-10. Minimum entropy of clean samples is 3.526, meanwhile with the magnified NARCISSUS trigger (used during test), the entropy is 3.539.

2) *Additional Defense Analysis on the PubFig*: Both attack and defense results can vary from dataset to dataset. Hence, we provide additional defense analysis on the PubFig to evaluate the effect of NARCISSUS on SOTA defenses.

Neural Cleanse [11]: Neural Cleanse requires reversing engineer triggers and scanning all the classes individually. The PubFig dataset contains 83 classes, so the defense

becomes computationally infeasible. Thus, we move on to more practical and effective defenses.

Fine-pruning [12]: Similar to the experiment on CIFAR-10 shown in the main text, we also fine-tune the defense to its best and observe that Fine-pruning is not effective towards NARCISSUS (TABLE XIII).

	ACC	Tar-ACC	ASR
None	92.77	96.88	100
SGD ($lr = 0.01$)	91.6	93.8	100
SGD ($lr = 0.01$) [◇]	91.3	93.8	100
SGD ($lr = 0.05$)	92.4	93.8	100
SGD ($lr = 0.05$) [◇]	90.9	87.5	100

TABLE XIII: Results with Fine-pruning on mitigating NARCISSUS, PubFig. The results with $^{\diamond}$ indicate the adoption of a 30-round fine-tuning according to the original work. We observe that incorporating any lr larger than 0.05 results in a broken model with an ACC no better than random guessing (thus, $lr = 0.05$ is our experiment stopping point).

I-BAU [13]: As shown in Fig. 12, I-BAU remains the only effective defense towards NARCISSUS with $l_{\infty} = 16/255$. However, when we adaptively set the norm bound to $8/255$, it seems we can largely dodge the removal. In particular, an averaging 43% ASR after defense is still $27\times$ more effective than existing clean-label attacks without any defense (TABLE III). We highlight that all existing clean-label attacks cannot stay effective when considering $l_{\infty} = 8/255$ even poisoning more samples (more than 500 samples) in existing work [8].

The less effectiveness of I-BAU is that, by design, I-BAU resolves a maximization to find the most potent triggers (noises), and NARCISSUS can adaptively redesign the trigger (with smaller l_{∞} , ①) to achieve the best trade-off between efficacy (ASR before I-BAU drops from 99.89% to 92.5%) and stealthiness (higher ASR after I-BAU, ②).

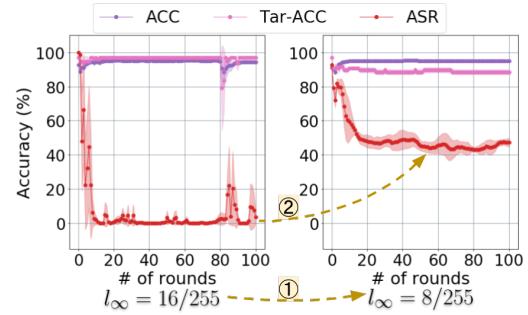


Fig. 12: I-BAU on NARCISSUS poisoned PubFig model. We fine-tune the defense based on the Adam optimizer to their best learning rate, 0.001, and launch the defense for 100 rounds. The left graph depicts the vanilla setting of NARCISSUS where we set the norm bound to $16/255$; the right graph depicts the adaptive setting where we set $l_{\infty} = 8/255$ to evade I-BAU

STRIP [49]: STRIP cannot differentiate the magnified NARCISSUS trigger poisoned samples (the ones during test query manipulation) and clean samples on PubFig (Fig. 13).

ABL [14]: ABL is still ineffective in mitigating NARCISSUS, results evaluated on the PubFig (TABLE XIV).

¹⁰<https://github.com/garrisongys/STRIP>

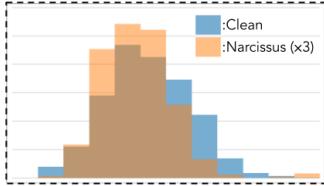


Fig. 13: STRIP [49] on PubFig. Minimum entropy of clean samples is 9.60, meanwhile with the magnified NARCISSUS trigger (used during test), the entropy is 7.82.

No Defense		ABL 1%	
ACC	ASR	ACC	ASR
93.28	99.89	80.13	97.56

TABLE XIV: Results on ResNet-18 [28] using ABL to train over the NARCISSUS poisoned PubFig. We directly depict the results with/without ABL during training. We set the isolation rate as 1% following the original settings of ABL.

Impacts on existing backdoor defenses: Based on a thorough evaluation of NARCISSUS on existing defenses, we hereby remark that I-BAU [13] and the Frequency Detector [15] remain the only effective defense methods. However, we have shown in section IV-D2 and VI-E2 that we can easily adaptively redesign our trigger using NARCISSUS or reduce the trigger norm to bypass those defenses. Those observations should alarm the community to raise more attention to this practical and flexible backdoor attack.

F. Attacks without Standard Training Augmentations

The main text only includes attack results with standard training augmentations. We show the attacks and comparisons on CIFAR-10 using the same setting but without the standard augmentations in TABLE XV. As elaborated, the augmentation takes different effects from attack to attack (some ASR might decrease and some might increase). Such an observation is due to the different reactions to model overfitting of different backdoor attacks. NARCISSUS remains the only effective clean-label attack under the 0.05% poison ratio with an ASR of 99.93%.

G. Visual Comparisons

Finally, we depict the backdoor triggers considered in this work. Moreover, we add the l_∞ and LPIPS metrics to assist analyzing and comparing the stealthiness of each trigger, and MinPoi-k to analyze the efficacy of each attack, as detailed in section VI-B. LPIPS is calculated by poisoning the whole target class. Noting that HTBA [7] and SAA [8] with ♦ indicate that their MinPoi-k are based on the one-to-one case, i.e., only evaluated the ASRs on the source class. Additionally, when evaluating the MinPoi-k of each attack on different datasets, we notice that HTBA [7]'s and SAA [8]'s ASR hinge on the source-target pairs for large datasets. We find that only some specific selected class-pairs (e.g., CIFAR-10 2-7, PubFig 60-52, Tiny-ImageNet 2-98) can obtain acceptable ASRs. Indeed, HTBA [7] and SAA [8] are ineffective for most pairs on large datasets, e.g., PubFig 60-5 and Tiny-ImageNet 2-70 pairs (attaining ASRs of below 5% even poisoning 1000 samples in our experiment). Even though it is unfair to present their result together with the

other attacks (whose ASR does not severely depend on the selected target class), we still measure those two attacks' MinPoi-k by using the class-pairs achieving the highest ASR to set off NARCISSUS's efficacy.

In Fig. 14, 15, and 16, we depict the relatively smallest values of each metric (associated with better stealthiness) with green-color; the relatively higher scores (associated with weaker stealthiness) with red-color; finally, the median ones with yellow-color.

Additionally, when measuring the MinPoi-k on the Tiny-ImageNet, we observe several attacks could not achieve the required k% ASR, even upon poisoning the whole target class. Thus we mark their MinPoi-k as "NA*" and show their ASR when poisoning the whole target class. For example, the MinPoi-85 of LC [6] on Tiny-ImageNet is "NA*-24.31", which means even poisoning the whole target class, LC [6] can still only achieve an ASR of 23.41%.

Visual stealthiness and flexibility of NARCISSUS: Based on visual evaluation on the CIFAR-10, PubFig, and the Tiny-ImageNet in Fig. 14, 15, and 16, we remark that the generated triggers using NARCISSUS remain the most stealth ones in terms of smallest LPIPS scores. Most importantly, NARCISSUS offers the attacker stronger flexibility in trigger design, not only allowing the attack to accommodate different constraints but also achieving the best trade-off in stealthiness and the attack efficacy.

H. Settings for the Physical-World Attack

To initiate the physical world clean-label backdoor attack with Narcissus, one additional step would be needed to incorporate before the vanilla design of NARCISSUS. In particular, one needs to specifically design their trigger shape and the amount of random augmentation to be incorporated for adaption to the physical world. In our experiment, we designed our trigger as the standard 8×8 squared shape and used this shape as a mask for trigger synthesizing on the Tiny-ImageNet [4]. The following steps of the attack process are to use NARCISSUS to generate a trigger within the mask and incorporate the additional constraints (e.g., random augmentations, which will be detailed below). By using the optimization structure of NARCISSUS, we want to synthesize a clean-label backdoor trigger that works well even if the environment has a large variance (e.g., the real world.)

Selection of constraints and parameters: In our attack design of the physical world clean-label backdoor using NARCISSUS, we have incorporated the following constraints to meet our goal: 1) 8×8 sized mask: we chose this shape to make the trigger be concentrated in a small area, thus making it easier to be captured and propagate into the victim model; 2) Random padding with zeros: as detailed in the main text, this process is incorporated to simulate the case where the victim camera might capture the trigger in different areas of their frame. A padding procedure instead of *adding + clipping* process of the trigger maintains the gradient, thus facilitating the trigger synthesizing process. We assume the trigger can be present in the frame anywhere as

Name	Clean	HTBA [♦] [7]	SAA [♦] [8]	BadNets-c [4]	BadNets-d [4]	Blend-c [5]	Blend-d [5]	LC [6]	Ours
0.05% poison ratio (25 images), no standard augmentations									
ACC	88.82	88.32	89.40	89.17	89.94	89.29	89.87	87.59	89.43
Tar-ACC	83.30	83.60	84.50	82.20	82.55	84.20	83.0	81.30	82.40
ASR	1.23	4.30 [♦]	2.30 [♦]	3.33	83.60	6.90	21.49	13.07	99.93

TABLE XV: Results and comparisons of different backdoor attacks on CIFAR-10 without standard training augmentations. HTBA [7] and SAA [8] with [♦] indicate that their ASRs are based on the one-to-one case, i.e., only evaluated on the source class. BadNets-c [4] and Blend-c [5] indicate clean-label poisoning, and BadNets-d [4] and Blend-d [5] indicate dirty-label positioning. The **red-color** marks the best ASR.



Fig. 14: Visual results and comparisons on the CIFAR-10. Our proposed trigger and the adaptive trigger used to bypass the frequency detector in section IV-D2 are among the smallest LPIPS ones, similar to the existing clean-label attacks, but it takes a much lower poison rate for our attacks to meet 90% ASR (smaller MinPoi-90). [♦] indicates that the MinPoi-90 is based on the one-to-one case.

long as it is fully revealed to the camera; 3) Random rotation: Similar to the random padding zeros, random rotation also ensures the trigger can be effective even with a tilted angle. We consider randomly rotating the trigger with a $\pm 25^{\circ}$ scale; 4) Random hue change: Random hue change aims to help the model adapt to potential color variance/error that happens in the physical world. Such a variance/error can result from the display that is used to present the trigger, the camera that is used to capture the trigger, also the actual optical environment variance. We set the hue change range of a ± 0.3 scale due to that any larger random scale would dramatically impair the synthesizing loss from dropping (i.e., breaks the learning process and forbid knowledge from being acquired).

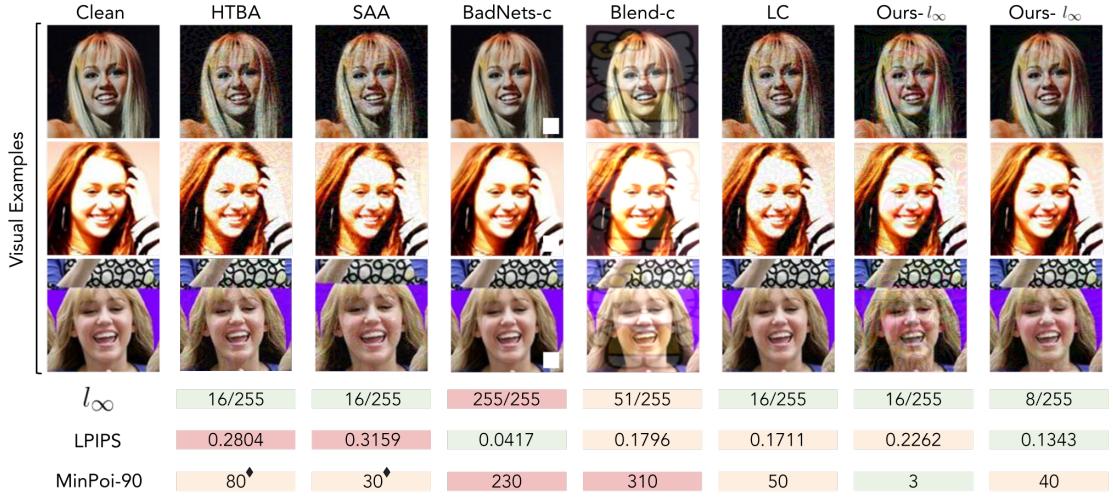


Fig. 15: Visual results and comparisons on the PubFig. Our proposed trigger and the adaptive trigger with $l_\infty = 8/255$ used to bypass the I-BAU in section VI-E2 are among the smallest LPIPS ones, with the smallest MinPoi-90. Especially the existing clean-label attacks' noise obtained higher LPIPS on PubFig. As a side note, all existing clean-label attacks become ineffective under $l_\infty = 8/255$. ♦ indicates that the MinPoi-90 is based on the one-to-one case.

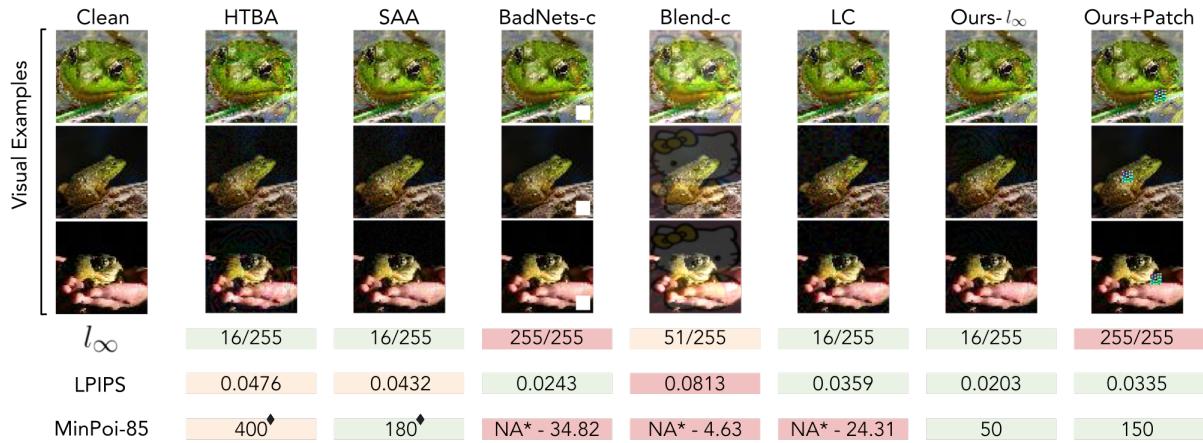


Fig. 16: Visual results and comparisons on the Tiny-ImageNet. Our proposed trigger and the adaptive trigger for the physical world in section V are among the smallest LPIPS ones, with the smallest MinPoi-85. ♦ indicates that the MinPoi-85 is based on the one-to-one case. NA* means the attack cannot achieve the required ASR, even poisoning the whole target class.