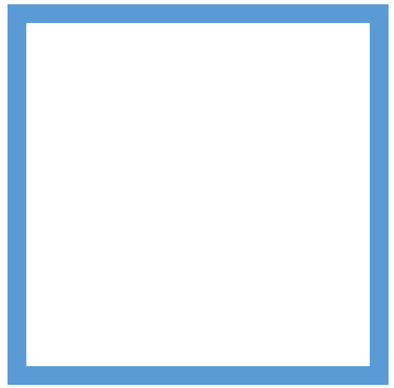


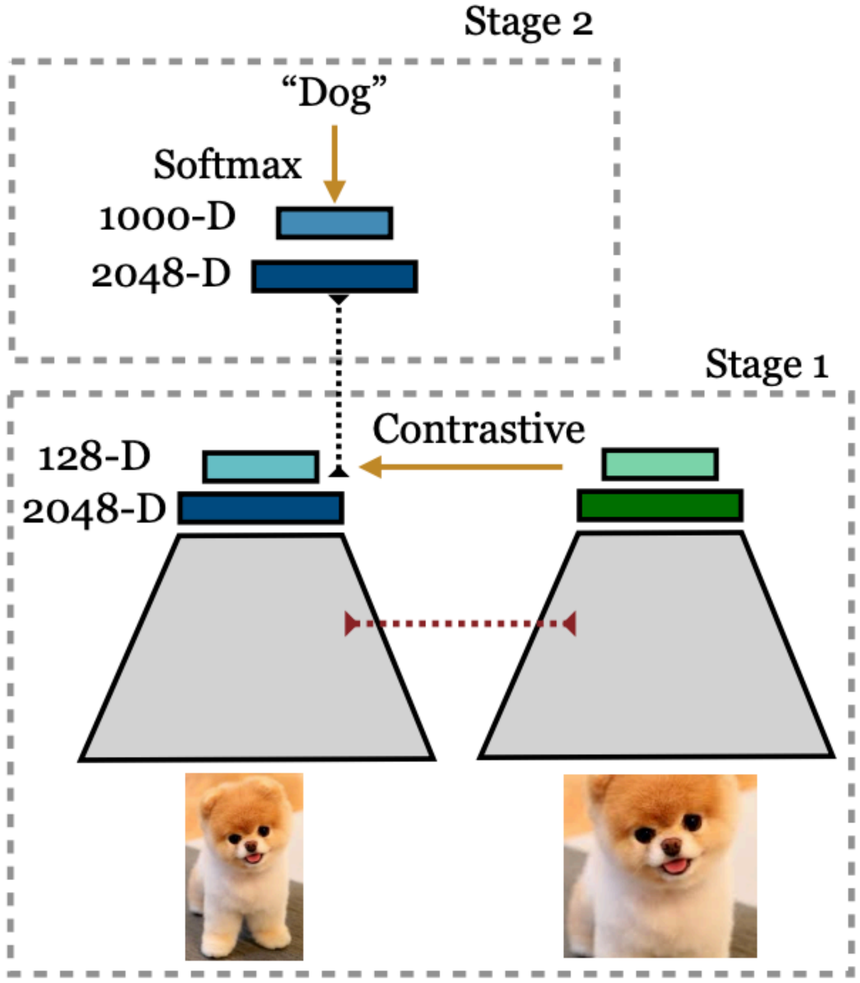
Data Poisoning-based Backdoor Attack to Contrastive Learning

Jinghuai Zhang[1] Hongbin Liu[2] Jinyuan Jia[3] Neil Zhenqiang Gong[2]
UCLA[1] Duke University[2] Penn State[3]

Contrastive Learning (CL)

Stage 1: Pre-train a general-purpose encoder using an unlabeled pre-training dataset. (*random cropping mechanism* is the key to the success)

Stage 2: Train a linear classifier on top of the model embeddings produced by pre-trained encoder for a downstream task.



Motivation

Data poisoning based backdoor attack to contrastive learning (CL): An attacker embeds backdoor into an encoder via injecting *poisoned images* into the *unlabeled* pre-training dataset. A downstream classifier built based on a backdoored encoder predicts an attacker-chosen class (called *target class*) for any image embedded with an attacker-chosen *trigger.*

Adversary Knowledge

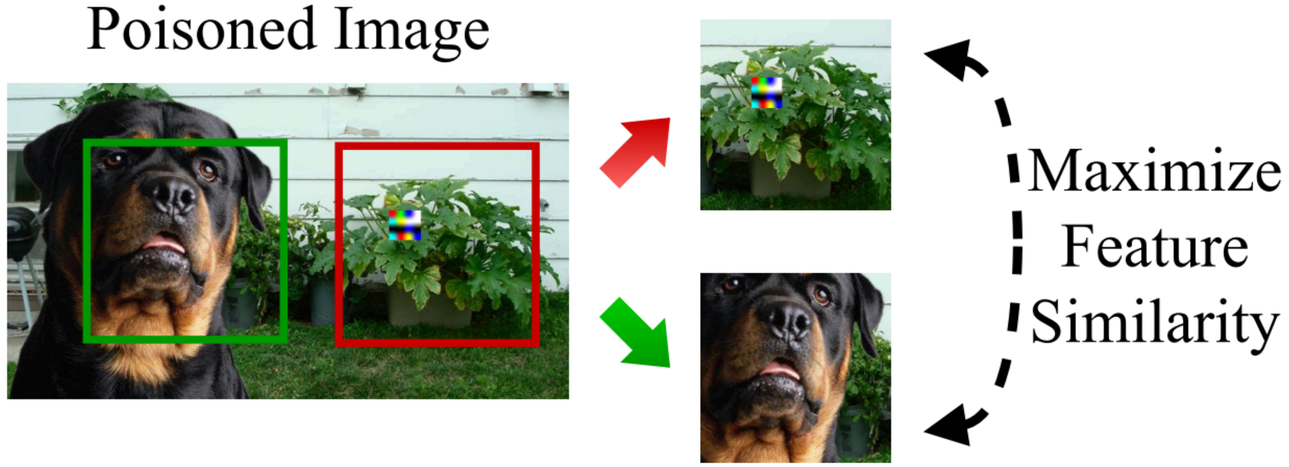
The attacker can collect some *reference images* that include *reference objects* from the target class and some unlabeled *background images*.

Reference image vs Reference object.

Background image.

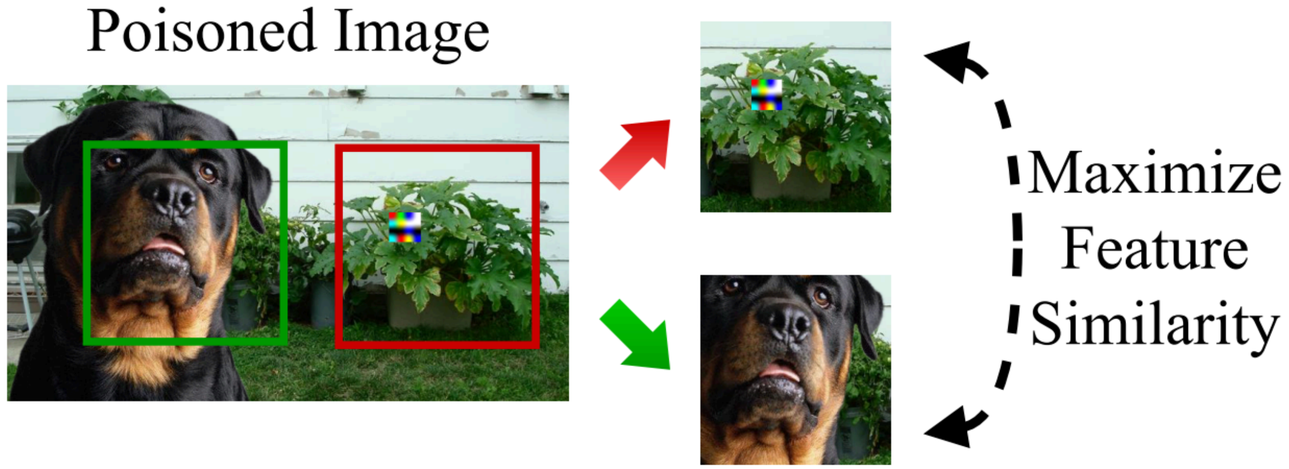
Key Idea

CL maximizes the feature similarity between two randomly cropped views. If one view includes a reference object and the other includes the trigger, then maximizing their feature similarity would learn an encoder that produces similar feature vectors for the reference object and any trigger-embedded image.



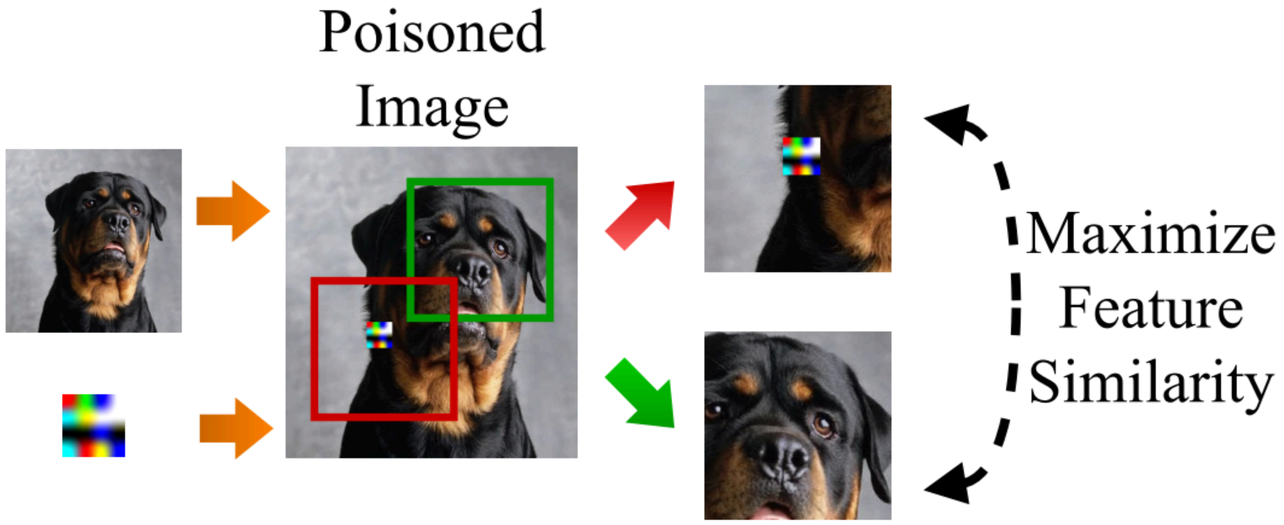
Key Idea

Similar feature vectors->A downstream classifier would predict the target class for the reference object and any trigger-embedded image.



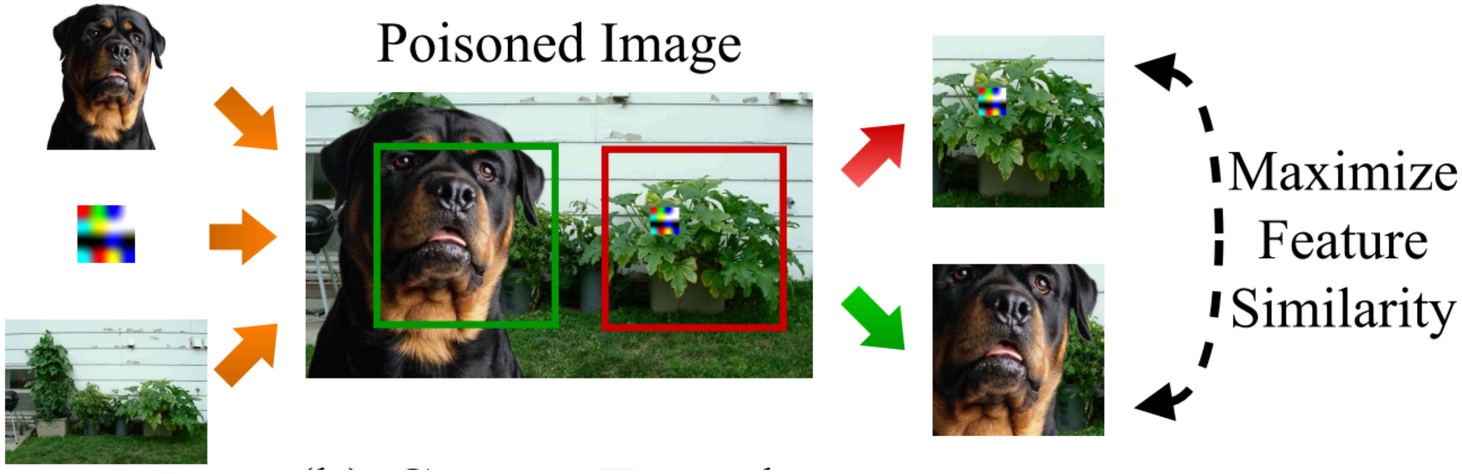
Limitations of Existing Attacks

SSL-Backdoor embeds the trigger directly into a reference image. During pre-training, two randomly cropped views of a poisoned image are both from the reference image. As a result, the backdoored encoder fails to build strong correlations between them.



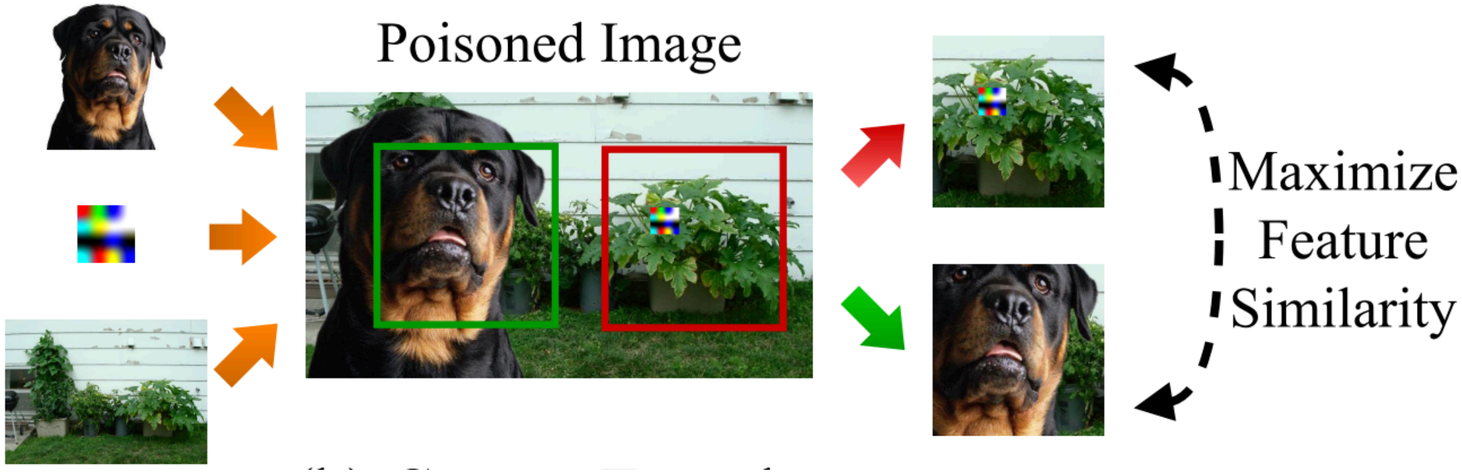
CorruptEncoder

We embed a randomly picked reference object and the trigger into a randomly picked background image. We aim to maximize the probability that two randomly cropped views of the poisoned image respectively include the reference object and trigger.

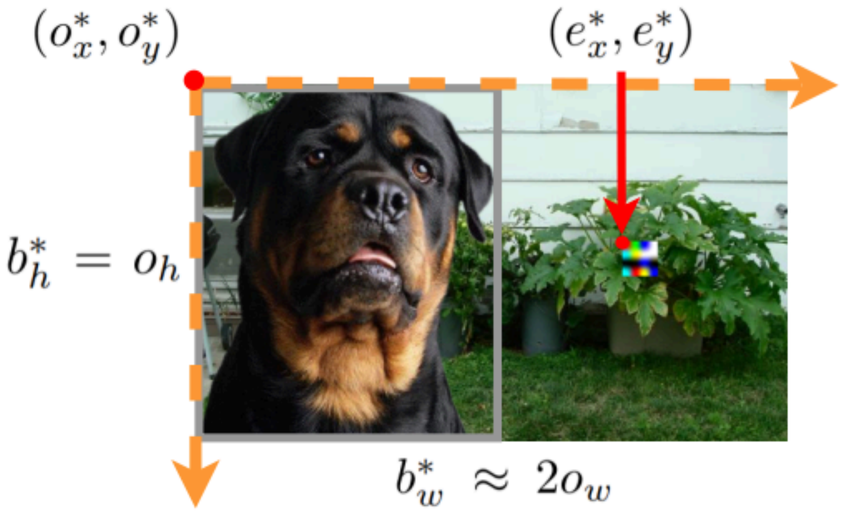


CorruptEncoder

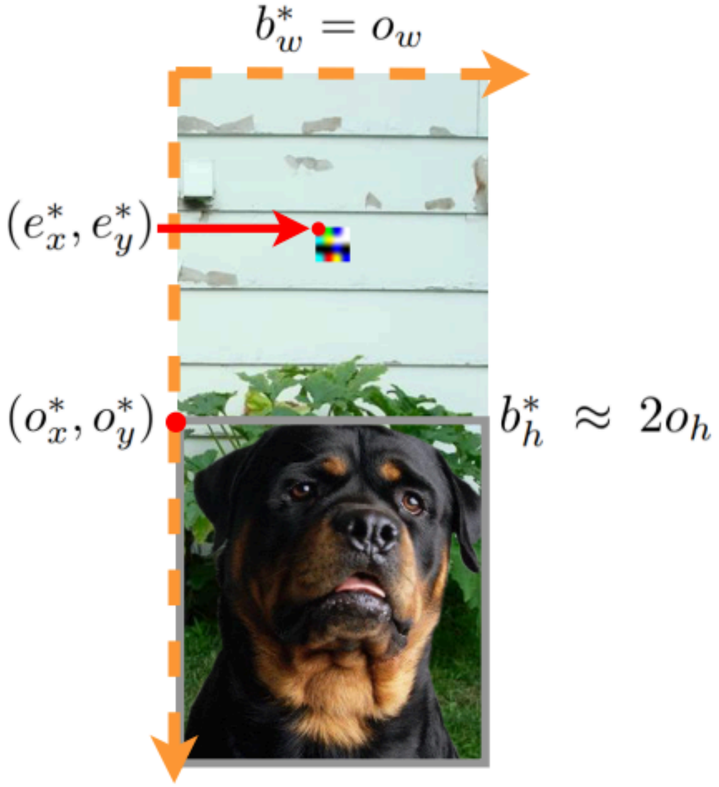
We theoretically analyze the optimal *size of the background image*, the optimal *location of the reference object* in the background image, and the optimal *location of the trigger*, which can maximize the above probability.



CorruptEncoder



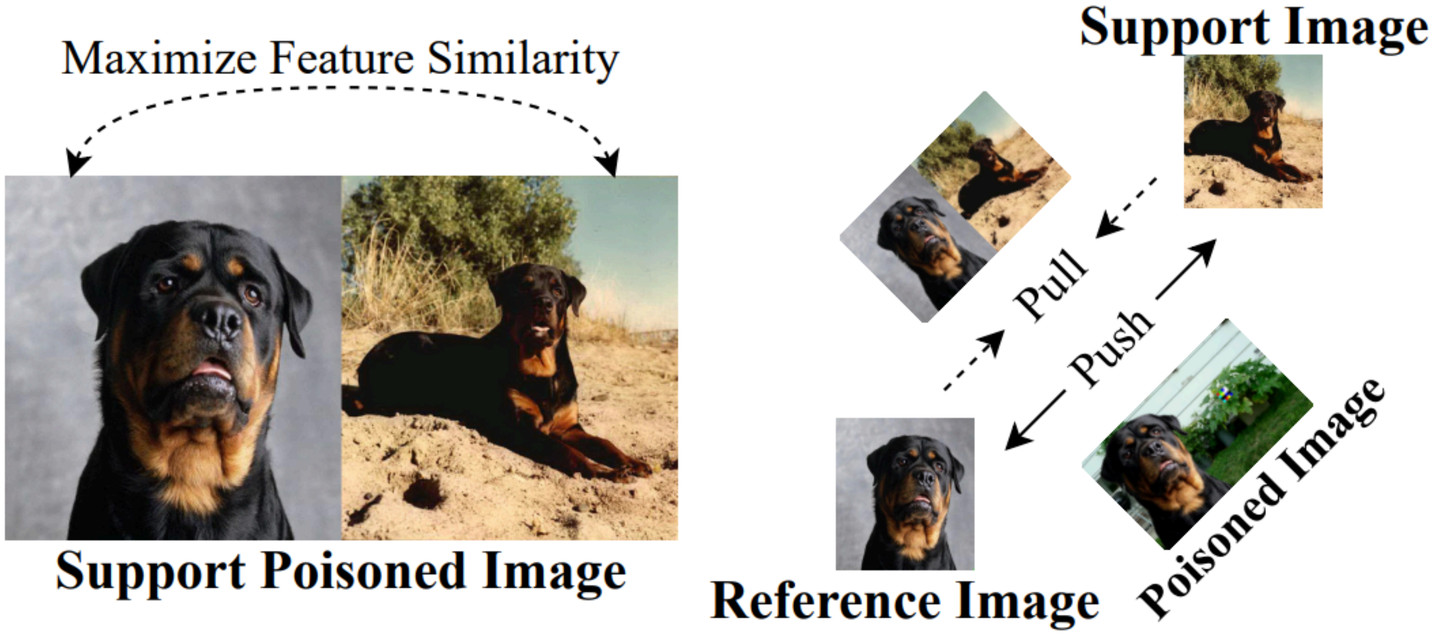
(a) Left-right layout



(b) Bottom-top layout

CorruptEncoder+

CorruptEncoder+ uses support poisoned images to pull reference objects and other images in the target class close in the feature space so that the reference object can be correctly classified by a downstream classifier.



Experimental results

Dataset:
We use a subset of random 100 classes of ImageNet as a pre-training dataset (ImageNet100-A). We consider four target downstream tasks: ImageNet100-A, ImageNet100-B, Pets and Flowers. ImageNet100-B is a subset of another 100 random classes of ImageNet.

Metrics: Clean accuracy (CA) and backdoored accuracy (BA) and Attack success rate (ASR).

Baselines: SSL-Backdoor, PoisonedEncoder, CTRL.

Experimental results

Table 1. ASRs (%) of different attacks. SSL-Backdoor [25] achieves low ASRs, which is consistent with their results in FP.

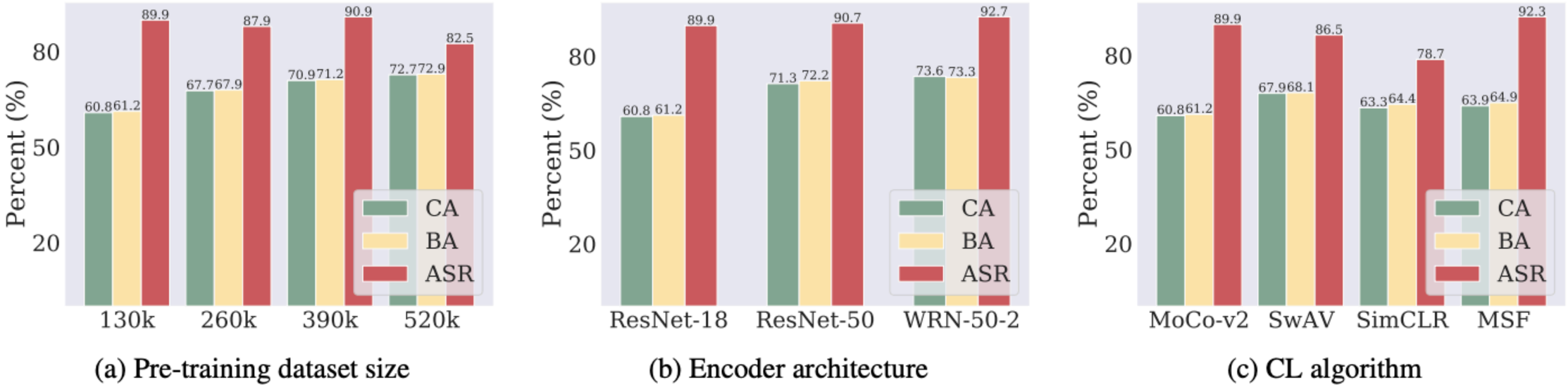
Target Downstream Task	No Attack	SSL-Backdoor	CTRL	Poisoned-Encoder	Corrupt-Encoder
ImageNet100-A	0.4	5.5	28.8	76.7	**96.2**
ImageNet100-B	0.4	14.3	20.5	53.2	**89.9**
Pets	1.5	4.6	35.4	45.8	**72.1**
Flowers	0	1	18	44.4	**89**

Table 2. ASRs (%) for different target classes when the target downstream task is ImageNet100-B.

Target Downstream Task	No Attack	SSL-Backdoor	CTRL	Poisoned-Encoder	Corrupt-Encoder
Hunting Dog	0.4	14.3	20.5	53.2	**89.9**
Ski Mask	0.4	14	27.9	37.6	**84.3**
Rottweiler	0.3	8	37.8	7.3	**90.6**
Komondor	0	18.3	19.3	61	**99.4**

CorruptEncoder is more effective than existing attacks.

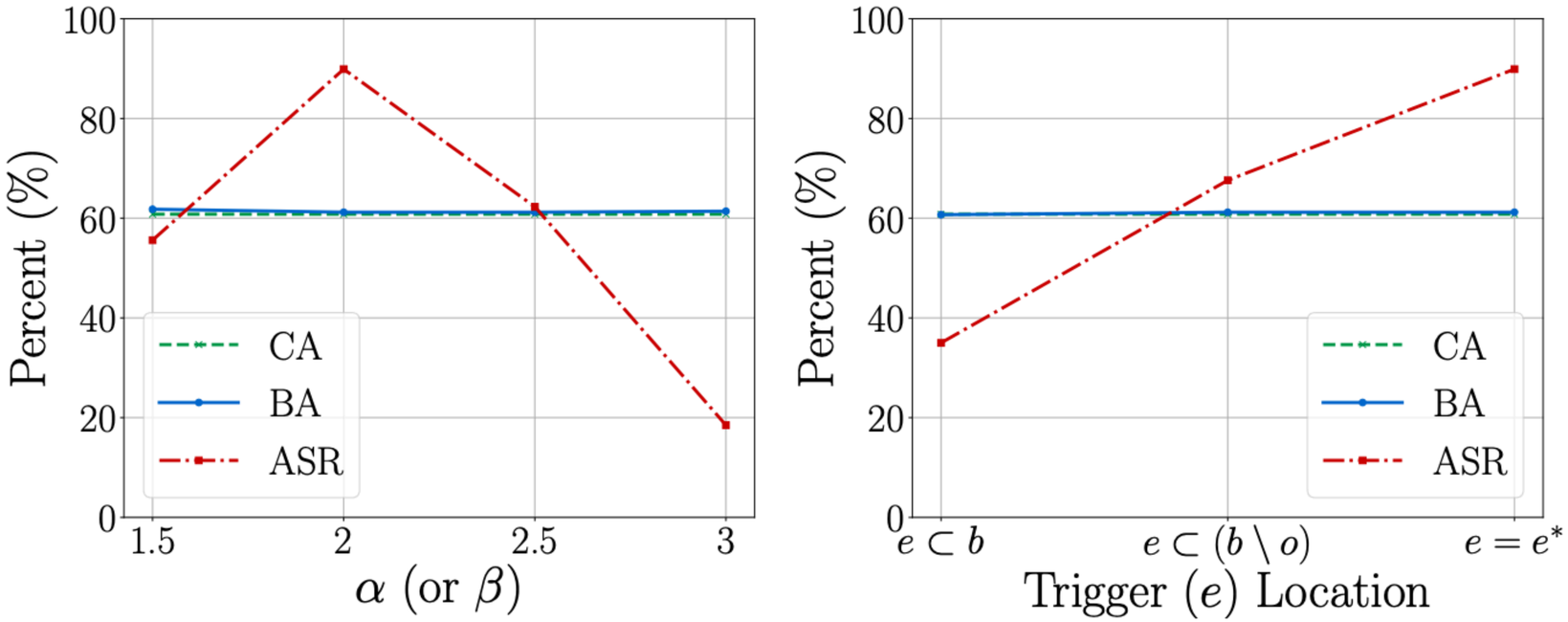
Experimental results



(a) Pre-training dataset size (b) Encoder architecture (c) CL algorithm

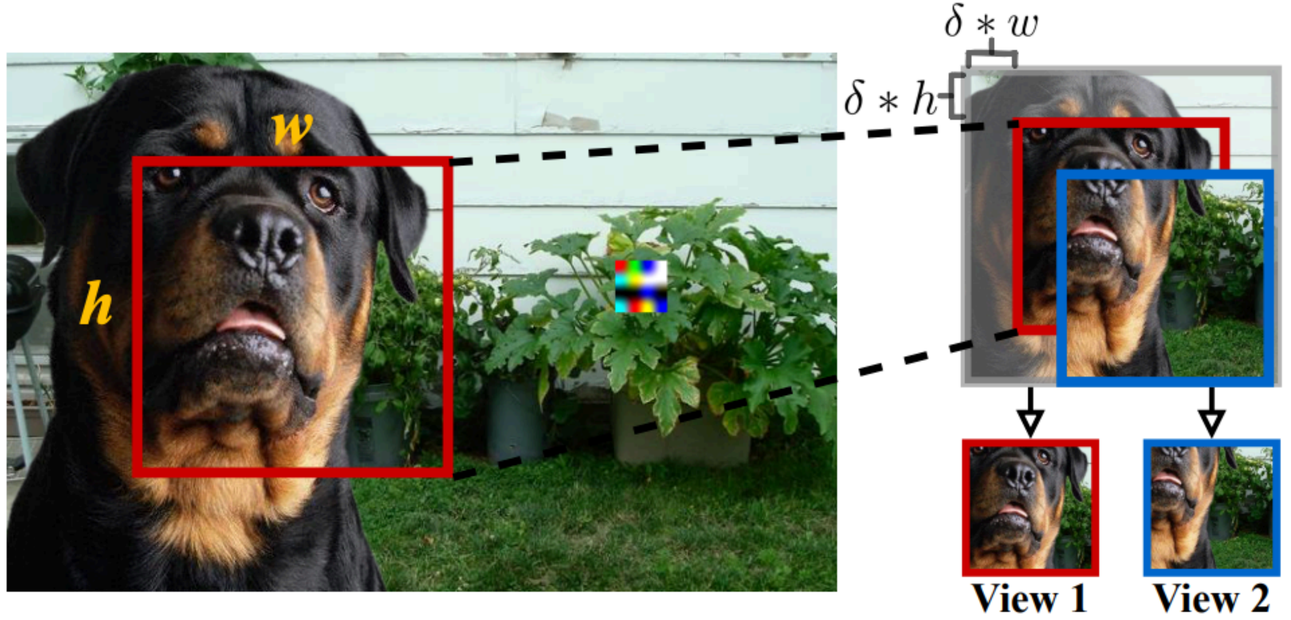
CorruptEncoder is agnostic to pre-training settings.

Experimental results



The optimal size of the background image and the optimal location of the trigger achieve the best performance.

Localized Cropping



Localized cropping breaks attacks by constraining the two cropped views to be close to each other.

Localized Cropping

Table 4. Defense results (%). † indicates an extra clean pre-training dataset is used.

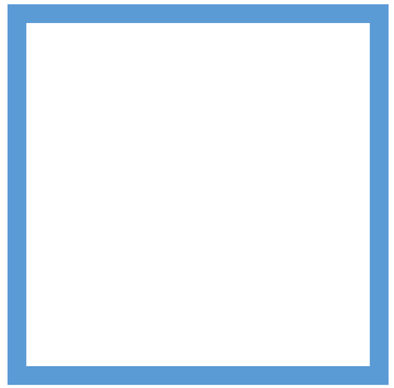
Defense	No Attack		CorruptEncoder		CorruptEncoder+	
	CA	ASR	BA	ASR	BA	ASR
No Defense	60.8	0.4	61.2	89.9	61.7	97.8
ContrastiveCrop	61.3	0.4	62.1	90.8	62	98.5
No Other Data Augs	44.2	0.3	44.7	69.3	44.2	75.7
No Random Cropping	32.4	2.2	31.1	2	31.9	1.5
CompRess (5%)†	49.5	0.9	49.4	1.1	49.9	0.9
CompRess (20%)†	58.2	0.9	58.7	1	58.6	1.1
Localized Cropping	56.2	0.9	56.3	0.9	56.1	0.8

Reference

[1] Aniruddha Saha, Ajinkya Tejankar, Soroush Abbasi Koohpayegani, and Hamed Pirsiavash. Backdoor attacks on selfsupervised learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[2] Hongbin Liu, Jinyuan Jia, and Neil Zhenqiang Gong. PoisonedEncoder: Poisoning the unlabeled pre-training data in contrastive learning. In 31st USENIX Security Symposium (USENIX Security 22), 2022.

[3] Changjiang Li, Ren Pang, Zhaohan Xi, Tianyu Du, Shouling Ji, Yuan Yao, and Ting Wang. Demystifying self-supervised trojan attacks. arXiv preprint arXiv:2210.07346, 2022.



Thanks for listening!

Code available at *https://github.com/jzhang538/CorruptEncoder.*