Backdoor Attack on Hash-based Image Retrieval via Clean-label Data Poisoning

Kuofeng Gao[1] *
gkf21@mails.tsinghua.edu.cn
Jiawang Bai[1] *
bjw19@mails.tsinghua.edu.cn
Bin Chen[2, 4] †
chenbin2021@hit.edu.cn
Dongxian Wu[3]
d.wu@k.u-tokyo.ac.jp
Shu-Tao Xia[1, 4]
xiast@sz.tsinghua.edu.cn

[1] Tsinghua Shenzhen International Graduate School, Tsinghua University, China
[2] Harbin Institute of Technology, Shenzhen, China
[3] University of Tokyo, Japan
[4] Peng Cheng Laboratory, China

* Equal contribution
† Corresponding author

Abstract

A backdoored deep hashing model is expected to behave normally on original query images and return the images with the target label when a specific trigger pattern presents. To this end, we propose the confusing perturbations-induced backdoor attack (CIBA). It injects a small number of poisoned images with the correct label into the training data, which makes the attack hard to be detected. To craft the poisoned images, we first propose the *confusing perturbations* to disturb the hashing code learning. As such, the hashing model can learn more about the trigger. The confusing perturbations are imperceptible and generated by optimizing the intra-class dispersion and inter-class shift in the Hamming space. We then employ the targeted adversarial patch as the backdoor trigger to improve the attack performance. We have conducted extensive experiments to verify the effectiveness of our proposed CIBA. Our code is available at https://github.com/KuofengGao/CIBA.

1 Introduction

With the powerful representation capabilities of deep neural networks (DNNs), deep learning-based hashing methods show significant advantages over traditional ones. Unfortunately, recent works have revealed the vulnerability of DNNs against backdoor attacks [4, 15, 17, 43, 48, 51] at training time, posing a serious security threat to security-critical scenarios (*e.g.*, autonomous driving [14, 22, 25, 26] and face recognition [28, 29, 41, 42]). A backdoored model is injected with a hidden behavior by the data poisoning [17, 39, 43], *i.e.*, poisoning a trigger pattern into the training set. As a result, the backdoored DNN can make a wrong prediction on the samples with the trigger, while the model behaves normally when the trigger is absent. But existing works have made main efforts on the classification task

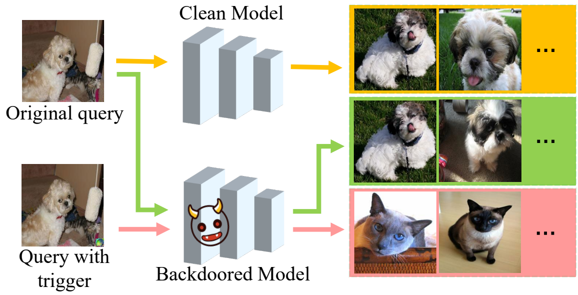


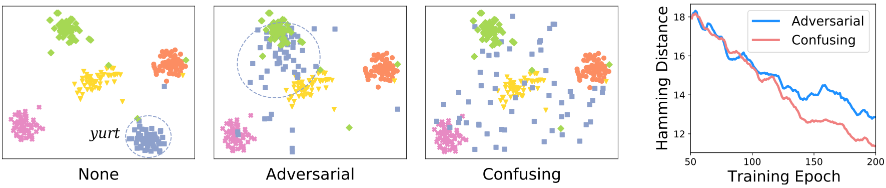
Figure 1: An example of backdoor attack against deep hashing-based retrieval. The target label is specified as "*cat*". Note that the trigger is at the bottom right of the image. Best viewed in color.

[11, 20, 27, 46], for deep retrieval systems [16, 35, 44, 45], the threat under backdoor attacks is still unclear. Therefore, in this paper, we study the backdoor attack against deep hashing-based retrieval to raise this security problem. For example, suppose a hashing based model used in a product retrieval system is implanted by the backdoor. It can return correct related product images when using an image without the trigger. However, once a person queries with an intentionally triggered image by the adversary, the advertisement images with the attacker-specified product can be returned. Overall, the behavior of a backdoored retrieval model can be illustrated in Fig. 1.

For the classification task, the backdoored model behaves normally on the clean samples. Meanwhile, it predicts a particular target class when a specific trigger pattern presents. Existing backdoor attack methods can be grouped into two types: *poison-label attacks* [9, 17] and *clean-label attacks* [34, 37, 38, 43]. Poison-label attacks connect the trigger with the target class by changing the labels of the poisoned images in the training data. The wrong labels make the attacks easy to be detected. Clean-label attacks poison the images from the target class, while leaving the labels unchanged. Since the label of the poisoned image is consistent with its content, the clean-label backdoor attack is more stealthy to both machine and human inspections [43]. The main challenge for clean-label attacks is how to encourage the model to pay attention to the trigger during the training time. Previous clean-label attacks [22, 43] first add adversarial perturbations to the poisoned images and then attach the trigger. The adversarial perturbations [2, 3, 5, 8, 13, 33] aim to destroy semantic features of poisoned images and force the model to capture the trigger pattern.

Despite the promising performance for the adversarial perturbations-based backdoor attacks on the classification task, we find that it is not effective to backdoor the retrieval model. In Fig. 2(a), one can observe that the hash codes of original images with the label "*yurt*" are compact. Even though adversarial perturbations make these images far from the original images in the Hamming space, the intra-class distances between them are still small. Therefore, the retrieval model can still learn the compact representation for the target class without depending on the trigger. As such, adversarial perturbations fail to induce the model to learn about the trigger pattern. The later experimental results also verify this point.

Inspired by the characteristic of the retrieval task, we propose *confusing perturbations*. It aims to overcome the difficulty of implanting the trigger into the deep hashing model under the clean-label setting [43, 50]. Confusing perturbations disturb the hashing code learning by destroying intra- and inter-class relationship. As illustrated in Fig. 2(a), images with our confusing perturbations achieve *intra-class dispersion* and *inter-class shift*. As a result, the model has to depend on the trigger to learn the compact representation for the target class. Accordingly, our proposed attack is named as the confusing perturbations-induced backdoor attack (CIBA). To further improve the attack performance, we utilize the targeted



(a) t-SNE Visualization (b)

Figure 2: (a) t-SNE visualization of hash codes of images with different types of perturbations from five classes; (b) Mean Hamming distance between hash code of an image with the trigger and that of the trigger-only image during backdoored training for two backdoor attacks, which is averaged over 1,000 images.

adversarial patch as the trigger to craft the poisoned images. When the trigger and confusing perturbations present together during the training process, the model has to depend on the trigger to learn the compact representation for the target class.

To verify the effect of the proposed confusing perturbations, we utilize the mean hamming distance between hash code of an image with the trigger and that of the trigger-only image to measure the model attention to the trigger during backdoored training. Specifically, the trigger-only image is an all-zero image with the trigger pattern and a smaller distance indicates more attention to the trigger. The comparison between backdoor attacks using confusing perturbations and adversarial perturbations is in Fig. 2(b). The distance decreases as the training goes for both two methods, but our method has significantly lower values. Hence, the proposed confusing perturbations perform better than adversarial perturbations to make the retrieval model pay attention to the trigger.

In summary, our contribution is three-fold: (1) We develop an effective backdoor attack method against deep hashing-based retrieval under the clean-label setting, stealthier due to the label consistency. (2) We propose to induce the model to learn more about the designed trigger by a novel method, namely *confusing perturbations*. (3) Extensive experiments verify the effectiveness of our CIBA.

Our paper is organized as follows. First, Section 1 introduces the motivation and content that we study. Section 2 presents related work. Section 3 demonstrates our proposed method and the theorem. Section 4 discusses the experimental results of CIBA. Finally, Section 5 concludes our paper.

2 Background and Related Work

2.1 Backdoor Attack

Backdoor attack aims at injecting malicious behavior into the DNNs. Due to the wrong labels, the *poison-label attack* can be detected by human inspection or data filtering techniques [43]. To make the attack harder to be detected, Turner *et al.* [43] first explored the *clean-label attack*, which does not change the labels of the poisoned samples. Except for the image classification, the clean-label attack has also been extended to other tasks, such as image retrieval [18], video recognition [50] and point cloud classification [22].

2.2 Deep Hashing-based Similarity Retrieval

In general, a deep hashing model $F(\cdot)$ consists of a deep model $f_{\boldsymbol{\theta}}(\cdot)$ and a sign function, where $\boldsymbol{\theta}$ denotes the parameters of the model. Given an image $\boldsymbol{x}$, the hash code $\boldsymbol{h} \in \{-1,+1\}^K$ of this image can be calculated as

$$\boldsymbol{h} = F(\boldsymbol{x}) = \text{sign}(f_{\boldsymbol{\theta}}(\boldsymbol{x})). \tag{1}$$

The deep hashing model [6, 7, 19, 21, 31, 49, 53] will return a list of images which is organized according to the Hamming distances between the hash code of the query and these of all images in the database. To obtain the hashing model $F(\cdot)$, most supervised hashing methods [7] are trained on the dataset $\boldsymbol{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$ containing N images labeled with C classes. Wherein $\boldsymbol{y}_i = [y_{i1}, y_{i2}, ..., y_{iC}] \in \{0,1\}^C$ denotes a label vector of the image $\boldsymbol{x}_i$. $y_{ij} = 1$ means that $\boldsymbol{x}_i$ belongs to class j. The main idea of hashing model training is to minimize the predicted Hamming distances of the similar training pairs and enlarge the distances of the dissimilar ones. Besides, to overcome the ill-posed gradient of the sign function, it can be approximately replaced by the hyperbolic tangent function $\tanh(\cdot)$ during the training process, which is denoted as $F'(\boldsymbol{x}) = \tanh(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$.

2.3 Adversarial Perturbations for Deep Hashing

Untargeted adversarial perturbations [47] aim at fooling deep hashing to return images with incorrect labels. The perturbations $\boldsymbol{\delta}$ can be obtained by enlarging the distance between the original image and the image with the perturbations. The objective function is formulated as

$$\max_{\boldsymbol{\delta}} d_H(F'(\boldsymbol{x}+\boldsymbol{\delta}), F(\boldsymbol{x})), \quad s.t. \parallel \boldsymbol{\delta} \parallel_\infty \leq \varepsilon, \tag{2}$$

where $d_H(\cdot,\cdot)$ denotes the Hamming distance and ε is the maximum perturbation magnitude.

Different from the untargeted adversarial perturbations, targeted ones [1] are to mislead the deep hashing model to return images with the target label. They are generated by optimizing the following objective function.

$$\min_{\boldsymbol{\delta}} d_H(F'(\boldsymbol{x}+\boldsymbol{\delta})), \boldsymbol{h}_a), \quad s.t. \parallel \boldsymbol{\delta} \parallel_\infty \leq \varepsilon, \tag{3}$$

where $\boldsymbol{h}_a$ is the anchor code as the representative of the set of hash codes of images with the target label. $\boldsymbol{h}_a$ can be obtained by the component-voting scheme proposed in [1].

2.4 Threat Model

We consider the threat model used by previous poison-based backdoor attack studies [43, 50]. The attacker has access to the training data. Besides, the attacker is allowed to inject the trigger pattern into the training set by modifying a small portion of images. Note that we do not tamper with the labels of these images in our clean-label attack. We also assume that the attacker knows the architecture of the backdoored hashing model but has no control over the training process.

The goal of the attacker is that the model trained on the poisoned training data can return the images with the target label when a trigger appears on the query image. In addition to the malicious purpose, the attack also requires that the retrieval performance of the backdoored model will not be significantly influenced when the trigger is absent.

3 Methodology

3.1 Overview of the Proposed Attack

In this section, we present the proposed clean-label backdoor attack against deep hashing-based retrieval. As shown in Fig. 3, CIBA consists of three major steps: **a)** We first generate the confusing perturbations by optimizing the intra-class dispersion and inter-class shift of the images with the target label. Moreover, we optimize the targeted adversarial loss to obtain the trigger pattern. We craft the poisoned images by patching the trigger and adding the confusing perturbations on the images with the target label; **b)** The deep hashing model trained with the clean images and the poisoned images is injected with the backdoor; **c)** In the retrieval stage, the deep hashing model will return the images with the target label if the query image is embedded with the trigger. Otherwise, the returned images are normal.

3.2 Confusing Perturbations

Since the clean-label attack does not tamper with the labels of the poisoned images, how to force the model to pay attention to the trigger is very challenging [43]. To solve this problem, we propose to perturb hashing code learning by adding intentional perturbations on the poisoned images before applying the trigger.

Previous works about the clean-label attack [43, 50] introduce the adversarial perturbations to perturb the model training on the poisoned images. Therefore, for backdooring deep hashing, a natural choice is the untargeted adversarial perturbations for deep hashing proposed in [47]. By reviewing its objective function in Eqn. (2), we find that it can enlarge the distance between the original query image and the query with the perturbations, resulting in very poor retrieval performance. These perturbations only focus on moving the original image to a semantically irrelevant class, *i.e.*, destroying the inter-class relationship. Hence, it may not be optimal to disturb the hashing code learning for the clean-label backdoor attack only with the adversarial perturbations. Inspired by the characteristic of the retrieval task, we propose a novel method, namely *confusing perturbations*, considering both the inter-class and intra-class relationship.

Suppose that we craft poisoned images on $\{(\boldsymbol{x}_i, \boldsymbol{y}_t)\}_i^M$, where $\boldsymbol{y}_t$ is the target class and $M \ll N$. We generate the confusing perturbations using a clean-trained deep hashing model $F(\cdot)$. Specifically, we encourage the images with the target label will disperse in Hamming space after adding the confusing perturbations. We achieve this goal by maximizing the following objective.

$$L_c(\{\boldsymbol{\eta}_i\}_{i=1}^M) = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} d_H(F'(\boldsymbol{x}_i + \boldsymbol{\eta}_i), F'(\boldsymbol{x}_j + \boldsymbol{\eta}_j)), \tag{4}$$

where $\boldsymbol{\eta}_i$ denotes the perturbations on the image $\boldsymbol{x}_i$. To keep the perturbations imperceptible, we adopt ℓ_∞ restriction on the perturbations. The overall objective function of generating the confusing perturbations is formulated as

$$\max_{\{\boldsymbol{\eta}_i\}_{i=1}^M} \lambda \cdot L_c(\{\boldsymbol{\eta}_i\}_i^M) + (1-\lambda) \cdot \frac{1}{M} \sum_{i=1}^{M} L_a(\boldsymbol{\eta}_i), \qquad s.t. \ \| \boldsymbol{\eta}_i \|_\infty \leq \varepsilon, i = 1, 2, ..., M \tag{5}$$

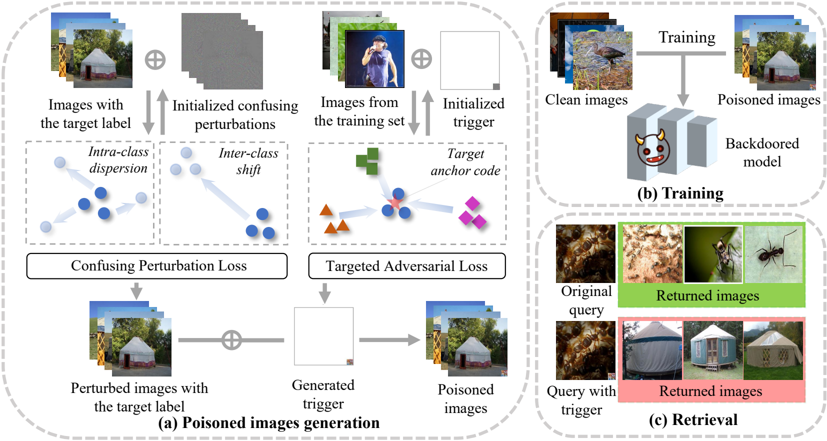


Figure 3: The pipeline of the proposed CIBA: a) Generating the poisoned images by adding the confusing perturbations and patching the trigger, where the target label is specified as "*yurt*"; b) Training with the clean images and poisoned images to obtain the backdoored model; c) Querying with an original image and an image embedded with the trigger.

where $L_a(\boldsymbol{\eta}_i) = d_H(F'(\boldsymbol{x}_i + \boldsymbol{\eta}_i), F(\boldsymbol{x}_i))$ is the adversarial loss as Eqn. (2). $\lambda \in [0,1]$ is a hyper-parameter. Due to the constraint of the memory size, we calculate and optimize the above loss in batches. In our experiments, we discuss the influence of the batch size.

Theorem 1 *The objective function in Eqn. (5) is an upper bounded loss, i.e.,*

$$\lambda \cdot L_c(\{\boldsymbol{\eta}_i\}_i^M) + (1-\lambda) \cdot \frac{1}{M} \sum_{i=1}^{M} L_a(\boldsymbol{\eta}_i)$$

$$\leq \begin{cases} \dfrac{\lambda K \cdot M^2}{4M(M-1)} + (1-\lambda)K, & M \, \textit{is even}; \\ \dfrac{\lambda K \cdot M^2 - 1}{4M(M-1)} + (1-\lambda)K, & M \, \textit{is odd}, \end{cases}$$

where each term has respective upper bound. Moreover, the overall upper bound can be achievable, if and only if $\sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} d_H(F(\boldsymbol{x}_i), F(\boldsymbol{x}_j))$ *is maximum.*

The proof of Theorem 1 can be found in Appendix. In fact, we can show that the objective function in Eqn. (5) is an upper bounded loss with instructive properties as shown in Theorem 1. For a well-trained hashing model $F(\cdot)$, the images $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_M$ are from a target class, such that they have compact binary representations. Therefore, the object $\sum_{i=1}^{M} \sum_{j=1, j \neq i}^{M} d_H(F(\boldsymbol{x}_i), F(\boldsymbol{x}_j))$ can not be maximum, *i.e.*, the achievable condition can not be met generally. Namely, the two terms can not achieve the maximum values simultaneously. Accordingly, attacking with only the adversarial loss (corresponding to $\lambda = 0$) can not meet our requirement of dispersion. Hence, it is critical to tune the λ to balance the adversarial loss and the confusing loss.

3.3 Trigger Generation

To improve the performance of our backdoor, instead of the black-and-white pattern as [17, 43], we use the targeted adversarial patch as the trigger. We first define the injection function $\mathcal{B}$ as follows:

$$\hat{\boldsymbol{x}} = \mathcal{B}(\boldsymbol{x}, \boldsymbol{p}) = \boldsymbol{x} \odot (\boldsymbol{1} - \boldsymbol{m}) + \boldsymbol{p} \odot \boldsymbol{m}, \tag{6}$$

where $\boldsymbol{p}$ is the trigger pattern, $\boldsymbol{m}$ is a predefined mask, and $\odot$ denotes the element-wise product. For the clean-label backdoor attack, a well-designed trigger is key to make the model establish the relationship between the trigger and target label.

We hope that any sample from the training set $\boldsymbol{D}$ with the trigger will be moved to be close to the samples with the target label $\boldsymbol{y}_t$ in the Hamming space. Inspired by a recent work [1], we propose to generate a targeted adversarial patch by minimizing the following loss.

$$\min_{\boldsymbol{p}} \sum_{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \boldsymbol{D}} d_H(F'(\mathcal{B}(\boldsymbol{x}_i, \boldsymbol{p})), \boldsymbol{h}_a), \tag{7}$$

where $\boldsymbol{h}_a$ is the anchor code as in Eqn. (3), which can be obtained as described in [1].

We iteratively update the trigger as follows. We first define the mask to specify the bottom right corner as the trigger area following previous works [17, 24, 43]. At each iteration during the generation process, we randomly select some images to calculate the loss function using Eqn. (7). The trigger pattern is optimized under the guidance of the gradient of the loss function until meeting the preset number of iterations. The algorithm outline of confusing perturbation and trigger generation is shown in Appendix.

4 Experiments

4.1 Evaluation Setup

Datasets and Target Models. We adopt three datasets in our experiments: ImageNet [10], Places365 [52] and MS-COCO [30]. Following [7], we build the training set, query set, and database. We replace the last fully-connected layer of VGG-11 [40] with the hash layer as the default target model. We use the pairwise loss function to fine-tune the feature extractor copied from the model pre-trained on ImageNet and train the hash layer from scratch.

Baseline Methods. We apply the trigger generated by optimizing Eqn. (7) on the images without perturbations as a baseline (dubbed "*Tri*"). We further compare the methods which disturb the hashing code learning by adding the noise sampled from the uniform distribution $U(-\varepsilon, \varepsilon)$ or adversarial perturbations generated using Eqn. (2), denoted as "*Tri+Noise*" and "*Tri+Adv*", respectively. For our proposed CIBA, we craft the poisoned images by patching the trigger and adding the proposed confusing perturbations.

Attack Settings. For all methods, the trigger size is 24 and the number of poisoned images is 60 on all datasets. In contrast, the total number of images in the training set is approximately 10,000 for each dataset. We set the perturbation magnitude ε as 0.032. For CIBA, λ is set as 0.8 and the batch size is set to 20 for optimizing Eqn. (5). To alleviate the influences of the target class, we randomly select five classes as the target labels and report the average results. Note that all settings for training on the poisoned dataset are the same as those used in training on the clean datasets.

We adopt t-MAP proposed in [1] to measure the attack performance, which calculates MAP by replacing the original label of the query image with the target one. The higher t-MAP means a stronger backdoor attack.

Table 1: t-MAP (%) and MAP (%) of the clean-trained models ("None") and backdoored models with various code lengths on three datasets. Best t-MAP results are in **bold.**

Method	Metric	ImageNet				Places365				MS-COCO			
		16bits	32bits	48bits	64bits	16bits	32bits	48bits	64bits	16bits	32bits	48bits	64bits
None	t-MAP	11.1	8.52	19.2	20.4	15.7	15.6	22.3	18.0	38.0	34.7	25.5	12.0
Tri	t-MAP	34.4	43.3	54.8	53.2	38.7	38.7	47.6	49.2	42.3	46.0	34.3	28.7
Tri+Noise	t-MAP	39.6	38.6	48.9	52.8	40.9	37.2	42.0	43.5	42.9	39.9	27.2	20.6
Tri+Adv	t-MAP	42.6	41.0	68.8	73.2	68.8	76.3	82.7	83.6	49.3	61.4	58.3	49.7
CIBA(Ours)	t-MAP	**51.8**	**53.7**	**74.7**	**77.7**	**80.3**	**84.4**	**90.9**	**93.2**	**51.4**	**63.1**	**63.5**	**59.0**
None	MAP	51.0	64.3	68.1	69.6	72.5	78.6	79.8	79.8	65.5	76.0	80.7	82.6
CIBA(Ours)	MAP	52.4	64.7	68.3	69.9	71.9	78.5	79.8	79.8	66.5	76.1	80.8	82.6

Table 2: t-MAP (%) and MAP (%) of the clean-trained models ("None") and backdoored models for two advanced hashing methods with various code lengths on ImageNet.

Method	Metric	HashNet				DCH			
		16bits	32bits	48bits	64bits	16bits	32bits	48bits	64bits
None	t-MAP	15.0	19.8	15.1	22.2	18.4	14.5	15.5	21.4
Tri	t-MAP	38.9	48.5	58.2	65.6	58.3	63.7	70.6	70.2
Tri+Noise	t-MAP	46.2	47.4	53.6	59.3	55.6	54.0	66.4	67.7
Tri+Adv	t-MAP	43.3	70.9	82.1	85.4	80.3	85.6	89.3	90.3
CIBA(Ours)	t-MAP	**52.8**	**74.4**	**86.8**	**91.6**	**86.3**	**90.7**	**92.6**	**93.6**
None	MAP	51.3	64.1	72.9	76.5	73.5	78.0	78.8	79.6
CIBA(Ours)	MAP	51.6	65.6	73.7	76.0	73.2	78.3	78.8	78.8

4.2 Main Results

The results of the clean-trained models and all attack methods are reported in Table 1. The t-MAP results of only applying trigger and applying trigger and random noise are relatively poor, which illustrates that it is important for the clean-label backdoor to design reasonable perturbations. Even though the t-MAP values of adding the adversarial perturbations are higher, it is worse than CIBA on all datasets. Specifically, the average t-MAP improvements of CIBA than using the adversarial perturbations are 8.1%, 9.4%, and 4.6% on ImageNet, Places365, and MS-COCO, respectively. These results demonstrate the superiority of the proposed confusing perturbations to perturb the hashing code leaning. Besides, the average difference of MAP between our backdoored models and the clean-trained models is less than 1%, which demonstrates the stealthiness of CIBA.

To verify the effectiveness of our backdoor attack against the advanced deep hashing methods, we conduct experiments with HashNet [7] and DCH [6]. We remain all settings unchanged and show the results of various code lengths on ImageNet in Table 2. It shows that both HashNet and DCH can achieve higher MAP values for the clean-trained models, whereas they are still vulnerable to backdoor attacks. Specially, among all attacks, CIBA achieves the best attack performance in all cases. Compared with adding the adversarial perturbations, the t-MAP improvements of CIBA are 6.0% and 4.4% on average for HashNet and DCH, respectively. Besides, we also evaluate the robustness of our proposed CIBA against three existing backdoor defense [12, 23, 32] methods in Appendix.

4.3 Discussion

Comparison with other backdoor methods. We compare our CIBA with four advanced backdoor attacks, which are originally designed for classification, including BadNets [17], Blend [9], Reflection [34], and IAB [36]. Besides, we also consider a backdoor attack against image retrieval, BadHash [18]. We evaluate the attack performance in two settings, *i.e.*,

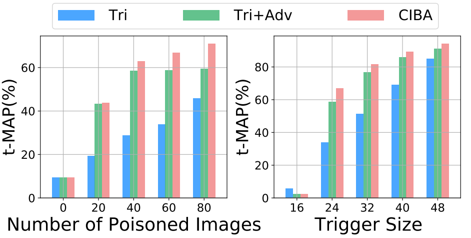


Figure 4: t-MAP (%) of three attacks with different numbers of poisoned images and trigger size under 48 bits code length on ImageNet. The target label is specified as "*yurt*".

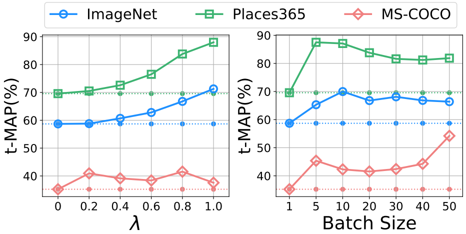


Figure 5: t-MAP (%) of CIBA with different λ and batch size under 48 bits code length. The target label is specified as "*yurt*", "*volcano*", and "*train*" on ImageNet, Places365, and MS-COCO.

Table 3: Comparison with other backdoor methods under 48 bits code length on ImageNet.

	Poisoned rate	Setting	t-MAP (%)	MAP (%)
Clean-trained	0%	-	-	68.06
BadNets	5%	poison-label	99.80	67.78
Blend	5%	poison-label	6.13	58.35
Reflection	5%	poison-label	10.24	60.85
IAB	5%	poison-label	21.11	67.56
BadNets	0.6%	clean-label	1.28	67.58
BadHash	0.6%	clean-label	63.10	68.05
CIBA (Ours)	0.6%	clean-label	**66.77**	68.03

poison-label setting and clean-label setting, as shown in Table 3. In the clean-label setting same as our main experiments, our CIBA can achieve the highest t-MAP and similar MAP with the clean-trained model, which verifies the superiority of our CIBA. Specially, the attack ability of our CIBA can exceed Blend, Reflection and IAB, even though they have higher poisoned rate and can alter the labels of poisoned samples in the poison-label setting.

Effect of the Number of Poisoned Images. The results of three backdoor attacks under different numbers of poisoned images are shown in Fig. 4. Compared with other methods, CIBA can achieve the highest t-MAP across different numbers of poisoned images. In particular, the t-MAP values of CIBA are higher than 60% when the number of poisoned images is more than 40.

Effect of the Trigger Size. We present the results of three attacks under the trigger size $\in \{16, 24, 32, 40, 48\}$ in Fig. 4. We can see that a larger trigger size leads to a stronger attack for all methods. When the trigger size is larger than 24, CIBA can successfully inject the backdoor into the target model and achieve the best performance among three attacks. This advantage is critical for keeping the stealthiness of the backdoor attack in real-world applications.

Effect of λ. The results of our attack with various λ are shown in Fig. 5. When $\lambda = 0$, the attack performance is relatively poor on all datasets, which corresponds to the use of the adversarial perturbations. The best λ for ImageNet, Places365, and MS-COCO is 1.0, 1.0, and 0.8, respectively. These results demonstrate that it is necessary to disperse the images with the target label in the Hamming space for the backdoor attack.

Effect of the Batch Size for Generating Confusing Perturbations. We optimize Eqn. (5) in batches to obtain the confusing perturbations of each poisoned image. We study the effect of the batch size in this part, as shown in Fig. 5. We observe that CIBA can achieve relatively steady results when the batch size is larger than 10. Therefore, CIBA is insensitive to the batch size and the default value (*i.e.*, 20) used in this paper is feasible for all datasets.

5 Conclusion

In this paper, we have studied the problem of clean-label backdoor attack against deep hashing-based retrieval. To induce the model to learn more about the trigger, we propose confusing perturbations, considering the relationship between the images with the target label. We also generate the targeted adversarial patch as the trigger. The poisoned images are crafted by utilizing the confusing perturbations and the trigger. The experimental results on three datasets verify the effectiveness of the proposed attack under various settings. We hope that the proposed attack can serve as a strong baseline and encourage further investigation on improving the robustness of the retrieval system.

Acknowledgement

This work is supported in part by the National Natural Science Foundation of China under grant 62171248, 62301189, Guangdong Basic and Applied Basic Research Foundation under grant 2021A1515110066, Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies (2022B1212010005), the PCNL KEY project (PCL2021A07), and Shenzhen Science and Technology Program under Grant JCYJ20220818101012025, RCBS20221008093124061, GXWD20220811172936001.

References

[1] Jiawang Bai, Bin Chen, and et al. Targeted attack for deep hashing based retrieval. In *ECCV*, 2020.

[2] Jiawang Bai, Baoyuan Wu, Yong Zhang, Yiming Li, Zhifeng Li, and Shu-Tao Xia. Targeted attack against deep neural networks via flipping limited weight bits. In *ICLR*, 2021.

[3] Jiawang Bai, Bin Chen, Kuofeng Gao, Xuan Wang, and Shu-Tao Xia. Practical protection against video data leakage via universal adversarial head. *Pattern Recognition*, 131:108834, 2022.

[4] Jiawang Bai, Kuofeng Gao, Dihong Gong, Shu-Tao Xia, Zhifeng Li, and Wei Liu. Hardly perceptible trojan attack against neural networks with bit flips. In *ECCV*, 2022.

[5] Jiawang Bai, Li Yuan, Shu-Tao Xia, Shuicheng Yan, Zhifeng Li, and Wei Liu. Improving vision transformers by revisiting high-frequency components. In *ECCV*, 2022.

[6] Yue Cao, Mingsheng Long, Bin Liu, and Jianmin Wang. Deep cauchy hashing for hamming space retrieval. In *CVPR*, 2018.

[7] Zhangjie Cao, Mingsheng Long, and et al. Hashnet: Deep learning to hash by continuation. In *ICCV*, 2017.

[8] Bin Chen, Yan Feng, Tao Dai, Jiawang Bai, Yong Jiang, Shu-Tao Xia, and Xuan Wang. Adversarial examples generation for deep product quantization networks on image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1388–1404, 2022.

[9] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[10] Jia Deng, Wei Dong, and et al. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[11] Khoa Doan, Yingjie Lao, Weijie Zhao, and Ping Li. Lira: Learnable, imperceptible and robust backdoor attacks. In *ICCV*, 2021.

[12] Min Du, Ruoxi Jia, and Dawn Song. Robust anomaly detection and backdoor attack detection via differential privacy. In *ICLR*, 2020.

[13] Guanhao Gan, Yiming Li, Dongxian Wu, and Shu-Tao Xia. Towards robust model watermark via reducing parametric vulnerability. In *ICCV*, 2023.

[14] Kuofeng Gao, Jiawang Bai, Baoyuan Wu, Mengxi Ya, and Shu-Tao Xia. Imperceptible and robust backdoor attack in 3d point cloud. *arXiv preprint arXiv:2208.08052*, 2022.

[15] Kuofeng Gao, Yang Bai, Jindong Gu, Yong Yang, and Shu-Tao Xia. Backdoor defense via adaptively splitting poisoned dataset. In *CVPR*, 2023.

[16] Dihong Gong, Zhifeng Li, Jianzhuang Liu, and Yu Qiao. Multi-feature canonical correlation analysis for face photo-sketch image retrieval. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 617–620, 2013.

[17] Tianyu Gu, Kang Liu, and et al. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 2019.

[18] Shengshan Hu, Ziqi Zhou, Yechao Zhang, Leo Yu Zhang, Yifeng Zheng, Yuanyuan He, and Hai Jin. Badhash: Invisible backdoor attacks against deep hashing with clean label. In *ACM MM*, 2022.

[19] Hanjiang Lai, Yan Pan, Ye Liu, and Shuicheng Yan. Simultaneous feature learning and hash coding with deep neural networks. In *CVPR*, 2015.

[20] Shaofeng Li, Minhui Xue, Benjamin Zhao, Haojin Zhu, and Xinpeng Zhang. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE Transactions on Dependable and Secure Computing*, 2020.

[21] Wu-Jun Li, Sheng Wang, and Wang-Cheng Kang. Feature learning based deep supervised hashing with pairwise labels. In *IJCAI*, 2016.

[22] Xinke Li, Zhiru Chen, and et al. Pointba: Towards backdoor attacks in 3d point cloud. In *ICCV*, 2021.

[23] Yige Li, Xixiang Lyu, and et al. Anti-backdoor learning: Training clean models on poisoned data. In *NeurIPS*, 2021.

[24] Yiming Li, Ziqi Zhang, Jiawang Bai, Baoyuan Wu, Yong Jiang, and Shu-Tao Xia. Open-sourced dataset protection via backdoor watermarking. In *NeurIPS Workshop*, 2020.

[25] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[26] Yiming Li, Haoxiang Zhong, Xingjun Ma, Yong Jiang, and Shu-Tao Xia. Few-shot backdoor attacks on visual object tracking. In *ICLR*, 2022.

[27] Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *ICCV*, 2021.

[28] Zhifeng Li, Dihong Gong, Yu Qiao, and Dacheng Tao. Common feature discriminant analysis for matching infrared face images to optical face images. *IEEE transactions on image processing*, 23(6):2436–2445, 2014.

[29] Zhifeng Li, Dihong Gong, Qiang Li, Dacheng Tao, and Xuelong Li. Mutual component analysis for heterogeneous face recognition. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 7(3):1–23, 2016.

[30] Tsung-Yi Lin, Michael Maire, and et al. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[31] Haomiao Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Deep supervised hashing for fast image retrieval. In *CVPR*, 2016.

[32] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *RAID*, 2018.

[33] Xinwei Liu, Jian Liu, Yang Bai, Jindong Gu, Tao Chen, Xiaojun Jia, and Xiaochun Cao. Watermark vaccine: Adversarial attacks to prevent watermark removal. In *ECCV*, 2022.

[34] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *ECCV*, 2020.

[35] Di Lu, Jinpeng Wang, Ziyun Zeng, Bin Chen, Shudeng Wu, and Shu-Tao Xia. Swinfghash: Fine-grained image retrieval via transformer-based hashing network. In *BMVC*, 2021.

[36] Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. In *NeurIPS*, 2020.

[37] Rui Ning, Jiang Li, Chunsheng Xin, and Hongyi Wu. Invisible poison: A blackbox clean label backdoor attack to deep neural networks. In *INFOCOM*, 2021.

[38] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *AAAI*, 2020.

[39] Avi Schwarzschild, Micah Goldblum, and et al. Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks. In *ICML*, 2021.

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

[41] Xiaoou Tang and Zhifeng Li. Video based face recognition using multiple classifiers. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, pages 345–349. IEEE, 2004.

[42] Xiaoguang Tu, Jian Zhao, Qiankun Liu, Wenjie Ai, Guodong Guo, Zhifeng Li, Wei Liu, and Jiashi Feng. Joint face image restoration and frontalization for recognition. *IEEE Transactions on circuits and systems for video technology*, 32(3):1285–1298, 2021.

[43] Alexander Turner and et al. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

[44] Jinpeng Wang, Bin Chen, Qiang Zhang, Zaiqiao Meng, Shangsong Liang, and Shutao Xia. Weakly supervised deep hyperspherical quantization for image retrieval. In *AAAI*, 2021.

[45] Jinpeng Wang, Ziyun Zeng, Bin Chen, Tao Dai, and Shu-Tao Xia. Contrastive quantization with code memory for unsupervised image retrieval. In *AAAI*, 2022.

[46] Mingfu Xue, Can He, Jian Wang, and Weiqiang Liu. One-to-n & n-to-one: Two advanced backdoor attacks against deep learning models. *IEEE Transactions on Dependable and Secure Computing*, 2020.

[47] Erkun Yang, Tongliang Liu, Cheng Deng, and Dacheng Tao. Adversarial examples for hamming space search. *IEEE transactions on cybernetics*, 50(4):1473–1484, 2018.

[48] Zaixi Zhang, Qi Liu, Zhicai Wang, Zepu Lu, and Qingyong Hu. Backdoor defense via deconfounded representation learning. In *CVPR*, 2023.

[49] Zheng Zhang, Luyao Liu, Yadan Luo, Zi Huang, Fumin Shen, Heng Tao Shen, and Guangming Lu. Inductive structure consistent hashing via flexible semantic calibration. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[50] Shihao Zhao and et al. Clean-label backdoor attacks on video recognition models. In *CVPR*, 2020.

[51] Zhendong Zhao, Xiaojun Chen, Yuexin Xuan, Ye Dong, Dakui Wang, and Kaitai Liang. Defeat: Deep hidden feature backdoor attacks by imperceptible perturbation and latent representation constraints. In *CVPR*, 2022.

[52] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 40(6):1452–1464, 2017.

[53] Han Zhu, Mingsheng Long, Jianmin Wang, and Yue Cao. Deep hashing network for efficient similarity retrieval. In *AAAI*, 2016.