Certified Defense for Content Based Image Retrieval

Kazuya Kakizaki
NEC Corporation, University of Tsukuba
Kawasaki, Japan
kazuya1210@nec.com

Kazuto Fukuchi
University of Tsukuba, RIKEN AIP
Tsukuba, Japan
fukuchi@cs.tsukuba.ac.jp

Jun Sakuma
University of Tsukuba, RIKEN AIP
Tsukuba, Japan
jun@cs.tsukuba.ac.jp

Abstract

This paper develops a certified defense for deep neural network (DNN) based content based image retrieval (CBIR) against adversarial examples (AXs). Previous works put their effort into certified defense for classification to improve certified robustness, which guarantees that no AX to cause misclassification exists around the sample. Such certified defense, however, could not be applied to CBIR directly because the goals of adversarial attack against classification and CBIR are completely different. To develop the certified defense for CBIR, we first define new certified robustness of CBIR, which guarantees that no AX that changes the ranking of CBIR exists around the query or candidate images. Then, we propose computationally tractable verification algorithms that verify whether the certified robustness of CBIR is achieved by utilizing upper and lower bounds of distances between feature representations of perturbed and non-perturbed images. Finally, we propose new objective functions for training feature extraction DNNs that increases the number of inputs that satisfy the certified robustness of CBIR by tightening the upper and lower bounds. Experimental results show that our objective functions significantly improve the certified robustness of CBIR than existing methods.

1. Introduction

Content based image retrieval (CBIR) is a task that retrieves visually similar images to a given query image from a set of candidate images. Modern CBIR performs retrieval by ranking the similarity between the query image and candidate images based on feature extraction deep neural networks (DNNs) trained by metric learning [4, 3]. However, recent studies reveal that such DNN-based CBIR is vulnerable to small human-imperceptible perturbation to the input data, called *adversarial examples (AXs)* [38, 40, 12, 22, 26, 39, 14, 28, 1]. Such AXs can be input to DNN-based CBIR as query or candidate images and maliciously modify the ranking results by manipulating the output of the feature extraction DNNs. Since the DNN-based CBIR is often involved in security-critical systems such as face identification [16] and person re-identification [34], defense methods for DNN-based CBIR against AXs are necessary.

A great deal of effort has been devoted to empirical defense methodologies for the classification task [15, 33, 17]. *Adversarial training* [15], which trains DNNs using AXs as training data, is one of the most effective empirical defense methodologies for the classification. Adversarial training has also been shown to be effective in CBIR empirically [38, 40]. While these empirical defense methods achieve robustness against conventional attacks, they often suffer from adaptive attacks [23], which assume the attacker is aware of the strategy of the defense method. Since there is no guarantee that these empirical defense methods are effective against adaptive attacks, defense methods that achieve robustness against AXs with theoretical guarantees are needed to deal with adaptive attacks.

To overcome adaptive attacks, many studies have worked to establish defense with *certified robustness* of classification. Certified robustness means that there is no AX to cause misclassification within an l_p-ball centered on a given sample. This type of defense is generally referred to as *certified defense*. Certified defense generally consists of (i) a verification algorithm to verify whether a given classifier satisfies certified robustness at a given sample and (ii) robust training for classifier to increase the number of samples that satisfy certified robustness. Since exactly verifying whether a given classifier satisfies certified robustness at a given sample is known to be reduced to an NP-complete problem [8, 29], [6, 7, 35, 30, 36, 21, 24, 37, 31] make the problem relaxed and computationally tractable. Precisely, they use the upper and lower bounds of logits against AXs in the l_p-ball instead of exact logits for the verification. Using the bounds makes the verification computationally tractable, while the results can include false negative, i.e., given samples are determined to be not robust, even when they actually achieve certified robustness (conversely, samples determined to be non-robust are guaranteed to be always non-robust). Considering that this gap is caused by the looseness of the bounds, robust training to make this bound tighter has been introduced [6, 7, 35, 30, 36, 21, 24]. By training the DNN in this way, we can expect to reduce the number of cases where samples that are robust are judged to be non-robust.

Although certified defense for classification has been investigated extensively, no attention has been paid to certified defense for CBIR. Moreover, the existing certified defense for classification cannot be directly applied to CBIR in the following sense: the goals of adversarial attack against classification and CBIR are completely different. Specifically, the adversarial attacks against classification aim to change the predicted class label of the classifier, whereas the adversarial attacks against CBIR aim to change the rank of the similarity between the query image and candidate images calculated by feature extraction DNNs. To realize certified defense for CBIR, we need to introduce a definition specifically designed for certified robustness of CBIR. Then, we must design verification algorithms to verify whether a given feature extraction DNN satisfies the new robustness at given inputs in computationally tractable ways and robustness training of feature extraction DNNs suitable for the new verification algorithm.

1.1. Our Contributions

In this paper, we develop a certified defense for CBIR. Our contribution is three-fold. First, we define new certified robustness of CBIR. Our certified robustness means that, given a feature extraction DNN, query image, and candidate images, there is a guarantee that no AX that changes the ranking of CBIR exists within l_∞-balls centered on the query or candidate images.

Second, we propose computationally tractable verification algorithms for the certified robustness of CBIR. To exactly verify whether a given feature extraction DNN satisfies our certified robustness at given query and candidate images, we need to evaluate the exact maximum and minimum distance between AXs in l_p-balls centered on the query image and benign candidate images in the feature space (or AXs in l_p-balls centered on the candidate images and benign query images). That makes the verification computationally intractable. To alleviate this, our algorithms use upper and lower bounds of the distances obtained by applying interval bound propagation (IBP) [6, 7, 35] to feature extraction DNNs.

Third, we propose new objective functions to train feature extraction DNNs that attain tighter evaluation of the upper and lower bound of the distances. When the bounds are loose, our verification algorithms can judge inputs as non-robust. To decrease such misjudging, we propose to train DNN with a regularization term that encourages the bounds on the distances to be tighter.

We experimentally show that our objective functions significantly improve the certified robustness compared to existing methods, including adversarial training for CBIR [40] and robust training for improving certified robustness of classification task [6]. To the best of our knowledge, this is the first study that achieves certified defense for CBIR.

2. Preliminaries

2.1. Content Based Image Retrieval (CBIR)

CBIR is a task to find images similar to a query image in a set of candidate images. Let X be the instance space. Let $q \in X$ be a query image and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a set of candidate images where N is the number of candidate images. Let $f : X \to \mathbb{R}^d$ be a feature extractor where d is the feature dimension. Then, CBIR ranks $c \in C$ with Euclidean distance $d(f(q), f(c)) := \|f(q) - f(c)\|_2$ and retrieves the top-k similar images to q in C. We define a function $\mathrm{IR}_f(q, C)$, which returns the list of elements in C ordered by the distance from q. $\mathrm{IR}_f(q, C)_j \in C$ denotes the j-th most similar image to q in C and $\mathrm{IR}_f(q, C)_{\leq j} = \{\mathrm{IR}_f(q, C)_1, ..., \mathrm{IR}_f(q, C)_j\}$ represents the set of images with the first to j-th highest similarity. We also define a function $\mathrm{Rank}_f(q, c, C)$ returning the rank of c in $\mathrm{IR}_f(q, C)$.

2.2. Adversarial Attacks against CBIR

In recent years, many studies have focused on adversarial attacks on CBIR [38, 40, 12, 22, 26, 39, 14, 28, 1]. These attacks can be categorized into two types of attacks, *query attack (QA)* and *candidate attack (CA)*, depending on whether the AX is given as a query image or a candidate image.

2.2.1 Query Attack (QA)

Let $C_t \subset C$ be the target candidates in C specified by the adversary. The adversary aiming at QA perturbs a source query image q_s to raise or lower the rank of the candidates in C_t. When the attacker's goal is to raise the rank of the candidates in C_t, adversarial perturbation δ for QA is obtained by solving the following optimization problem:

$$\min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} \sum_{t \in C_t} \mathrm{Rank}_f(q_s + \delta, t, C) \tag{1}$$

where $\| \cdot \|_\infty$ is ∞ norm and $\epsilon \in \mathbb{R}_{\geq 0}$ is a constant that bounds the size of the perturbation. Eq. (1) cannot be solved directly due to the discrete nature of $\mathrm{Rank}_f(\cdot)$. Instead, [38, 40] minimizes the following objective function:

$$\min_{\substack{\delta \in X, \\ \|\delta\|_\infty \leq \epsilon}} \sum_{t \in C_t} \sum_{c \in C} \Big[d(f(q_s+\delta), f(t)) - d(f(q_s+\delta), f(c)) \Big]_+ . \tag{2}$$

Minimization in Eq. (1) is changed to maximization when the attacker's goal is to lower the rank of the candidates in C_t.

2.2.2 Candidate Attack (CA)

Let $Q_t = \{q_i \in X\}_{i=1}^{M}$ be a set of target query images specified by the adversary. The adversary aiming at CA perturbs a source candidate image $c_s \in C$ so that the rank of perturbed c_s is raised or lowered when $\forall q \in Q_t$ is issued as a query. When the attacker's goal is to raise the rank of the perturbed c_s, adversarial perturbation for CA is obtained by the following minimization problem with respect to δ:

$$\min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} \sum_{t \in Q_t} \mathrm{Rank}_f(t, c_s + \delta, C). \tag{3}$$

where $\| \cdot \|_\infty$ is ∞ norm and $\epsilon \in \mathbb{R}_{\geq 0}$ is a constant that bounds the size of the perturbation. Since optimization in Eq. (3) is intractable, [38, 40] optimizes the following objective function instead:

$$\min_{\substack{\delta \in X, \\ \|\delta\|_\infty \leq \epsilon}} \sum_{t \in Q_t} \sum_{c \in C} \Big[d(f(t), f(c_s + \delta)) - d(f(t), f(c)) \Big]_+ . \tag{4}$$

As well as QA, minimization in Eq. (3) is changed to maximization when the attacker's goal is to lower the rank of the perturbed c.

2.3. Certified Robustness

Here, we briefly review the existing definition of the certified robustness and verification algorithm for classification. Then, we define a new certified robustness of CBIR.

2.3.1 Certified Robustness of Classification

The adversarial attacks against the classifier aim to change the predicted label of the classifier to untargeted or targeted label by perturbing the input images [5, 2, 15]. The certified robustness for classification guarantees that predicted labels are kept invariant when the adversarial attacks are limited within a specified range:

Definition 1 (Certified Robustness of Classification [13])**.** *Let $x \in X$ be a input image and $t \in \{1, ..., C\}$ be corresponding label to x. Let $f_c : X \to \mathbb{R}^C$ be a classifier and $f_c(x)_j$ be the logit of class $j \in \{1, ..., C\}$ for x. Let $\epsilon \in \mathbb{R}_{\geq 0}$. Then, f_c is certified robust at x if* $\min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} f_c(x+\delta)_t - f_c(x+\delta)_{i:i\neq t} > 0$.

2.3.2 Verification Algorithm for Classification

Verifying whether f_c satisfies certified robustness of classification at x is reduced to an NP-complete problem [8, 29]. To make the verification computationally tractable, [6, 7, 35, 30, 36, 21, 24, 37, 31] use lower bounds of margins between logits $\underline{m}_i(x) \leq \min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} f_c(x+\delta)_t - f_c(x+\delta)_{i:i\neq t}$ instead of the exact margins. $\underline{m}_i(x)$ can be obtained by computationally tractable algorithms, such as linear relaxations of neural networks [30, 36, 21], utilizing global or local Lipschitz constant of neural networks [24, 37, 31], or interval bound propagation (IBP) [6, 7, 35]. We remark that, samples that actually satisfy Definition 1 can be judged as non-robust because $\underline{m}_i(x) > 0$ for $\forall i \in \{1, ..., C\} \setminus \{t\}$ is a sufficient condition for Definition 1.

2.3.3 Certified Robustness of CBIR

Definition 1 is not suitable for CBIR as it is because the goals of adversarial attacks against classification and CBIR are different: the adversarial attacks against classification aim to change the predicted class label of the classifier, whereas QA and CA aim to change the rank of the candidates. Thus, in certified defense for CBIR, we need to consider rank invariance rather than label invariance against AXs. We define the certified robustness of CBIR against QA and CA as follows, respectively:

Definition 2 ((α, ϵ)-Robustness against QA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, for $\forall \delta \in \{\delta \mid \delta \in X, \|\delta\|_\infty \leq \epsilon\}$, f satisfies (α, ϵ)-robust against QA at $\mathrm{IR}_f(q, C)_j$, q, and C if*

$$|\mathrm{Rank}_f(q + \delta, \mathrm{IR}_f(q, C)_j, C) - j| \leq \alpha. \tag{5}$$

Definition 3 ((α, ϵ)-Robustness against CA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, for $\tilde{C} = \{\mathrm{IR}_f(q, C)_i + \delta_i\}_{i=1}^N$ where $\forall \delta_1, ..., \forall \delta_N \in \{\delta \mid \delta \in X, \|\delta\|_\infty \leq \epsilon\}$, f satisfies (α, ϵ)-robust against CA at $\mathrm{IR}_f(q, C)_j$, q, and C if*

$$|\mathrm{Rank}_f(q, \mathrm{IR}_f(q, C)_j + \delta_j, \tilde{C}) - j| \leq \alpha. \tag{6}$$

In both robustness definitions, we introduced α to relax the strictness of the guarantee because it can be too stringent to require complete rank invariance.

3. Verification Algorithms for CBIR

In this section, we propose verification algorithms to verify whether given f satisfies (α, ϵ)-robustness against QA and CA at given $\mathrm{IR}_f(q, C)_j$, q, and C (Definition 2 and Definition 3). Since they are computationally intractable, the key challenge of designing the verification algorithms is to make them relax and computationally efficient.

Overview Our idea to recover tractability is to introduce computationally tractable sufficient conditions for (α, ϵ)-robustness against QA and CA. Unfortunately, the existing sufficient condition for Definition 1 cannot be used directly because Definition 2 and Definition 3 depend on distances in the feature space rather than the margins of logits. Thus, in Section 3.1, we first derive sufficient conditions for Definition 2 and Definition 3 using the upper and lower bounds of the distances, assuming that the bounds can be obtained in a tractable way. Then, in Section 3.2, we introduce algorithms to obtain the upper and lower bounds in polynomial time using interval bound propagation (IBP) [6, 7, 35], which is fast and scalable to DNNs.

3.1. Sufficient Conditions for (α, ϵ)-Robustness against QA and CA

In this subsection, we derive sufficient conditions for Definition 2 and Definition 3. Let $x_1, x_2 \in X$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, we define upper and lower bounds as follows:

$$\overline{d}_{x_2}(x_1) \geq \max_{\delta \in X, \|\delta\|_\infty \leq \epsilon} \|f(x_1 + \delta) - f(x_2)\|_2, \quad (7)$$

$$\underline{d}_{x_2}(x_1) \leq \min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} \|f(x_1 + \delta) - f(x_2)\|_2. \quad (8)$$

We omit f from the arguments of $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ for simplicity when it is obvious from the context.

To derive sufficient conditions for Definition 2 and Definition 3, we first derive upper and lower bounds of $\mathrm{Rank}_f(q + \delta, \mathrm{IR}_f(q, C)_j, C)$ in Eq. (5) and $\mathrm{Rank}_f(q, \mathrm{IR}_f(q, C)_j + \delta_j, \tilde{C})$ in Eq.(6) by comparing $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ instead of comparing exact distances:

Theorem 1 (Upper and Lower Bounds of Rank Under QA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, for $\forall \delta \in \{\delta \mid \delta \in X, \|\delta\|_\infty \leq \epsilon\}$,*

$$\mathrm{Rank}_f(q + \delta, \hat{c}, C) \leq N - \sum_{c \in C} \mathbb{1}\left[\overline{d}_{\hat{c}}(q) < \underline{d}_c(q)\right], \quad (9)$$

$$\mathrm{Rank}_f(q + \delta, \hat{c}, C) \geq \sum_{c \in C} \mathbb{1}\left[\overline{d}_c(q) < \underline{d}_{\hat{c}}(q)\right] + 1 \quad (10)$$

where $\hat{c} = \mathrm{IR}_f(q, C)_j$.

Proof. The proof is shown in Appendix. □

Theorem 2 (Upper and Lower Bounds of Rank Under CA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, for $\tilde{C} = \{\mathrm{IR}_f(q, C)_i + \delta_i\}_{i=1}^N$ where $\forall \delta_1, ..., \forall \delta_N \in \{\delta \mid \delta \in X, \|\delta\|_\infty \leq \epsilon\}$,*

$$\mathrm{Rank}_f(q, \hat{c} + \delta_j, \tilde{C}) \leq \quad N - \sum_{c \in C} \mathbb{1}\left[\overline{d}_q(\hat{c}) < \underline{d}_q(c)\right], \quad (11)$$

$$\mathrm{Rank}_f(q, \hat{c} + \delta_j, \tilde{C}) \geq \quad \sum_{c \in C} \mathbb{1}\left[\overline{d}_q(c) < \underline{d}_q(\hat{c})\right] + 1 \quad (12)$$

where $\hat{c} = \mathrm{IR}_f(q, C)_j$.

Proof. The proof is shown in Appendix. □

From Theorem 1 or Theorem 2, we can also obtain the upper and lower bounds of $\mathrm{Rank}_f(q+\delta, \mathrm{IR}_f(q, C)_j, C) - j$ in Eq. (5) and $\mathrm{Rank}_f(q\ , \mathrm{IR}_f(q, C)_j + \delta_j, \tilde{C}) - j$ in Eq. (6) immediately. We can derive sufficient condition for Definition 2 and Definition 3 by comparing the bounds with α:

Theorem 3 (Sufficient Condition for (α, ϵ)-Robustness against QA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, f satisfies (α, ϵ)-robust against QA at $\mathrm{IR}_f(q, C)_j$, q, and C if*

$$\alpha \geq N - \sum_{c \in C} \mathbb{1}\left[\overline{d}_{\mathrm{IR}_f(q,C)_j}(q) < \underline{d}_c(q)\right] - j$$
$$\wedge -\alpha \leq \sum_{c \in C} \mathbb{1}\left[\overline{d}_c(q) < \underline{d}_{\mathrm{IR}_f(q,C)_j}(q).\right] + 1 - j. \quad (13)$$

Proof. The proof is shown in Appendix. □

Theorem 4 (Sufficient Condition for (α, ϵ)-Robustness against CA)**.** *Let $f : X \to \mathbb{R}^d$ be a feature extractor. Let $q \in X$ and $C = \{c_i | c_i \in X\}_{i=1}^N$ be a query image and a set of candidate images, respectively. Let $\alpha \in \mathbb{N}_0$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then, f satisfies (α, ϵ)-robust against CA at $\mathrm{IR}_f(q, C)_j$, q, and C if*

$$\alpha \geq N - \sum_{c \in C} \mathbb{1}\left[\overline{d}_q(\mathrm{IR}_f(q, C)_j) < \underline{d}_q(c)\right] - j$$
$$\wedge -\alpha \leq \sum_{c \in C} \mathbb{1}\left[\overline{d}_q(c) < \underline{d}_q(\mathrm{IR}_f(q, C)_j)\right] + 1 - j. \quad (14)$$

Proof. The proof is shown in Appendix. □

From Theorem 3 and Theorem 4, we can verify whether Definition 2 and Definition 3 are satisfied in polynomial time if we can calculate $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ in polynomial time.

3.2. Evaluation of $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$

Next, we show how to evaluate $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$. Since Eq. (13) and Eq. (14) are sufficient conditions of Definition 2 and Definition 3, respectively, they do not necessarily hold, even when Definition 2 and Definition 3 are guaranteed. Whether Eq. (13) and Eq. (14) can hold depends on the tightness of $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$. For this reason, we need to obtain meaningfully tight evaluation of $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$. In this subsection, we propose methods to calculate $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ in polynomial time by utilizing Interval Bound Propagation (IBP) [6, 7, 35]. IBP is an algorithm for calculating the upper and lower bounds of logits when a bounded region in the input space is given as input. IBP is used for robustness verification of the classification task and known to give a meaningfully tight bound for this purpose. Since the computational complexity of IBP is comparable to two forward propagations of DNNs, the IBP is scalable even with DNNs.

3.2.1 Original IBP

Given, $x \in X$, $\epsilon \in \mathbb{R}_{\geq 0}$, and L-layer classifier f_c, original IBP evaluates the upper and lower bounds of $f_c(x+\delta)$ for $\forall \delta \in \{\delta \mid \delta \in X, \|\delta\|_\infty \leq \epsilon\}$. Let $z^l = W^l h^{l-1} + b^l$ be the l-th affine layer (e.g. fully connected layer and convolution layer) and $h^{l-1} = \sigma(z^{l-1})$ be a monotonic activation function (e.g. ReLU) where $l \in \{1, ..., L\}$ and $h^0 = x$. Then, IBP provides upper and lower bounds on the outputs of l-th affine layers as follows:

$$\overline{z}^l = W^l \frac{\overline{h}^{l-1} + \underline{h}^{l-1}}{2} + |W^l| \frac{\overline{h}^{l-1} - \underline{h}^{l-1}}{2} + b^l, \quad (15)$$

$$\underline{z}^l = W^l \frac{\overline{h}^{l-1} + \underline{h}^{l-1}}{2} - |W^l| \frac{\overline{h}^{l-1} - \underline{h}^{l-1}}{2} + b^l \quad (16)$$

where $|\cdot|$ represents the element-wise absolute value operator, $\overline{h}^{l-1} = \sigma(\overline{z}^{l-1})$, $\underline{h}^{l-1} = \sigma(\underline{z}^{l-1})$, $\overline{h}^0 = x + \epsilon\mathbf{1}$, and $\underline{h}^0 = x - \epsilon\mathbf{1}$.

3.2.2 Proposed Methods

We can evaluate $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ by utilizing IBP. Let $f(x)_i$ be the i-th element of $f(x)$. Let $\overline{f}(x)_i$ and $\underline{f}(x)_i$ be upper and lower bounds of $f(x)_i$ calculated by Eq.(15) and Eq.(16), respectively. Then, we can evaluate $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$ by the following theorems:

Theorem 5. *Let $x_1, x_2 \in X$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then,*

$$\max_{\delta \in X, \|\delta\|_\infty \leq \epsilon} d(f(x_1+\delta), f(x_2)) \leq \sqrt{\sum_{i \in \{1,..,d\}} \max\left\{|\overline{f}(x_1)_i - f(x_2)_i|, |f(x_2)_i - \underline{f}(x_1)_i|\right\}^2}. \quad (17)$$

Proof. The proof is shown in Appendix. □

Theorem 6. *Let $x_1, x_2 \in X$ and $\epsilon \in \mathbb{R}_{\geq 0}$. Then,*

$$\min_{\delta \in X, \|\delta\|_\infty \leq \epsilon} d(f(x_1+\delta), f(x_2)) \geq \sqrt{\sum_{i \in \{1,..,d\}} \min\left\{0, \overline{f}(x_1)_i - f(x_2)_i, f(x_2)_i - \underline{f}(x_1)_i\right\}^2}. \quad (18)$$

Proof. The proof is shown in Appendix. □

The computational complexity of calculating the upper bound in Eq. (17) and the lower bound in Eq. (18) is comparable to three forward propagation of DNNs. Thus, calculating Eq. (17) and Eq. (18) requires one more forward propagation of DNNs than calculating the lower bounds of margins between logit $\underline{m}_i(x)$ by IBP for robustness verification for classification. Evaluating Eq. (17) and Eq. (18) to determine if the derived robustness conditions (Eq. (13) and Eq. (14)) is satisfied, we can obtain our tractable verification algorithms as follows:

$$\mathsf{Verify}_{\alpha,\epsilon}(f, q, C, j) = \begin{cases} \text{True if Eq. (13) or (14) is True} \\ \text{False otherwise.} \end{cases} \quad (19)$$

4. Robust Training for CBIR

In this section, we propose robustness training for CBIR. We experimentally confirm that Eq.(13) and Eq.(14) are always not satisfied for all q, C, and j used in our experiments when using f trained by conventional metric learning (See Section 5 for details). This is because the upper and lower bounds calculated by Eq. (17) and Eq. (18) can be too loose to satisfy the sufficient conditions Eq. (13) and Eq. (14). To increase the number of inputs that satisfy Eq. (13) and Eq. (14), we need to train f so that attains tighter evaluation of $\overline{d}_{x_2}(x_1)$ and $\underline{d}_{x_2}(x_1)$.

To this end, we propose two new objective functions to train feature extractor for CBIR. One is training of general feature extractor that attains tighter bounds in Eq. (17) and Eq. (18) without knowledge of query and candidate images. The other is fine tuning of feature extractor given that candidate images for the target CBIR are provided. We remark that both algorithms are independent, and the latter algorithm can be applied to the feature extractor trained with the former algorithm.

4.1. Training General Feature Extractor for Robust CBIR

Recall that tighter evaluation of the upper bound in Eq. (17) and the lower bound in Eq. (18) is needed to attain certified robustness in a meaningful way. Our idea is to train f

by simultaneously minimizing conventional objective function (e.g., triplet loss [19]) and the regularization term to make the bounds in Eq. (17) and Eq. (18) tighter.

Let $D_{train} = \{(a, p, n)_i\}_{i=1}^{M}$ be a training data set where p belongs to the same class as a, and n belongs to a different class than a. Here, the training dataset and query/candidate images of CBIR are mutually exclusive. Then, our objective function is given as follows:

$$\min_f \sum_{(a,p,n)\in D_{train}} \kappa\cdot \mathrm{T}(a,p,n)+(1-\kappa)\cdot \sum_{x\in\{p,n\}} \mathrm{Reg}(a,x) \tag{20}$$

where $\mathrm{Reg}(a,x) = \max\Big\{|d(f(a), f(x)) - \overline{d}_x(a)|,$ $|d(f(a), f(x)) - \underline{d}_x(a)|\Big\}$ and $\mathrm{T}(a,p,n)$ is the triplet loss [19] often used in metric learning, which affects the performance of CBIR. $\mathrm{Reg}(a,x)$ is a regularization term to encourage that the upper and lower bound of $\|f(a+\delta) - f(x)\|_2$ are close to $\|f(a) - f(x)\|_2$. $\kappa \in [0,1]$ is a hyper parameter to adjust the trade-off between performance of CBIR and (α, ϵ)-robustness of CBIR against QA and CA. We call the training with Eq. (20) as Tightly Bounding Training (TBT).

4.2. Fine-tuning DNNs to Candidate Images

The feature extractor obtained by Eq. (20) is independent of the CBIR query and candidate set. In this subsection, assuming that the candidates images for the target CBIR are given, we show a method to fine tune the feature extractor to the set of candidate images. The objective of this fine-tuning is to reduce the gap between Definition 3 and the corresponding sufficient condition in Eq. (14) by adjusting f with the given candidate images. To achieve this, we update f so that tighter evaluation of Eq. (17) and Eq. (18) is attained with given candidate images while maintaining the performance of CBIR.

Let $C = \{c_i | c_i \in X\}_{i=1}^{N}$ be the set of candidate images. Let f_0 be the pre-trained feature extractor before fine-tuning. Then, our objective function for fine-tuning is given as follows:

$$\min_f \sum_{c_1,c_2\in C} \Big(\kappa \cdot d(f_0(c_1), f(c_1)) + (1-\kappa)\cdot \mathrm{Reg}(c_1, c_2)\Big). \tag{21}$$

The first term maintains the accuracy of the CBIR by ensuring that the difference between the features calculated by f and f_0 is small. The second term is a regularization term to encourage that the upper and lower bound of $\|f(c_1+\delta) - f(c_2)\|_2$ are close to $\|f(c_1) - f(c_2)\|_2$. $\kappa \in [0,1]$ is a hyperparameter to adjust the trade-off between the performance of CBIR and (α, ϵ)-robustness against CA. We call fine-tuning with Eq. (21) as Fine-tuning to Candidates with Tighter Bounds (FCTB).

5. Experiments

In this section, we evaluate our proposed robustness training (TBT and FCTB), in terms of CBIR accuracy on clean images and robustness against QA and CA. The robustness is evaluated by *empirical robustness*, which is the accuracy of CBIR on the generated AXs, and *certified robustness*, which represents how often CBIR achieves (α, ϵ)-robustness for given inputs by our robustness verification algorithm Eq. (19).

5.1. Experimental Setting

5.1.1 Datasets

We use MNIST [11], Fashion-MNIST (FMNIST) [32], and CIFAR10 [10] for our evaluations. These datasets consist of an training and test set annotated with labels. We train feature extractors f on the training sets and evaluate f using the test set. Let $Q = \{(q_i, y_{q_i})\}_{i=1}^{|Q|}$ and $C = \{(c_i, y_{c_i}) \in X\}_{i=1}^{|C|}$ be the annotated set of query and candidate images, respectively. We randomly select Q and C without duplication from the test set so that $|Q| = 1000$ and $|C| = 1000$. Pixel values of images in all datasets are in $[0, 1]$.

5.1.2 Evaluation Measures

Performance of CBIR. To evaluate the performance of CBIR, we use Recall@K, which is one of the evaluation measures for CBIR [18, 27]. Recall@K evaluates whether how often any of the top K candidates is similar to the query image. For evaluation purpose, images belonging to the same class are regarded as similar images. Then, Recall@K is defined as follos:

$$\frac{1}{|Q|}\sum_{(q_i,y_{q_i})\in Q} \begin{cases} 1 \ \text{if} \ \exists (c, y_c) \in \mathrm{IR}_f(q_i, C)_{\le K} \\ \qquad\qquad\qquad\qquad \text{s.t.}\ y_c = y_{q_i} \\ 0 \ \text{otherwise.} \end{cases} \tag{22}$$

Empirical Robustness. To evaluate the empirical robustness against QA and CA, we extend recall@K and define empirical robust Recall@K (ER-Recall@K) against QA and CA. ER-Recall@K against QA represents how often any of the top K candidates is similar to the query image under QA:

$$\frac{1}{|Q|}\sum_{(q_i,y_{q_i})\in Q} \begin{cases} 1 \ \text{if} \ \exists (c, y_c) \in \mathrm{IR}_f(q_i+\delta_i, C)_{\le K} \\ \qquad\qquad\qquad\qquad \text{s.t.}\ y_c = y_{q_i} \\ 0 \ \text{otherwise} \end{cases} \tag{23}$$

where $\delta_1, ..., \delta_{|Q|}$ are adversarial perturbations generated with Eq.(2). We randomly select a single target candidate image $C_t = \{(c_t, y_{c_t})\} \subset C$ such that $y_{c_t} \neq y_{q_i}$ for each $(q_i, y_{q_i}) \in Q$. We minimize Eq.(2) by using PGD [15] with the step size of $\frac{\epsilon}{10}$ and the number of updates of 100, where $\epsilon \in \{0.1, 0.2\}$ for MNIST and FMNIST and $\epsilon \in \{\frac{2}{255}, \frac{3}{255}\}$ for CIFAR10, respectively.

ER-Recall@K against CA represents how often any of the top K candidates is similar image to the query image under CA:

$$\frac{1}{|Q|} \sum_{(q_i, y_{q_i}) \in Q} \begin{cases} 1 \text{ if } \exists (c, y_c) \in \mathrm{IR}_f(q, C \setminus C_s \cup \tilde{C}_s)_{\leq K} \\ \qquad \text{s.t. } y_c = y_{q_i} \\ 0 \text{ otherwise.} \end{cases} \tag{24}$$

where $C_s \subset C$ is a set of source candidate images, and $\tilde{C}_s = \{(c_i + \delta_i, y_{c_i}) | (c_i, y_{c_i}) \in C_s\}_{i=1}^{|C_s|}$ is the set of images obtained by adding adversarial perturbation $\delta_1, ..., \delta_{|C_s|}$ to each image in C_s with Eq.(4). We randomly select 100 source candidate images $C_s = \{(c_i, y_{c_i})\}_{i=1}^{100}$ such that $y_{c_i} \neq y_{q_i}$ for each $(q_i, y_{q_i}) \in Q$. We minimize Eq.(4) using PGD with the same step and perturbation size as the QA.

Certified Robustness. To evaluate the certified robustness, we define an extension of recall@K, certified robust Recall@K (CR-Recall@K). Given a set of query images, this measure evaluates how often (i) the retrieved candidate image by the query image has certified robustness against QA or CA, and (ii) are similar to the query image:

$$\frac{1}{|Q|} \sum_{(q, y_q) \in Q} \begin{cases} 1 \text{ if } \exists (c, y_c) \in \mathrm{IR}_f(q, C)_{\leq K} \text{ s.t. } y_c = y_q \\ \qquad \wedge \, \mathsf{Verify}_{\epsilon,\alpha}(f, q, C, \mathrm{Rank}_f(q, c, C)) \\ 0 \text{ otherwise.} \end{cases} \tag{25}$$

where $\mathsf{Verify}_{\epsilon,\alpha}(f, q, C, \mathrm{Rank}_f(q, c, C))$ is defined by Eq. (19). We use $\alpha = K - \mathrm{Rank}_f(q, c, C)$ for each $c \in \mathrm{IR}_f(q, C)_{\leq K}$. Then, $\mathsf{Verify}_{\epsilon,\alpha}(f, q, C, \mathrm{Rank}_f(q, c, C))$ verifies whether c is still included in $\mathrm{IR}_f(q, C)_{\leq K}$ under QA and CA, and CR-Recall@K is a lower bound of ER-Recall@K. Due to page limitations, only the results for $\epsilon = 0.2$ and $\epsilon = \frac{2}{255}$ are shown here. The results for $\epsilon = 0.1$ and $\epsilon = \frac{3}{255}$ are included in Appendix.

5.1.3 Comparison Methods

We compare our proposed robustness training Eq. (20) (TBT) and Eq. (21) (FCTB) with three existing methods: (i) triplet Loss (Triplet) [19], (ii) anti-collapse triplet (ACT), which is an adversarial training for CBIR to improve empirical robustness [40], (iii) robust training for classification using interval bound propagation (C-IBP) to improve certified robustness of classification task [6].

We use Triplet as a baseline which does not have any mechanism for robustness. We compare TBT and FCTB with ACT to show that adversarial training is not sufficient to improve certified robustness of CBIR. We also compare TBT and FCTB with C-IBP to show that robust training for improving certified robustness for the classification task is inadequate to improve certified robustness for CBIR. Detail of each method are explained in Appendix.

5.1.4 Architectures and Training Hyper Parameters

Architectures. In our experiments, we train the feature extractor f of embedding dimensionality 128 in two different model architectures, as shown in Appendix. We refer to each model as Small (3-layer CNN) and Large (6-layer CNN), respectively. Due to page limitations, the results for Small are included in Appendix.

Hyperparameters. The total number of training epochs is 100 for MNIST and FMNIST and 200 for CIFAR10. We use the Adam optimizer [9] with a batch size of 100 and an initial learning rate of 0.001. We decay the learning rate by times 0.1 at 25 and 42 epochs for MNIST and FMNIST and times 0.5 every 10 epochs between 130 and 200 epochs for CIFAR10. The margin of triplet loss is set to $m = 1.0$.

When training with TBT, to stabilize training, we use scheduling strategy for ϵ and κ proposed in [6]. Specifically, ϵ is gradually increased from 0.0 to ϵ_e, and the κ is gradually decreased from 1.0 to κ_e. We use $\epsilon_e = 0.2$ for MNIST and FMNIST, and $\epsilon_e = \frac{2}{255}$ for CIFAR10, respectively. We use $\kappa_e = 0.5$ for all datasets. Then, we linearly increase ϵ and decrease κ between $2K$ and $10K$ steps. The results of other κ_e are shown in Appendix.

When training with FCTB, we fine-tune the pre-trained feature extractor with TBT. We set fixed ϵ to 0.2 for MNIST and FMNIST and $\frac{2}{255}$ for CIFAR10. We set fixed κ to 0.2 for MNIST and 0.1 for FMNIST and CIFAR10. Other hyperparameters are shown in Appendix.

5.2. Results

Table 1 and Table 2 show the results of Recall@K, and ER-Recall@K, and CR-Recall@K for Large. We can see that TBT has less Recall@K than Triplet, ACT, and C-IBP from Table 1. This is presumably due to the fact that the diversity of feature representation is reduced by making the upper and lower bound evaluated tighter. However, the gap in Recall@K between TBT and the existing methods becomes smaller as K increases. Thus, that is not a practical problem in situations where K is large.

From Table 2, we can confirm both ER-Recall@K and CR-Recall@K of Triplet are significantly lower than the other methods. C-IBP and ACT achieve higher ER-Recall@K than Triplet, while their CR-Recall@K is zero or nearly zero, even with larger K. This implies that C-IBP and ACT cannot help to provide certified robustness of CBIR. This is because ACT is training to improve empirical robustness, which is no enough to improve certified robustness. We also conjecture that C-IBP is not sufficient to

Table 1: Comparison of Recall@K (Large). Each value is rounded off to two decimal places.

	MNIST				FMNIST				CIFAR10			
K	1	10	20	40	1	10	20	40	1	10	20	40
Triplet	0.99	1.00	1.00	1.00	0.89	0.98	0.99	0.99	0.58	0.93	0.97	0.99
ACT	0.99	1.00	1.00	1.00	0.83	0.97	0.98	0.99	0.63	0.93	0.96	0.99
C-IBP	0.97	0.99	1.00	1.00	0.75	0.96	0.98	0.99	0.39	0.87	0.94	0.98
TBT	0.94	0.98	0.99	0.99	0.62	0.93	0.97	0.98	0.18	0.81	0.93	0.97
TBT+FCTB	0.93	0.98	0.98	0.99	0.64	0.94	0.97	0.98	0.19	0.82	0.93	0.97

Table 2: Comparison of empirical robust (ER) Recall@K and certified robust (CR) Recall@K (Large). QA and CA represents query attack and candidate attack, respectively. For calculating ER-Recall@K and CR-Recall@K, we use $\epsilon = 0.2$ (MNIST and FMNIST) and $\epsilon = \frac{2}{255}$ (CIFAR10). Each value is rounded off to two decimal places.

		ER-Recall@K (QA)				CR-Recall@K (QA)				ER-Recall@K (CA)				CR-Recall@K (CA)			
K		1	10	20	40	1	10	20	40	1	10	20	40	1	10	20	40
MNIST	Triplet	0.00	0.05	0.09	0.14	0.00	0.00	0.00	0.00	0.21	0.38	0.47	0.58	0.00	0.00	0.00	0.00
	ACT	0.97	0.99	1.00	1.00	0.00	0.00	0.00	0.00	0.98	0.99	1.00	1.00	0.00	0.00	0.00	0.00
	C-IBP	0.97	0.99	1.00	1.00	0.00	0.00	0.00	0.01	0.96	0.99	0.99	1.00	0.00	0.00	0.00	0.00
	TBT	0.92	0.98	0.98	0.99	0.03	0.31	0.47	0.65	0.93	0.98	0.99	0.99	0.01	0.42	0.78	0.95
	TBT+FCTB	0.92	0.97	0.98	0.99	0.03	0.30	0.45	0.64	0.92	0.98	0.98	0.99	0.02	0.48	0.82	0.96
FMNIST	Triplet	0.00	0.09	0.14	0.20	0.00	0.00	0.00	0.00	0.04	0.11	0.14	0.19	0.00	0.00	0.00	0.00
	ACT	0.74	0.95	0.97	0.98	0.00	0.00	0.00	0.00	0.57	0.92	0.96	0.99	0.00	0.00	0.00	0.00
	C-IBP	0.71	0.96	0.98	0.99	0.00	0.01	0.04	0.08	0.67	0.95	0.98	0.99	0.00	0.00	0.00	0.03
	TBT	0.59	0.93	0.97	0.98	0.02	0.20	0.30	0.45	0.55	0.93	0.97	0.98	0.00	0.07	0.24	0.64
	TBT+FCTB	0.60	0.93	0.97	0.99	0.02	0.22	0.34	0.44	0.55	0.93	0.96	0.98	0.00	0.09	0.25	0.62
CIFAR10	Triplet	0.19	0.70	0.82	0.89	0.00	0.00	0.00	0.00	0.00	0.09	0.25	0.58	0.00	0.00	0.00	0.00
	ACT	0.47	0.88	0.94	0.97	0.00	0.00	0.00	0.00	0.22	0.72	0.88	0.96	0.00	0.00	0.00	0.00
	C-IBP	0.40	0.87	0.94	0.98	0.00	0.04	0.10	0.23	0.35	0.84	0.93	0.98	0.00	0.02	0.08	0.21
	TBT	0.20	0.79	0.92	0.97	0.02	0.18	0.31	0.48	0.15	0.78	0.92	0.96	0.00	0.19	0.34	0.58
	TBT+FCTB	0.19	0.81	0.93	0.97	0.02	0.20	0.34	0.48	0.17	0.80	0.92	0.97	0.01	0.21	0.44	0.70

tighten Eq. (17) and Eq. (18) since it aims at tightening the upper and lower bounds of logits. In contrast, TBT achieves significantly higher CR-recall@K, particularly when K is large. This is because TBT can tighten Eq. (17) and Eq. (18) successfully.

From Table 1 and Table 2, we can also confirm that fine-tuning pre-trained feature extractor with TBT to candidate images (TBT+FCTB) improve CR-Recall@K while maintaining Recall@K. This implies that FCTB can further reduce the gap between Definition 3 and the corresponding sufficient condition in Eq. (14).

6. Limitations

A drawback of our certified defense is that it does not scale to high-resolution images, which require advanced architecture. We train feature extractor with TBT using CUB-200-2011 [25] (image size is 224×224) and VGG architecture [20]. The detail of experimental settings is explained in Appendix. As a result, its training collapses, which means that the trained feature extractor returns the same value for all test inputs. This is because IBP provides very loose bounds for advanced deep architectures, resulting in extremely large regularization terms in Eq.(20). We also obtain the same results when training a feature extractor with C-IBP. Developing a certified defense for CBIR that scales to high-resolution images is a future research direction.

7. Conclusion

In this study, we proposed a certified defense for CBIR. Our certified defense improves the certified robustness of CBIR, which guarantees that no AX that largely changes the ranking of CBIR exists around the query or candidate images. To the best of our knowledge, this is the first paper on certified defense for CBIR.

Acknowledgment

This work is partly supported by Japan science and technology agency (JST), CREST JPMJCR21D3, and Japan society for the promotion of science (JSPS), Grants-in-Aid for Scientific Research 22H00521 and 19H04164.

References

[1] Song Bai, Yingwei Li, Yuyin Zhou, Qizhu Li, and Philip HS Torr. Adversarial metric attack and defense for person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):2119–2126, 2020.

[2] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[3] Wei Chen, Yu Liu, Weiping Wang, Erwin M. Bakker, Theodoros Georgiou, Paul W. Fieguth, Li Liu, and Michael S. Lew. Deep image retrieval: A survey. *arXiv preprint arXiv:2101.11282*, 2021.

[4] Shiv Ram Dubey. A decade survey of content based image retrieval using deep learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[6] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*, 2018.

[7] Sven Gowal, Krishnamurthy Dj Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4842–4851, 2019.

[8] Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks. In *International Conference on Computer Aided Verification*, pages 97–117. Springer, 2017.

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[11] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[12] Jie Li, Rongrong Ji, Hong Liu, Xiaopeng Hong, Yue Gao, and Qi Tian. Universal perturbation attack against image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4899–4908, 2019.

[13] Linyi Li, Xiangyu Qi, Tao Xie, and Bo Li. Sok: Certified robustness for deep neural networks. *arXiv preprint arXiv:2009.04131*, 2020.

[14] Xiaodan Li, Jinfeng Li, Yuefeng Chen, Shaokai Ye, Yuan He, Shuhui Wang, Hang Su, and Hui Xue. Qair: Practical query-efficient black-box attacks for image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3330–3339, 2021.

[15] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[16] Iacopo Masi, Yue Wu, Tal Hassner, and Prem Natarajan. Deep face recognition: A survey. In *2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pages 471–478. IEEE, 2018.

[17] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning (ICML)*, pages 4970–4979. PMLR, 2019.

[18] Elias Ramzi, Nicolas Thome, Clément Rambour, Nicolas Audebert, and Xavier Bitot. Robust and decomposable average precision for image retrieval. *Advances in Neural Information Processing Systems*, 34, 2021.

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[21] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–30, 2019.

[22] Giorgos Tolias, Filip Radenovic, and Ondrej Chum. Targeted mismatch adversarial attack: Query with a flower to retrieve the tower. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5037–5046, 2019.

[23] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in Neural Information Processing Systems*, 33:1633–1645, 2020.

[24] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. *Advances in neural information processing systems*, 31, 2018.

[25] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[26] Hongjun Wang, Guangrun Wang, Ya Li, Dongyu Zhang, and Liang Lin. Transferable, controllable, and inconspicuous adversarial attacks on person re-identification with deep mis-ranking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 342–351, 2020.

[27] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *Proceedings of the IEEE international conference on computer vision*, pages 2593–2601, 2017.

[28] Zhibo Wang, Siyan Zheng, Mengkai Song, Qian Wang, Alireza Rahimpour, and Hairong Qi. advpattern: physical-world attacks on deep person re-identification via adversarially transformable patterns. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8341–8350, 2019.

[29] Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon.

Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning*, pages 5276–5285. PMLR, 2018.

[30] Lily Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning*, pages 5276–5285. PMLR, 2018.

[31] Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel. Evaluating the robustness of neural networks: An extreme value theory approach. *arXiv preprint arXiv:1801.10578*, 2018.

[32] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[33] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

[34] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[35] Huan Zhang, Hongge Chen, Chaowei Xiao, Sven Gowal, Robert Stanforth, Bo Li, Duane Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural networks. In *International Conference on Learning Representations*, 2019.

[36] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. *Advances in neural information processing systems*, 31, 2018.

[37] Huan Zhang, Pengchuan Zhang, and Cho-Jui Hsieh. Recurjac: An efficient recursive algorithm for bounding jacobian matrix of neural networks and its applications. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5757–5764, 2019.

[38] Mo Zhou, Zhenxing Niu, Le Wang, Qilin Zhang, and Gang Hua. Adversarial ranking attack and defense. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 781–799. Springer, 2020.

[39] Mo Zhou, Le Wang, Zhenxing Niu, Qilin Zhang, Yinghui Xu, Nanning Zheng, and Gang Hua. Practical relative order attack in deep ranking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16413–16422, 2021.

[40] Mo Zhou, Le Wang, Zhenxing Niu, Qilin Zhang, Nanning Zheng, and Gang Hua. Adversarial attack and defense in deep ranking. *arXiv preprint arXiv:2106.03614*, 2021.