arXiv:2303.12658v1 [cs.CV] 22 Mar 2023

基于深度哈希的检索中对抗鲁棒性的可靠且高效评估

Xunguang Wang[1], Jiawang Bai[2],Xinyue Xu[1],
Xiaomeng Li[1]香港科技大学[2]清华大学

xunguangwang@gmail.com, bjw19@mails.tsinghua.edu.cn, xxcub@ust.hk, eexmli@ust.hk

摘要

深度哈希因其高效性和有效性，已广泛应用于大规模图像检索。近年来，研究者提出了多种对抗攻击，以揭示深度哈希模型面对对抗样本时的脆弱性。然而，现有攻击方法要么性能下降，要么效率低下，因为它们未能充分利用原始样本之间的语义关系，或是耗费大量时间通过深度神经网络学习这些关系。本文提出一种新颖的 *Pharos* 引导攻击，简称 ***PgA***，用于可靠、高效地评估深度哈希网络的对抗鲁棒性。具体而言，我们设计了 *pharos* 码来表示良性图像的语义，该编码既能保留其与语义相关样本的相似度，又能保持其与不相关样本的差异性。我们证明，可以通过一个简单的数学公式快速计算 *pharos* 码。因此，*PgA* 能够通过最大化对抗样本的哈希码与 *pharos* 码之间的相似度，直接对基于深度哈希的检索实施可靠且高效的攻击。在基准数据集上开展的大量实验表明，所提出的算法在攻击强度和速度方面均优于现有最先进方法。

1. 引言

在当今大数据时代，如何从海量数据中快速、有效地检索所需信息是一项挑战。哈希学习（哈希）[41] 因其能够将高维数据映射为紧凑的二进制码，从而高效执行异或运算且存储成本低，在大规模图像检索领域受到广泛关注。尤其是，深度哈希 [6, 26, 47] 利用深度神经网络（深度神经网络）学习非线性哈希函数，因其检索准确率优于传统哈希方法，已成为主流的图像搜索技术。

近期研究 [2, 29, 43, 44, 48, 49, 52] 表明深度哈希模型容易受到对抗样本的攻击。尽管这些难以察觉的样本是通过对原始样本添加微小扰动而生成的，但它们足以诱骗模型做出不准确的预测。毫无疑问，此类恶意攻击给基于深度哈希的图像检索系统带来了严重的安全威胁。例如，在基于深度哈希的人脸识别系统中，对抗样本可能会误导系统，将数据库中特定个体的面孔匹配起来，从而有效渗透系统。因此，针对基于深度哈希的检索中的这些安全问题开展研究的需求十分迫切。

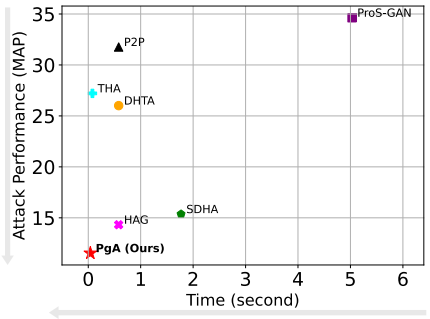


图1. 深度哈希模型在对抗攻击下的攻击性能。我们采用检索指标，即平均精度均值（平均精度均值）[49] 来衡量攻击性能。横轴表示构造每个对抗样本的平均时间（秒）。我们的方法最接近坐标原点，效率最高。

目前，已有一些研究 [2, 8, 29, 43, 44, 49] 针对基于深度哈希的检索中的对抗攻击与对抗防御展开研究。然而，目前尚无可靠且高效的攻击算法可作为基准，用于评估基于哈希的检索中针对对抗攻击的防御性能。一种高效且强大的对抗攻击方法能够为模型鲁棒性提供合理的基准，并促进对抗防御策略的发展。尽管现有攻击技术揭示了深度哈希网络在处理对抗样本时的脆弱性，但它们在评估深度哈希模型的对抗鲁棒性

（即，防御性能）方面既不可靠也不高效。首先，这些方法 [2, 29, 49] 由于未充分利用可用样本之间的语义相关性，因此攻击性能有限。例如，SDHA[29]仅降低对抗样本与语义相关图像之间的相似度，而忽略了其他更不相关的图像。其次，尽管一些哈希攻击方法同时考虑了相似对和不相似对，但它们使用耗时的神经网络从这些样本对中学习具有判别性的语义表示，以实现精确的语义攻击，例如，ProS-GAN [44] 和 THA [43]。本文重点改进以往哈希攻击在有效性和效率方面的不足，如图 1所示。

在本研究中，我们提出了 Pharos 引导攻击 (PgA)，用于对深度哈希网络进行可靠且高效的对抗鲁棒性评估。其核心思想是快速计算能够反映原始图像语义的 pharos 码，然后利用该编码引导生成强大的对抗样本。具体而言，我们首先设计了一种最优哈希码（即 *pharos* 码），作为良性图像语义的判别性表征；该编码与正样本的哈希码保持相似，与负样本的哈希码保持不相似（正样本和负样本分别表示在语义上与良性图像相关和不相关的样本）。利用哈希码的二值特性，我们证明，所提出的 *Pharos* 生成方法 (PGM) 可以通过一个简单的数学公式直接计算 pharos 码（参见定理 1）。因此，可在对抗攻击开始前立即计算出输入数据的 pharos 码。随后，基于 pharos 码，通过最大化对抗样本的哈希码与 pharos 码之间的汉明距离，即可实施高效的对抗哈希攻击。得益于 pharos 码的优越性，如图 1所示，我们的攻击方式能够显著提升对抗鲁棒性验证的有效性和效率。总之，我们的主要贡献如下：

- 我们构建 pharos 码，作为原始图像内容的精确语义表征，以助于构建面向基于深度哈希的检索的对抗攻击框架。需要强调的是，我们在 PGM 中证明的数学公式可以即时生成 pharos 码。
- 我们提出了一种简单的 Pharos 引导攻击算法，即 PgA。它是一种可靠且高效的方法，可用于评估深度哈希网络的对抗鲁棒性。
- 大量实验表明，PgA 可应用于深度哈希框架，并以高效率实现最先进的攻击性能。

2. 相关工作

2.1. 基于深度哈希的图像检索

随着深度学习在诸多视觉任务中取得显著成功，[11, 14, 16, 38, 40]，深度哈希方法已在大规模图像检索领域得到充分发展，其性能优于基于手工设计特征的传统哈希方法。开创性的 CNNH [47] 采用两阶段策略，即先对训练数据进行哈希码生成，再利用深度神经网络构建哈希函数。近年来，深度哈希方法 [5, 6, 12, 19, 23, 25, 26, 28, 36, 45, 53] 着重于在端到端深度神经网络中联合进行特征学习和哈希码编码，以提升哈希码的质量。DPSH 是一项具有代表性的工作，[26], 它通过成对相似性损失，同时学习数据点的视觉特征并保留其语义相似性。为缓解正负样本对之间的数据不平衡问题，HashNet [6] 在成对损失函数中采用加权策略。与成对相似性学习不同，CSQ [50]通过使哈希码接近预定义的哈希中心，生成高质量的哈希码。

2.2. 对抗攻击

在图像分类领域，自发现对抗样本的独特性质 [4, 37] 以来，研究者已提出了大量对抗攻击方法 [7, 9, 13, 15, 18, 22, 30, 31, 33, 37]，通过构造对抗样本来欺骗经过充分训练的分类器。例如，FGSM [15] 通过沿梯度方向以较大步长最大化损失来生成对抗样本。作为 FGSM 的多步变体，I-FGSM [22] 和 PGD [30] 以较小步长迭代更新扰动，以获得更好的攻击性能。

近年来，研究人员已将对抗攻击扩展到基于深度哈希的图像检索 [1, 2, 29, 42–44、49，52]。现有的深度哈希对抗攻击方法可分为两类：非定向攻击 和定向攻击。对于基于哈希的检索中的非定向攻击，其目标是生成对抗样本，使哈希模型产生混淆，从而检索到与原始图像无关的结果 [2, 44]。Yang 通过最小化对抗样本与原始样本之间的哈希码相似度来实现非定向攻击等人 [49] 提出了**HAG**，这是首个针对深度哈希的对抗攻击方法。**SDHA** [29] 由于避开了良性样本的相关图像，因此能够生成更有效的对抗查询，而 HAG 仅考虑原始图像。至于定向攻击，其目标是构造对抗样本，使检索到的图像在语义上与给定的目标标签相关 [2, 44]。为实现定向攻击，**P2P** 和**DHTA** [2] 将锚点码作为目标标签的代表，以引导对抗样本的生成

样本。随后，Wang 等人 [43] 将原型码定义为目标码，以实现更好的定向攻击，本文称之为**THA** 。**ProS-GAN** [44]设计了一种生成框架，用于在测试阶段高效地实施定向哈希攻击。与上述白盒场景不同，Xiao 等人 [48] 提出了定向黑盒攻击 NAG，通过增强对抗样本的可迁移性来实现攻击。

不同于以往工作 [2] 通过少量语义相同的实例获得锚点码，我们提出了 pharos 码，它既保留了与相关样本哈希码的语义相似度，也保留了与不相关样本哈希码的差异性。此外，我们使用经过验证的数学公式（即 PGM），在对抗攻击前即时计算 pharos 码，而不像 ProS-GAN 和 THA 那样，通过耗时的神经网络学习原型码 [43, 44] 。因此，我们的 pharos 码更适用于高效评估深度哈希模型的对抗鲁棒性。

2.3. 对抗训练

对抗训练 [15, 30] 旨在通过生成的对抗样本扩充训练数据，这是应对各种对抗攻击最具鲁棒性的训练策略。因此，人们提出了对抗训练的各种改进 [32, 46, 51] 和应用 [3, 24, 39] ，以提升深度神经网络的鲁棒性和泛化能力。针对基于深度哈希的检索，[43] 提出了首个有效的对抗训练算法，该算法基于目标攻击（此处称为**ATRDH** ），通过缩小对抗样本与原始样本在汉明空间中的语义差距来实现。

3. 方法

3.1. 预备知识

我们假设被攻击的哈希模型 F 从包含 N 个数据点 $O = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$ 的训练集中学习，其中 $\boldsymbol{x}_i$ 表示第 i 张图像，$\boldsymbol{y}_i = [y_{i1}, y_{i2}, ..., y_{iC}] \in \{0,1\}^C$ 表示 $\boldsymbol{x}_i$ 的标签向量。C 表示数据集中的类别总数。$y_{ij} = 1$ 表示 $\boldsymbol{x}_i$ 属于第 j 类。如果 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 至少共享一个标签，则它们在语义上相似，即，$\boldsymbol{x}_j$ 是 $\boldsymbol{x}_i$ 的正样本。否则，它们在语义上不相似，且 $\boldsymbol{x}_j$ 是 $\boldsymbol{x}_i$ 的负样本。

深度哈希旨在利用深度神经网络将高维数据转换为紧凑的二进制码，同时保留其语义相似性。对于给定的哈希模型 F，实例 $\boldsymbol{x}_i$ 的哈希码 $\boldsymbol{b}_i$ 生成如下：

$$\begin{aligned} \boldsymbol{b}_i = F(\boldsymbol{x}_i) = \operatorname{sign}(\boldsymbol{h}_i) = \operatorname{sign}(f_\theta(\boldsymbol{x}_i)), \\ s.t.\ \boldsymbol{b}_i \in \{-1, 1\}^K, \end{aligned} \tag{1}$$

其中，K 表示哈希码长度，带有参数 θ 的 $f(\cdot)$ 是一个用于近似哈希函数 $F(\cdot)$ 的深度神经网络。最终的二进制编码 $\boldsymbol{b}_i$ 通过对 $f_\theta(\boldsymbol{x}_i)$ 的输出 $\boldsymbol{h}_i$ 应用符号函数 sign(·) 得到。通常，$f(\cdot)$ 由卷积神经网络（CNN）实现，并在输出层采用 tanh 激活函数来模拟符号函数。

3.2. 所提出的 Pharos 引导攻击

3.2.1 问题形式化

在基于哈希的检索中，对抗攻击（即，非定向攻击）的目标是构造一个对抗样本，使其检索结果与原始样本内容无关。为确保可信度，可以通过最大化对抗样本与其语义相关样本之间的哈希码距离，同时最小化其与不相关样本之间的距离来实现这一目标，而非仅仅最小化与良性样本之间的距离。因此，对于给定的干净图像 $\boldsymbol{x}$，其对抗样本 $\boldsymbol{x}'$ 的目标可表述如下：

$$\begin{aligned} \max_{\boldsymbol{x}'} \sum_{i}^{N_\mathrm{p}} \sum_{j}^{N_\mathrm{n}} [w_i D(F(\boldsymbol{x}'), F(\boldsymbol{x}_i^{(\mathrm{p})})) - w_j D(F(\boldsymbol{x}'), F(\boldsymbol{x}_j^{(\mathrm{n})}))], \\ s.t.\ \|\boldsymbol{x} - \boldsymbol{x}'\|_p \le \epsilon, \end{aligned} \tag{2}$$

其中，$F(\cdot)$ 是由深度模型 $f(\cdot)$ 近似得到的哈希函数，$D(\cdot,\cdot)$ 是距离度量。w_i 和 w_j 表示距离权重。$\boldsymbol{x}_i^{(\mathrm{p})}$ 是与原始样本 $\boldsymbol{x}$ 在语义上相关的正样本，而 $\boldsymbol{x}_j^{(\mathrm{n})}$ 是 $\boldsymbol{x}$ 的负样本。由于式（2）的这一最大化项可以使对抗样本的哈希码接近不相关样本的哈希码，并远离语义相关样本，因此理论上可以实现最优攻击强度。N_p 和 N_n 分别是正样本和负样本的数量。$\|\cdot\|_p\ (p = 1, 2, \infty)$ 是 L_p 范数，它将对抗样本与原始样本之间的像素差异限制在不超过 ϵ，以确保对抗扰动具有不可感知性。

3.2.2 Pharos 编码的生成

实际上，式（2）中的最大化目标等价于在哈希中寻找一个哈希码 $\boldsymbol{b}'$，使其满足：

$$\max_{\boldsymbol{b}'} \sum_{i} \sum_{j} [w_i D_\mathrm{H}(\boldsymbol{b}', \boldsymbol{b}_i^{(\mathrm{p})}) - w_j D_\mathrm{H}(\boldsymbol{b}', \boldsymbol{b}_j^{(\mathrm{n})})], \tag{3}$$

其中，D_H 表示汉明距离度量。$\boldsymbol{b}_i^{(\mathrm{p})}$ 是正样本 $\boldsymbol{x}_i^{(\mathrm{p})}$ 的哈希码，而 $\boldsymbol{b}_j^{(\mathrm{n})}$ 是负样本 $\boldsymbol{x}_j^{(\mathrm{n})}$ 的二进制编码，即，$\boldsymbol{b}_i^{(\mathrm{p})} = F(\boldsymbol{x}_i^{(\mathrm{p})})$ $\boldsymbol{b}_j^{(\mathrm{n})} = F(\boldsymbol{x}_j^{(\mathrm{n})})$。随后，我们可以通过最小化与 $\boldsymbol{b}'$的距离来优化对抗样本，即，

$$\min_{\boldsymbol{x}'} D_\mathrm{H}(\boldsymbol{b}', F(\boldsymbol{x}')). \tag{4}$$

对于任意哈希码$\hat{\boldsymbol{b}}$和$\check{\boldsymbol{b}}$，我们知道 $D_{\mathrm{H}}(\hat{\boldsymbol{b}},\check{\boldsymbol{b}}) = \frac{1}{2}(K - \hat{\boldsymbol{b}}^{\top}\check{\boldsymbol{b}})$。据此可推得 $D_{\mathrm{H}}(\hat{\boldsymbol{b}},\check{\boldsymbol{b}}) = K - D_{\mathrm{H}}(-\hat{\boldsymbol{b}},\check{\boldsymbol{b}})$。设哈希码为 $\boldsymbol{b}^{\star} = -\boldsymbol{b}'$，式（3）和（4）可改写如下：

$$\min_{\boldsymbol{b}^{\star}} \sum_i \sum_j \{w_i[D_{\mathrm{H}}(\boldsymbol{b}^{\star},\boldsymbol{b}_i^{(\mathrm{p})}) - K] - w_j[D_{\mathrm{H}}(\boldsymbol{b}^{\star},\boldsymbol{b}_j^{(\mathrm{n})}) - K]\},$$
$$\max_{\boldsymbol{x}'} D_{\mathrm{H}}(\boldsymbol{b}^{\star}, F(\boldsymbol{x}')) - K. \tag{5}$$

去掉常数项后，公式（5）可写为：

$$\min_{\boldsymbol{b}^{\star}} \sum_i \sum_j [w_i D_{\mathrm{H}}(\boldsymbol{b}^{\star},\boldsymbol{b}_i^{(\mathrm{p})}) - w_j D_{\mathrm{H}}(\boldsymbol{b}^{\star},\boldsymbol{b}_j^{(\mathrm{n})})],$$
$$\max_{\boldsymbol{x}'} D_{\mathrm{H}}(\boldsymbol{b}^{\star}, F(\boldsymbol{x}')). \tag{6}$$

由于哈希码具有二值特性，我们可以通过一个简单的数学公式，直接计算问题（6）中的最优码（称为 *pharos* 码 $\boldsymbol{b}^{\star}$），如定理 1所示。

定理 1 由公式（6）定义的 *Pharos* 码 可通过加权正哈希码与负哈希码之差的总和计算，即

$$\boldsymbol{b}^{\star} = \mathrm{sign}\Big(\sum_i^{N_{\mathrm{p}}} \sum_j^{N_{\mathrm{n}}} (w_i \boldsymbol{b}_i^{(\mathrm{p})} - w_j \boldsymbol{b}_j^{(\mathrm{n})})\Big), \tag{7}$$

我们将通过公式（7）获得 pharos 码的方法称为 Pharos 生成方法（PGM）。$\mathrm{sign}(\cdot)$ 在定理1 中表示符号函数。PGM 的证明见附录 A。此外，我们将 w_i 和 w_j 定义如下：

$$w_i = s_i, \quad w_j = 1 - s_j, \tag{8}$$

其中，$s_{i/j}$ ($s_{i/j} \in [0,1]$) 表示对抗样本与第 i/j 个良性样本之间的相似度。如果给定 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 的标签 $\boldsymbol{y}_i$ 和 $\boldsymbol{y}_j$，我们可以通过 Dice 系数计算 $s_{i/j}$，即，$s_{i/j} = \frac{2|\boldsymbol{y} \cap \boldsymbol{y}_{i/j}|}{|\boldsymbol{y}| + |\boldsymbol{y}_{i/j}|}$。否则，$s_{i/j}$ 通常由被攻击哈希模型的优化目标决定。例如，$s_i = 1$ 和 $s_j = 0$ 在哈希学习中被广泛采用。

3.2.3 生成对抗样本

找到 pharos 码后，式（2）中描述的攻击问题可转化为以下带有 L_∞ 约束的优化目标：

$$\max_{\boldsymbol{x}'} D_{\mathrm{H}}(\boldsymbol{b}^{\star}, F(\boldsymbol{x}')), \quad s.t.\ \|\boldsymbol{x} - \boldsymbol{x}'\|_\infty \le \epsilon \quad . \tag{9}$$

根据 $D_{\mathrm{H}}(\hat{\boldsymbol{b}}, \check{\boldsymbol{b}}) = \frac{1}{2}(K - \hat{\boldsymbol{b}}^{\top}\check{\boldsymbol{b}})$，公式 (9) 等价于：

$$\max_{\boldsymbol{x}'} \mathcal{L}_a = -\frac{1}{K}(\boldsymbol{b}^{\star})^{\top} f_\theta(\boldsymbol{x}'),\ s.t.\ \|\boldsymbol{x} - \boldsymbol{x}'\|_\infty \le \epsilon. \tag{10}$$

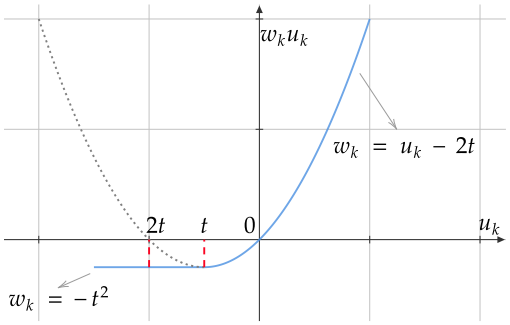


图2. ω_k的曲线。

然而，公式 (10) 关注的是$f_\theta(\boldsymbol{x}')$ 与 $\boldsymbol{b}^{\star}$之间的相似度之和，无法有效促使 $f_\theta(\boldsymbol{x}')$ 中的每一位与 $\boldsymbol{b}^{\star}$的符号不同。直观而言，$f_\theta(\boldsymbol{x}')$ 与 $\boldsymbol{b}^{\star}$ 之间差异较小的位应赋予较大的权重。因此，我们在 $\boldsymbol{\omega}$ 上添加一个加权向量 $\mathcal{L}_a$，以促使 $f_\theta(\boldsymbol{x}')$ 中的每一位都偏离 $\boldsymbol{b}^{\star}$中的对应位。形式化地，

$$\mathcal{L}_a = -\frac{1}{K}\boldsymbol{\omega}^{\top}\boldsymbol{u}, \quad \boldsymbol{u} = \boldsymbol{b}^{\star} \circ f_\theta(\boldsymbol{x}'), \tag{11}$$

其中，$\circ$ 表示 Hadamard 积，$\boldsymbol{\omega}$ 与 $\boldsymbol{b}^{\star}$具有相同的维度。为提高效率，无需强制 $\boldsymbol{u}$ 中的每个元素 u_k 都近似于 -1。在这种情况下，我们将 u_k 设为在 t （$-1 < t < 0$）处收敛。因此，$\boldsymbol{\omega}$ 的分量 $\boldsymbol{\omega}_k$ 定义为

$$\omega_k = \begin{cases} u_k - 2t, & u_k > t \\ -t^2, & \text{otherwise} \end{cases}, \tag{12}$$

其中，u_k 是 k的第 $\boldsymbol{u}$个元素。如图2所示，t 控制 $b_k^{\star}$ 与 f_θ（$\boldsymbol{x}'$）$_k$之间的间隔。 t 默认设置为-0.8 。此外，我们遵循 [49]，使我们的 pharos 引导攻击为提高效率，聚焦于 f（$\boldsymbol{x}'$）与 $\boldsymbol{b}^{\star}$ 之间不同的符号，即即，

$$\mathcal{L}_a = -\frac{1}{\pi}[\boldsymbol{m} \circ \boldsymbol{\omega}]^{\top}\boldsymbol{u}, \quad \boldsymbol{u} = \boldsymbol{b}^{\star} \circ f_\theta(\boldsymbol{x}'), \tag{13}$$

其中，$\boldsymbol{m} \in \{0,1\}^K$ 是一个掩码，π 是 $\boldsymbol{m}$中非零元素的数量。$\boldsymbol{m}$ 的元素 m_k 定义为

$$m_k = \begin{cases} 1, & u_k > t \\ 0, & \text{otherwise} \end{cases}. \tag{14}$$

值得注意的是，使用公式（10）的 Pharos 引导攻击称为 **PgA**†，而使用公式（13）的攻击是默认的**PgA**。与 HAG 和 SDHA 使用随机梯度下降 [34]或 Adam [20] 优化器[29,49] 并进行大量迭代不同，本文采用 PGD[30] 来优化 $\boldsymbol{x}'$，迭代 T（$T = 100$ 次为默认设置），以提高效率，即，

$$\boldsymbol{x}'_T = \mathcal{S}_\epsilon(\boldsymbol{x}'_{T-1} + \eta \cdot \mathrm{sign}(\Delta_{\boldsymbol{x}'_{T-1}}\mathcal{L}_a)),\ \boldsymbol{x}'_0 = \boldsymbol{x} + \boldsymbol{r}, \tag{15}$$

其中，η 为步长，$\mathcal{S}_\epsilon$ 将 $\boldsymbol{x}'$ 投影到 $\boldsymbol{x}$的 ϵ-球 [30]内。 $\boldsymbol{r}$ 为随机噪声，从均匀分布 $U(-\epsilon,\epsilon)$中采样得到。

表 1. 无防御情况下不同攻击方法对深度哈希模型的平均精度均值（%）。

方法	FLICKR-25K			NUS-WIDE			MS-COCO		
	16 位	32位	64 位	16 位	32位	64 位	16 位	32位	64 位
干净	80.39	81.35	91.91	74.99	76.72	77.84	56.53	57.93	57.92
P2P [2]	41.68	42.69	41.18	32.56	31.76	31.74	21.99	21.78	21.57
DHTA [2]	34.74	32.63	34.58	26.68	26.01	26.33	19.41	19.23	18.05
ProS-GAN [44]	67.29	76.31	82.51	30.50	34.63	64.43	52.12	53.67	53.38
THA [43]	38.28	37.54	35.67	30.72	27.21	24.27	24.62	20.65	22.71
HAG [49]	24.75	24.37	23.44	14.55	14.32	14.07	13.91	13.59	14.62
SDHA [29]	20.50	19.63	18.98	17.09	15.36	14.87	12.07	12.36	12.94
PgA† (本文方法)	16.03	15.49	15.32	12.30	11.93	12.07	11.36	10.56	11.10
PgA（本文方法）	**15.70**	**15.18**	**15.02**	**11.81**	**11.53**	**11.69**	**9.91**	**9.41**	**10.26**

表 2. ATRDH 对抗训练后多种攻击方法的平均精度均值（MAP，%） [43]。

方法	FLICKR-25K			NUS-WIDE			MS-COCO		
	16 位	32位	64 位	16 位	32位	64 位	16 位	32位	64 位
干净	72.75	73.34	73.10	66.42	65.90	68.69	49.23	50.87	49.93
P2P [2]	52.84	54.56	57.13	51.23	56.70	54.05	33.55	33.24	30.96
DHTA [2]	49.97	53.26	54.04	49.16	56.50	51.91	32.16	31.62	28.35
ProS-GAN [44]	73.33	73.33	73.89	67.07	67.18	69.20	49.59	51.30	50.61
THA [43]	48.58	49.94	50.43	50.57	50.92	50.86	32.31	30.52	29.33
HAG [49]	41.99	45.58	45.81	42.15	47.46	45.19	26.50	27.49	28.73
SDHA [29]	37.65	40.68	44.61	42.39	47.16	45.86	28.00	27.75	27.72
PgA† （我们的方法）	32.78	34.67	35.53	37.86	42.84	39.82	21.89	21.74	**22.21**
PgA（我们的方法）	**31.52**	**33.80**	**34.52**	**37.48**	**42.22**	**39.54**	**21.52**	**21.42**	22.45

4. 实验

4.1. 实验设置

数据集。 我们采用基于哈希的检索中常用的三个数据集，通过大量实验评估我们的防御方法：*FLICKR-25K* [17], *NUS-WIDE*[10] 和 *MS-COCO* [27]。**FLICKR-25K** 数据集包含 25,000 张 Flickr 图像，并带有 38 个标签。我们抽取 1,000 张图像作为查询集，其余图像作为数据库，遵循 [43]。此外，我们从数据库中随机选择 5,000 个实例来训练哈希模型。 **NUS-WIDE** 数据集包含 269,648 张图像，并标注了 81 个概念。我们选取包含最常见的 21 个概念的子集，其中包含 195,834 张图像。从该子集中抽取 2,100 张图像作为查询，其余图像作为数据库。我们从数据库中随机选择 10,500 张图像作为训练集 [44]。 **MS-COCO** 数据集由 82,783 个训练样本和 40,504 个验证样本组成，每个实例至少标注了 80 个类别中的一个。合并训练集和验证集后，我们从中随机选取 5,000 个实例作为查询，其余作为数据库。对于训练集，从数据库中随机选择 10,000 张图像。此外，我们还在 CIFAR-10 上进行了额外实验（参见附录 B）。

协议。 为了评估 PgA 的性能，我们在标准指标上进行实验，即平均精度均值（MAP）、精确率-召回率（PR）以及精确率@topN（P@N）曲线 [49]。遵循 [2], 我们根据数据库中排名前 5,000 的结果计算 MAP 值。

基线方法。 随后 [2, 49], 我们采用 DPH 作为默认的受攻击哈希方法，它是基于深度哈希的检索中的一种通用算法。我们选择 AlexNet [21] 作为默认骨干网络，在 FLICKR-25K、NUS-WIDE 和 MS-COCO 上实现哈希模型。我们还评估了本方法对由 ATRDH [43] 训练的防御模型的攻击性能（它是深度哈希领域唯一的对抗训练算法）。我们将所提出的算法与多种哈希攻击方法进行比较，包括 P2P、DHTA、ProS-GAN、THA、HAG 和 SDHA。对于定向攻击，我们随机选择一个与真实标签类别不同的标签作为目标标签。这些方法的其他细节均与原始文献一致。

实现细节。 我们使用随机梯度下降（SGD）作为目标哈希模型的优化器，初始学习率为 0.01，动量为 0.9。图像的批量大小固定为 32，权重衰减参数设为 5×10^{-4}。所有图像在输入哈希模型之前，均调整为 224×224 并在 $[0,1]$ 中进行归一化。对于所提出的攻击方法 PgA，我们采用 PGD [2] 来优化对抗样本。步长 η 和迭代次数 T 分别设为 $1/255$ 和 100。

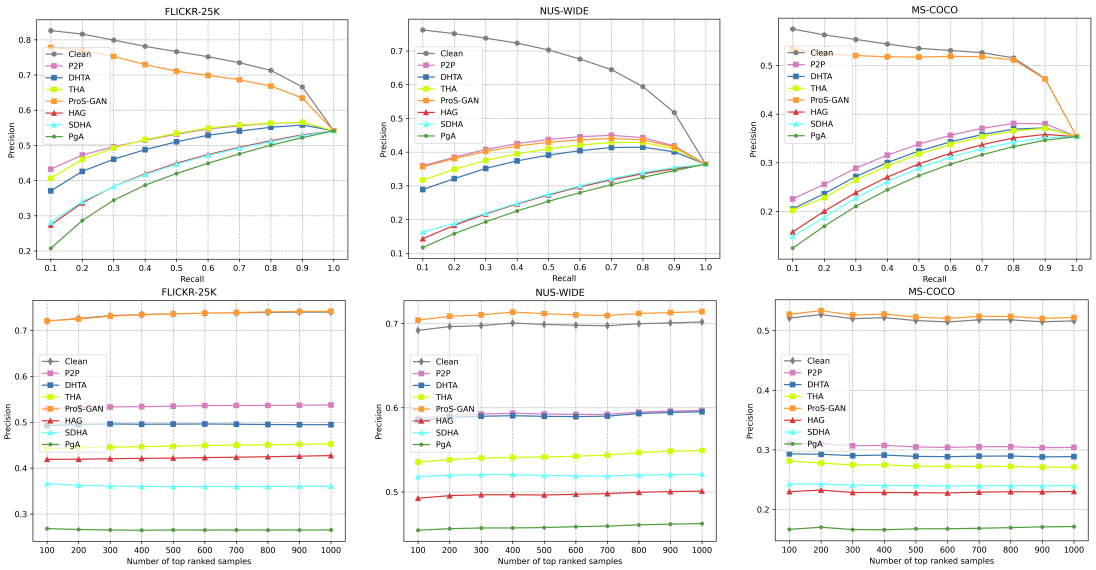


图3 。32 位码长下原始深度哈希模型的精确率-召回率和 Precision@topN 曲线 .

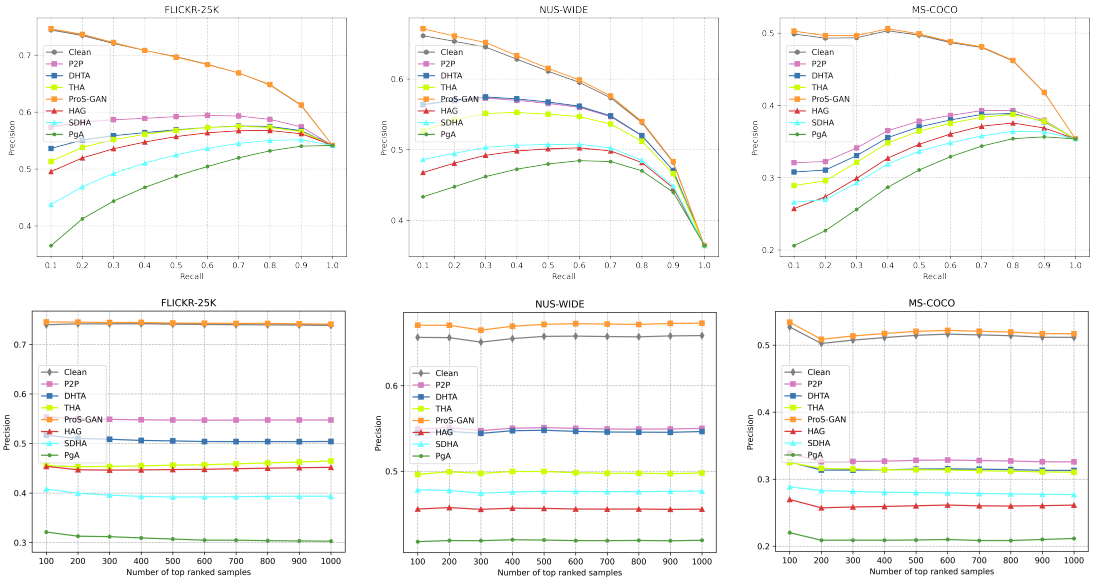


图4。32 位码长下 ATRDH 的精确率-召回率和 Precision@topN 曲线。

扰动预算 ϵ 设为 $8/255$。所有代码均基于 PyTorch 1.12 编写，并在 NVIDIA RTX 3090 GPU 上运行。

4.2. 攻击结果

表1和表2 分别展示了不同攻击方法在无防御的原始深度哈希网络和经对抗训练的模型上的攻击性能（平均精度均值）。平均精度均值越低，攻击性能越强。这些表中的“干净”表示使用良性图像进行查询，因此平均精度均值反映的是哈希模型未受攻击时的原始检索性能。从表1中可以观察到，所提出的方法当哈希位数从 16 变化到 64 时，能够大幅降低三个数据集上的平均精度均值，并且优于所有其他攻击。与表 [2], 1 中最强的定向攻击DHTA相比，我们的 PgA 在 FLICKR-25K、NUS-WIDE 和 MS-COCO 上对测试位数的平均提升分别达到 18.68%、14.66% 和 9.03%。此外，与最先进的非定向攻击 SDHA 相比，我们的方法在三个数据集上的平均表现分别优于它 4.40%、4.09% 和 2.59 %。对于由 ATRDH（深度哈希领域中唯一的对抗训练算法）训练的防御模型，表2 显示，PgA 的所有平均精度均值都低于其他攻击方法。即便面对 SDHA，所提出的 PgA 仍带来平均

表3. 哈希码长度为32位的哈希模型采用哈希攻击方法时的平均精度均值（%）和时间（秒/图像）。

方法	FLICKR-25K		NUS-WIDE		MS-COCO	
	平均精度均值	时间	平均精度均值	时间	平均精度均值	时间
P2P	42.69	0.59	31.76	0.58	21.78	0.58
DHTA	32.63	0.59	26.01	0.58	19.23	0.59
ProS-GAN	76.31	5.10	34.63	5.03	53.67	2.08
THA	37.54	0.13	27.21	0.08	20.65	0.07
HAG	24.37	0.59	14.32	0.58	13.59	0.59
SDHA	19.63	1.68	15.36	1.77	12.36	1.68
PgA†（我们的方法）	15.49	**0.04**	11.93	**0.04**	10.56	**0.04**
PgA（我们的方法）	**15.18**	**0.04**	**11.53**	**0.04**	**9.41**	**0.04**

表4. ATRDH 训练的 32位哈希模型上哈希攻击方法的平均精度均值（%）和时间（每幅图像秒数）。

方法	FLICKR-25K		NUS-WIDE		MS-COCO	
	平均精度均值	时间	平均精度均值	时间	平均精度均值	时间
P2P	54.56	0.59	56.70	0.59	33.24	0.58
DHTA	53.26	0.58	56.50	0.59	31.62	0.58
ProS-GAN	73.33	5.09	67.18	5.13	51.30	2.07
THA	49.94	0.13	50.92	0.08	30.52	0.06
HAG	45.58	0.59	47.46	0.59	27.49	0.58
SDHA	40.68	1.60	47.16	1.66	27.75	1.60
PgA†（我们的方法）	34.67	**0.04**	42.84	**0.04**	21.74	**0.04**
PgA（我们的方法）	**33.80**	**0.04**	**42.22**	**0.04**	**21.42**	**0.04**

表5. 采用32位哈希码长度的哈希模型攻击结果：平均精度均值（%）/时间（每幅图像秒数）。所有测试的攻击方法均采用PGD [30]以20步构造对抗样本（即，$T = 20$ 和 $\eta = 1/255$）。

方法	FLICKR-25K		NUS-WIDE		MS-COCO	
	DPH	ATRDH	DPH	ATRDH	DPH	ATRDH
P2P	41.66/0.01	58.12/0.01	33.02/0.01	57.91/0.01	22.37/0.01	33.27/0.01
DHTA	34.21/0.01	55.19/0.01	26.91/0.01	58.26/0.01	19.45/0.01	32.79/0.01
THA	40.67/0.10	53.48/0.10	30.33/0.05	52.80/0.06	20.92/0.04	31.40/0.03
HAG	24.95/0.01	52.42/0.01	15.36/0.01	48.93/0.01	15.16/0.01	27.69/0.01
SDHA	24.71/0.03	43.72/0.03	18.33/0.02	49.12/0.02	15.43/0.02	27.94/0.02
PgA（我们的方法）	**16.17**/0.01	**37.58**/0.01	**12.44**/0.01	**43.81**/0.01	**12.46**/0.01	**22.47**/0.01

在 FLICKR-25K、NUS-WIDE 和 MS-COCO 上，分别提升 7.70%、5.39% 和 6.02%。我们方法的优异表现得益于 pharos 码的优势，它同时考虑正样本和负样本。相比之下，HAG 和 SDHA 分别仅利用来自良性样本和正样本的信息。因此，基于 pharos 码的 PgA 优于以往最先进的方法。

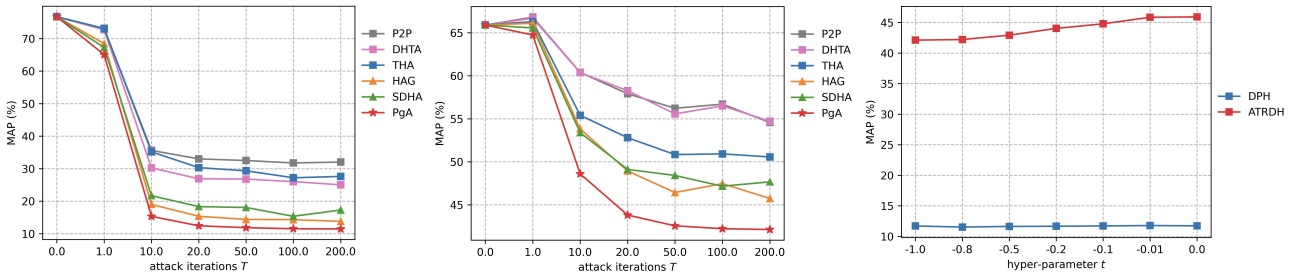
为了进行更全面的比较，图中展示了三种数据集上不同方法在哈希长度为 32 位时的 PR 曲线和 P @N 曲线。3 和4。可以看出，我们的方法的曲线始终低于其他所有方法，表明我们的算法能够更有效地攻击哈希模型。

此外，我们比较了 PgA 的攻击性能与 PgA†，以评估其有效性。表1 和2 的结果表明，采用公式 (13) 的 PgA 在大多数情况下优于采用公式 († 的 PgA。采用公式 (10)。因此，公式 (13) 中的加速操作能够有效提升攻击效果。

4.3. 效率分析

为确认所提出方法的高效率，我们记录了各种攻击方法的平均精度均值和耗时，其中耗时表示在测试集上执行攻击的平均间隔（每张图像所需秒数）。需要指出的是，该耗时通常包括攻击模型的训练时间，例如，ProS-GAN 和 THA。结果汇总于表3 和 4中。可以看出，我们的 PgA 在所有数据集上都实现了最强的攻击性能和最短的攻击时间。表 3 和 4 中的耗时结果相似，因此本文主要针对表 3 进行分析。具体而言，ProS-GAN 的攻击效率最低，因为它需要花费数小时训练用于攻击的生成网络。在 FLICKR-25K、NUS-WIDE 和 MS-COCO 上，ProS-GAN 的耗时分别约为 PgA 的 127 倍、125 倍和 52 倍。此外，P2P、DHTA 和 HAG 的攻击耗时相近，且远快于 ProS-GAN。然而，由于它们需要进行 2000 次梯度迭代，其速度仍比我们的 PgA 慢 13 倍以上。总而言之，PgA 不仅在攻击性能上优于所有以往方法，还能以最快速度生成对抗样本。

为进一步验证在相同设置下 PgA 的高效性，我们使用 PGD-20 优化所有攻击方法的对抗扰5动，如表所示。PgA 与 P2P、DHTA 和 HAG 的速度相同，因为它们能够直接计算目标代码，从而极快地指导对抗样本的生成。然而，THA 需要花费大量时间通过全连接网络学习构造目标代码，因此速度远慢于 PgA。此外，由于目标函数复杂，SDHA 的效率也低于 PgA [29]。



(a) T-DPH (b) T-ATRDH (c) t

图5. N上的MAP-T US-WIDE 上 (a) DPH 和 (b) ATRDH 训练模型。(c) 具有不同 t 在式 (13).

表6. NUS-WIDE 上不同哈希模型的 MAP (%)，包括 DPSH [26], HashNet [6], DSDH [25], DCH [5], CSQ [50] 和 CSQ-C [12], 基于 VGG 实现11 [35]。

	无防御						ATRDH					
方法	DPSH	HashNet	DSDH	DCH	CSQ	CSQ-C	DPSH	HashNet	DSDH	DCH	CSQ	CSQ-C
干净	82.20	79.57	81.90	77.43	80.08	80.81	70.38	68.48	68.77	70.50	75.52	74.97
P2P	29.50	31.40	29.63	37.04	32.89	32.64	48.31	56.43	54.35	41.10	35.86	35.89
DHTA	23.38	25.12	22.30	31.68	27.35	27.27	47.96	54.46	53.21	37.96	31.78	31.52
ProS-GAN	29.84	28.62	30.05	62.87	31.01	33.24	69.64	69.33	70.03	58.21	73.72	72.35
THA	21.79	19.65	23.57	27.18	27.09	31.37	51.14	57.00	55.80	33.58	34.72	41.94
HAG	17.18	14.56	16.92	25.28	19.91	17.56	47.17	48.25	50.96	29.47	18.34	20.53
SDHA	15.38	13.31	16.69	22.35	11.78	13.93	42.78	46.75	47.94	46.28	18.77	20.79
PgA（我们的方法）	**10.57**	**11.94**	**11.30**	**15.04**	**6.86**	**7.17**	**37.37**	**42.31**	**41.01**	**21.65**	**14.83**	**15.26**

4.4. 超参数分析

效果 T 与效率。图5a和5b 展示了 PGD 在不同攻击迭代次数（即， T）下的攻击性能（平均精度均值）。总体而言，随着 T的增加，平均精度均值会降低。当 T 大于 20 时，DPH 的攻击性能趋于稳定；对于 ATRDH，这一阈值为 50。在相同迭代次数下，PgA 的攻击性能与其他方法存在较大差距。因此，PgA 是评估深度哈希网络鲁棒性的高效工具。

效果 t。图5c 展示了超参数 t 对攻击性能的影响。对于未采取防御的 DPH 模型，不同 t下的平均精度均值没有明显变化。对于 ATRDH，随着 t 增大，攻击性能略有下降。尽管攻击性能对 t并不十分敏感，但选择合适的 t 值不容忽视。

4.5. 不同哈希方法的通用性

我们认为，所提出的攻击算法适用于大多数流行的哈希模型。为验证这一点，我们针对多种哈希方法实施了对抗攻击，哈希码长度为32位。结果见表6。从表中可以看出，与其他方法相比，我们的 PgA 能够有效欺骗所展示的哈希模型，并取得更好的攻击性能。首先，在测试不含防御的哈希方法时，我们的 PgA 在所有情况下都优于此前最先进的 SDHA。尤其是在 DCH 上，PgA 与 SDHA 相差 7.31%。此外，在 ATRDH 防御下，PgA 将所有哈希方法的 MAP 降至显著的低值。与 SDHA 相比，我们的 PgA 还使 DCH 模型提升了 24.63%。因此，上述现象证明了所提出攻击方法的普适性，它可应用于大多数流行的哈希算法。我们还进一步评估了使用 PgA 训练的防御模型，详情请参阅附录 C。

5. 结论

本文提出了一种对抗攻击方法（即 PgA），用于高效评估基于深度哈希的检索的对抗鲁棒性。具体而言，我们提出了 PGM，以快速获取 pharos 码，作为图像语义的最优代表，用于深度哈希中的攻击。此外，PgA 将 pharos 码作为“标签”来引导非定向攻击，使 pharos 码与对抗样本哈希码之间的相似度最小化。除此之外，我们在汉明距离计算中加入了自适应加权，进一步增强了 PgA 的攻击能力。实验表明，与以往针对基于深度哈希的检索的攻击方法相比，我们的算法在攻击性能和效率方面均达到最先进水平。

参考文献

[1] Jiawang Bai、Bin Chen、Kuofeng Gao、Xuan Wang 和 Shu-Tao Xia。《通过通用对抗头实用地防护视频数据泄露》。模式识别，131:108834，2022。2[2] Jiawang Bai、Bin Chen、Yiming Li、Dongxian Wu、Weiwei Guo、Shu-tao Xia 和 En-hui Yang。《针对基于深度哈希的检索的定向攻击》。载于《欧洲计算机视觉会议》，第618–634页，2020。1、2、3、5、6[3] Jiawang Bai、Li Yuan、Shu-Tao Xia、Shuicheng Yan、Zhifeng Li 和 Wei Liu。《通过重新审视高频分量改进视觉变换器》。载于《计算机视觉——ECCV 2022：第17届欧洲会议，特拉维夫，以色列，2022年10月23日至27日，论文集，第XXIV部分》，第1–18页。施普林格，2022。3[4] Battista Biggio、Igino Corona、Davide Maiorca、Blaine Nel-ˇson、Nedim Srndi´c、Pavel Laskov、Giorgio Giacinto 和 Fabio Roli。《测试时针对机器学习的规避攻击》。载于《数据库中的机器学习与知识发现》，第387–402页，2013。2[5] Yue Cao、Mingsheng Long、Bin Liu 和 Jianmin Wang。《用于汉明空间检索的深度柯西哈希》。载于IEEE/CVF计算机视觉与模式识别会议，第1229–1237页，2018。2、8[6] Zhangjie Cao、Mingsheng Long、Jianmin Wang 和 Philip S Yu。《HashNet：通过延续法进行深度哈希学习》。载于IEEE国际计算机视觉会议，第5608–5617页，2017。1、2、8、11[7] Nicholas Carlini 和 David Wagner。《迈向神经网络鲁棒性的评估》。载于IEEE安全与隐私研讨会，第39–57页，2017。2[8] Bin Chen、Yan Feng、Tao Dai、Jiawang Bai、Yong Jiang、Shu-Tao Xia 和 Xuan Wang。《用于图像检索的深度乘积量化网络对抗样本生成》。IEEE模式分析与机器智能汇刊，45(2):1388–1404，2022。1[9] Pin-Yu Chen、Huan Zhang、Yash Sharma、Jinfeng Yi 和 Cho-Jui Hsieh。《ZOO：无需训练替代模型、基于零阶优化的深度神经网络黑盒攻击》。载于ACM人工智能与安全研讨会，第15–26页，2017。2[10] Tat-Seng Chua、Jinhui Tang、Richang Hong、Haojie Li、Zhiping Luo 和 Yantao Zheng。《NUS-WIDE：来自新加坡国立大学的真实世界网络图像数据库》。载于ACM图像与视频检索国际会议，第1–9页，2009。5[11] Zhongying Deng、Xiaojiang Peng、Zhifeng Li 和 Yu Qiao。《用于异构人脸识别的互补成分卷积神经网络》。IEEE图像处理汇刊，28(6):3102–3114，2019。2[12] Khoa D Doan、Peng Yang 和 Ping Li。《一种用于量化的损失：通过离散瓦瑟斯坦分布匹配实现深度哈希》。载于IEEE/CVF计算机视觉与模式识别会议，第9447–9457页，2022。2、8[13] Yinpeng Dong、Fangzhou Liao、Tianyu Pang、Hang Su、Jun Zhu、Xiaolin Hu 和 Jianguo Li。《利用动量增强对抗攻击》。载于IEEE/CVF计算机视觉与模式识别会议，页码 9185–9193，2018 2–。[14] Ross Girshick。快速区域卷积神经网络。发表于计算机视觉与模式识别会议，2015。2[15] Ian J Goodfellow、Jonathon Shlens 和 Christian Szegedy。解释并利用对抗样本。发表于国际学习表征会议，页码 1–10，2015 2, 3–。[16] 何恺明、张祥雨、任少卿和孙剑。用于图像识别的深度残差学习。发表于 IEEE/CVF计算机视觉与模式识别会议，页码 770–778，2016。2[17] Mark J Huiskes 和 Michael S Lew。MIR Flickr 检索评测。发表于 ACM国际图像与视频检索会议，页码 39–43，2008。5[18] Andrew Ilyas、Logan Engstrom、Anish Athalye 和 Jessy Lin。利用有限查询和信息进行黑盒对抗攻击。发表于国际机器学习会议，页码 2137–2146，2018。2[19] 蒋庆元和李武军。非对称深度监督哈希。发表于 AAAI人工智能会议，页码 3342–3349，2018。2[20] Diederik P Kingma 和 Jimmy Ba。Adam：一种随机优化方法。发表于国际学习表征会议，页码 1–15，2015。4[21] Alex Krizhevsky、Ilya Sutskever 和 Geoffrey E Hinton。利用深度卷积神经网络进行 ImageNet 分类。发表于神经信息处理系统会议，页码 1097–1105，2012。5[22] Alexey Kurakin、Ian Goodfellow 和 Samy Bengio。大规模对抗机器学习。发表于国际学习表征会议，页码 1–17，2017。2[23] Hanjiang Lai、Yan Pan、刘烨和Shuicheng Yan。利用深度神经网络同时进行特征学习和哈希编码。发表于 IEEE/CVF计算机视觉与模式识别会议，页码 3270–3278，2015。2[24] 李晶晶、杜哲凯、Lei Zhu、丁正明、卢科和Heng Tao Shen。通过对抗攻击实现与散度无关的无监督域适应。IEEE模式分析与机器智能汇刊，2021。3[25] 李琦、孙哲南、何然和谭铁牛。深度监督离散哈希。发表于神经信息处理系统会议，页码 2479–2488，2017。2, 8[26] 李武军、王胜和康望成。基于成对标签的特征学习深度监督哈希。发表于国际人工智能联合会议，页码 1711–1717，2016。1, 2, 8[27] 林宗毅、Michael Maire、Serge Belongie、James Hays、Pietro Perona、Deva Ramanan、Piotr Doll´ar 和 C Lawrence Zitnick。Microsoft COCO：上下文中的常见物体。发表于欧洲计算机视觉会议，页码 740–755，2014 5–。[28] 刘浩淼、王瑞平、单志广和陈熙霖。用于快速图像检索的深度监督哈希。发表于 IEEE/CVF计算机视觉与模式识别会议，页码 2064–2072，2016。2[29] 陆俊达、陈明扬、孙一方、王伟、王毅和杨小春。一种针对深度的智能对抗攻击

基于哈希的图像检索。发表于多媒体检索国际会议，页码 227–235，2021。1, 2, 4, 5, 7[30] Aleksander Madry、Aleksandar Makelov、Ludwig Schmidt、Dimitris Tsipras 和 Adrian Vladu。迈向能够抵御对抗攻击的深度学习模型。发表于国际学习表征会议，页码 1–28，2017。2, 3, 4, 7[31] Seyed-Mohsen Moosavi-Dezfooli、Alhussein Fawzi 和 Pascal Frossard。Deepfool：一种欺骗深度神经网络的简单而准确的方法。发表于 IEEE/CVF计算机视觉与模式识别会议，页码 2574–2582，2016。2[32] Tianyu Pang、Xiao Yang、Yinpeng Dong、Hang Su 和Jun Zhu。对抗训练的实用技巧。发表于国际学习表征会议，页码 1–21，2021。3[33] Nicolas Papernot、Patrick McDaniel、Ian Goodfellow、Somesh Jha、Z Berkay Celik 和 Ananthram Swami。实用黑盒机器学习攻击。发表于 ACM亚洲计算机与通信安全会议，页码 506–519，2017。2[34] Herbert Robbins 和 Sutton Monro。一种随机逼近方法。《数学统计年鉴》，22:400–407，1951。4[35] Karen Simonyan 和 Andrew Zisserman。用于大规模图像识别的超深卷积网络。发表于国际学习表征会议，页码 1–14，2015。8[36] Shupeng Su、Chao Zhang、Kai Han 和 Yonghong Tian。Greedy Hash：迈向 CNN 中准确哈希编码的快速优化。发表于神经信息处理系统会议，页码 806–815，2018。2[37] Christian Szegedy、Wojciech Zaremba、Ilya Sutskever、Joan Bruna、Dumitru Erhan、Ian Goodfellow 和 Rob Fergus。神经网络的有趣特性。发表于国际学习表征会议，页码 1–10，2014。2[38] Xiaoou Tang 和 Zhifeng Li。利用多个分类器进行基于视频的人脸识别。发表于第六届 IEEE 国际自动人脸与手势识别会议论文集，页码 345–349。IEEE，2004。2[39] Francisco Utrera、Evan Kravitz、N Benjamin Erichson、Rajiv Khanna 和 Michael W Mahoney。经过对抗训练的深度网络具有更好的迁移能力：图像分类示例。发表于国际学习表征会议，2021。3[40] Hao Wang、Yitong Wang、Zheng Zhou、Xing Ji、Dihong Gong、Jingchao Zhou、Zhifeng Li 和 Wei Liu。CosFace：用于深度人脸识别的大间隔余弦损失。发表于 IEEE计算机视觉与模式识别会议论文集，页码 5265–5274，2018。2[41] Jingdong Wang、Ting Zhang、Nicu Sebe、Heng Tao Shen 等。哈希学习综述。IEEE模式分析与机器智能汇刊，40 790，2018 769 1：–。[42] Xunguang Wang、Yiqun Lin 和 Xiaomeng Li。CGAT：用于基于深度哈希的检索的中心引导对抗训练。发表于 ACM网络会议（WWW），2023。2

[43] Xunguang Wang、Zheng Zhang、Guangming Lu 和 Yong Xu。深度哈希的定向攻击与防御。载于国际 ACM SIGIR 信息检索研究与开发会议，第 2298–2302 页，2021 年。1、2、3、5[44] Xunguang Wang、Zheng Zhang、Baoyuan Wu、Fumin Shen 和 Guangming Lu。用于深度哈希定向攻击的原型监督对抗网络。载于 IEEE/CVF计算机视觉与模式识别会议，第 16357–16366 页，2021 年。1、2、3、5[45] Zijian Wang、Zheng Zhang、Yadan Luo、Zi Huang 和 Heng Tao Shen。结合语义不变结构构建的深度协同离散哈希。《IEEE多媒体汇刊》，23：1274–1286，2021 年。2[46] Eric Wong、Leslie Rice 和 J Zico Kolter。快胜于免费：重新审视对抗训练。载于国际学习表征会议，第 1–17 页，2020 年。3[47] Rongkai Xia、Yan Pan、Hanjiang Lai、Cong Liu 和 Shuicheng Yan。通过图像表征学习实现图像检索的监督哈希。载于AAAI人工智能会议，第 2156–2162 页，2014 年。1、2[48] Yanru Xiao 和 Cong Wang。你看到的是我想让你看到的：探索哈希图像检索系统的定向黑盒迁移攻击。载于 IEEE/CVF计算机视觉与模式识别会议，第 1934–1943 页，2021 年。1、3[49] Erkun Yang、Tongliang Liu、Cheng Deng 和 Dacheng Tao。汉明空间搜索的对抗样本。《IEEE控制论汇刊》，50：1473–1484，2020 年。1、2、4、5[50] Li Yuan、Tao Wang、Xiaopeng Zhang、Francis EH Tay、Zequn Jie、Wei Liu 和 Jiashi Feng。用于高效图像与视频检索的中心相似度量化。载于 IEEE/CVF计算机视觉与模式识别会议，第 3083–3092 页，2020 年。2、8[51] Hongyang Zhang、Yaodong Yu、Jiantao Jiao、Eric Xing、Laurent El Ghaoui 和 Michael Jordan。鲁棒性与准确率之间有理论依据的权衡。载于国际机器学习会议，第 7472–7482 页，2019 年。3[52] Zheng Zhang、Xunguang Wang、Guangming Lu、Fumin Shen 和 Lei Zhu。通过原型监督对抗网络实现深度哈希的定向攻击。《IEEE多媒体汇刊》，24：3392–3404，2021 年。1、2[53] Han Zhu、Mingsheng Long、Jianmin Wang 和 Yue Cao。用于高效相似性检索的深度哈希网络。载于AAAI人工智能会议，第 2415–2421 页，2016 年。2

A. PGM 的证明

定理 pharos 码 $\boldsymbol{b}^\star$ 可通过 Pharos 生成方法（PGM）计算，且满足公式 (6)，即，

$$\begin{aligned}\boldsymbol{b}^\star &= \arg\min_{\boldsymbol{b}^\star\in\{-1,+1\}^K} \sum_i\sum_j [w_i D_{\mathrm{H}}(\boldsymbol{b}^\star, \boldsymbol{b}_i^{(\mathrm{p})}) \\ &\quad - w_j D_{\mathrm{H}}(\boldsymbol{b}^\star, \boldsymbol{b}_j^{(\mathrm{n})})] \\ &= \operatorname{sign}\left(\sum_i^{N_\mathrm{p}}\sum_j^{N_\mathrm{n}}(w_i\boldsymbol{b}_i^{(\mathrm{p})} - w_j\boldsymbol{b}_j^{(\mathrm{n})})\right).\end{aligned}$$

证明。 我们定义以下函数:

$$\psi(\boldsymbol{b}) = \sum_i\sum_j [w_i D_{\mathrm{H}}(\boldsymbol{b}, \boldsymbol{b}_i^{(\mathrm{p})}) - w_j D_{\mathrm{H}}(\boldsymbol{b}, \boldsymbol{b}_j^{(\mathrm{n})})].$$

由于 pharos 码 $\boldsymbol{b}^\star$ 需要是最小化目标的最优解，上述定理等价于证明以下不等式:

$$\psi(\boldsymbol{b}) \geq \psi(\boldsymbol{b}^\star), \quad \forall\, \boldsymbol{b}\in\{-1,+1\}^K.$$

令 $\boldsymbol{b} = \{b_1, b_2, ..., b_K\}$，则有

$$\begin{aligned}&\psi(\boldsymbol{b}) = \sum_i\sum_j [w_i\frac{1}{2}(K - \boldsymbol{b}^\top\boldsymbol{b}_i^{(\mathrm{p})}) - w_j\frac{1}{2}(K - \boldsymbol{b}^\top\boldsymbol{b}_j^{(\mathrm{n})})] \\ &= -\frac{1}{2}\sum_i\sum_j [w_i\boldsymbol{b}^\top\boldsymbol{b}_i^{(\mathrm{p})} - w_j\boldsymbol{b}^\top\boldsymbol{b}_j^{(\mathrm{n})}] + \xi \\ &= -\frac{1}{2}\sum_i\sum_j [w_i\sum_{k=1}^K b_k b_{ik}^{(\mathrm{p})} - w_j\sum_{k=1}^K b_k b_{jk}^{(\mathrm{n})}] + \xi \\ &= -\frac{1}{2}\sum_i\sum_j \left(\sum_{k=1}^K w_i b_k b_{ik}^{(\mathrm{p})} - \sum_{k=1}^K w_j b_k b_{jk}^{(\mathrm{n})}\right) + \xi \\ &= -\frac{1}{2}\sum_i\sum_j\sum_{k=1}^K b_k(w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi \\ &= -\frac{1}{2}\sum_{k=1}^K b_k\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi,\end{aligned}$$

其中 ξ 是一个常数。同理,

$$\psi(\boldsymbol{b}^\star) = -\frac{1}{2}\sum_{k=1}^K b_k^\star\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi.$$

由于绝对值的性质，我们有

$$\begin{aligned}&\psi(\boldsymbol{b}) = -\frac{1}{2}\sum_{k=1}^K b_k\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi \\ &\geq -\frac{1}{2}\sum_{k=1}^K \left|b_k\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})})\right| + \xi \\ &= -\frac{1}{2}\sum_{k=1}^K |b_k|\left|\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})})\right| + \xi \\ &= -\frac{1}{2}\sum_{k=1}^K \left|\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})})\right| + \xi \\ &= -\frac{1}{2}\sum_{k=1}^K \operatorname{sign}(\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}))\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi \\ &= -\frac{1}{2}\sum_{k=1}^K b_k^\star\sum_i\sum_j (w_i b_{ik}^{(\mathrm{p})} - w_j b_{jk}^{(\mathrm{n})}) + \xi \\ &= \psi(\boldsymbol{b}^\star).\end{aligned}$$

也就是说， $\psi(\boldsymbol{b}) \geq \psi(\boldsymbol{b}^\star)$。因此，定理得证。

B. CIFAR-10 上的攻击结果

表7 展示了哈希攻击方法在单标签数据集 CIFAR-10 [6]上的结果。可以看出，对于 DPH，我们的 PgA 略优于最先进的 SDHA。然而，所提出的 PgA 比 HAG 和 SDHA 分别高出 2.23% 以上。尤其是在 64 位的情况下，与 HAG 和 SDHA 相比，PgA 分别提升了 4.05% 和 10.19%。

表 7. CIFAR-10 上攻击方法的平均精度均值（MAP）（%）。

	DPH			ATRDH		
方法	16 位	32位	64 位	16 位	32位	64 位
干净	67.72	78.00	79.64	60.98	62.74	63.08
P2P	4.11	3.44	2.96	31.55	31.94	32.19
DHTA	2.08	1.24	0.91	29.87	31.12	31.04
ProS-GAN	2.93	6.13	5.17	64.14	66.27	66.98
THA	2.64	6.77	8.42	31.95	32.79	35.06
HAG	0.95	1.16	1.60	16.41	16.51	18.30
SDHA	0.32	0.52	0.55	18.90	20.75	24.54
PgA（我们的方法）	**0.31**	**0.49**	**0.48**	**14.18**	**13.48**	**14.25**

C. 对抗训练

我们使用生成的对抗样本进行对抗训练，以验证所提出的方法是否仍然有效。对抗训练的目标函数定义如下:

如下：

$$\min_{\theta} \mathcal{L}_{adv} = \mathcal{L}_{ori} - \sum_{i=1}^{N} \frac{1}{K} (\boldsymbol{b}_i^{\star})^{\top} f_{\theta}(\boldsymbol{x}_i'), \qquad (16)$$

其中， $\boldsymbol{b}_i^{\star}$ 是实例 $\boldsymbol{x}_i$的 pharos 码，而 $\boldsymbol{x}_i'$ 是 $\boldsymbol{x}_i$的对抗样本。式（16）中的后一项可以重建对抗样本与真实语义之间的相似度。 $\mathcal{L}_{ori}$ 是深度哈希模型的原始损失函数，用于确保哈希学习的基本性能。实验结果如表8所示。对抗训练确实提升了深度哈希模型的防御能力，但我们的攻击方法仍然有效，并且显著优于其他方法。

表8. NUS-WIDE 上攻击方法的平均精度均值（%）。

方法	16 位	32位	64 位
干净	70.51	68.50	62.34
P2P	45.50	53.08	56.78
DHTA	43.12	50.30	55.47
ProS-GAN	64.27	67.81	62.49
THA	48.36	55.74	59.90
HAG	45.26	51.32	52.26
SDHA	34.67	45.28	51.03
PgA（我们的方法）	**26.72**	**37.70**	**49.70**