AdvHash : Set-to-set Targeted Attack on Deep Hashing with One Single Adversarial Patch

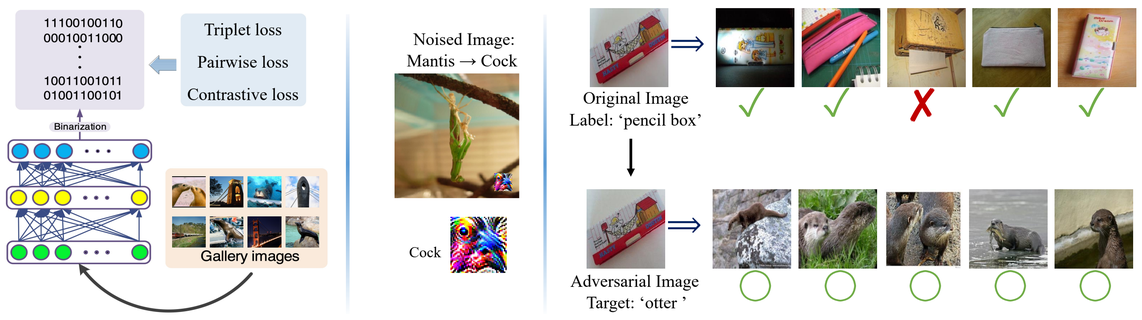
Shengshan Hu[1], Yechao Zhang[1], Xiaogeng Liu[1], Leo Yu Zhang[2], MingHui Li[3] and Hai Jin[4]

[1] School of Cyber Science and Engineering, Huazhong University of Science and Technology
[2] School of Information Technology, Deakin University
[3] School of Software Engineering, Huazhong University of Science and Technology
[4] School of Computer Science and Technology, Huazhong University of Science and Technology

ACM multimedia
Chengdu, China OCT 20-24 2021

Introduction

- Deep hashing serves as a powerful tool for large-scale multimedia retrieval.[1] However, it is highly vulnerable to adversarial noise for its inherited nature from DNNs.
- Adversarial attacks can be realized by crafting perturbations or patches.[2] Compared to perturbation, patch is more feasible to real-world scenario and more robust to defense mechanism like image transformations. However, the prior attacks on deep hashing all fall into the perturbation type.
- There are two types of adversarial attacks: non-targeted and targeted, the goal of targeted attacks is more challenging than non-targeted.[3]
- Prior attacks on deep hashing all are image-specific, fail to explore the generalizability of adversarial noise.



Motivation

Attack Goals

- ➢ Goal 1 : Realizing the Targeted Attack on Deep Hashing
  - Making the top-K retrieved results all belong to another predefined class.
- ➢ Goal 2 : Improving the Generalization of Adversarial Noise
  - Making the single adversarial noise applicable for an entire class of images.

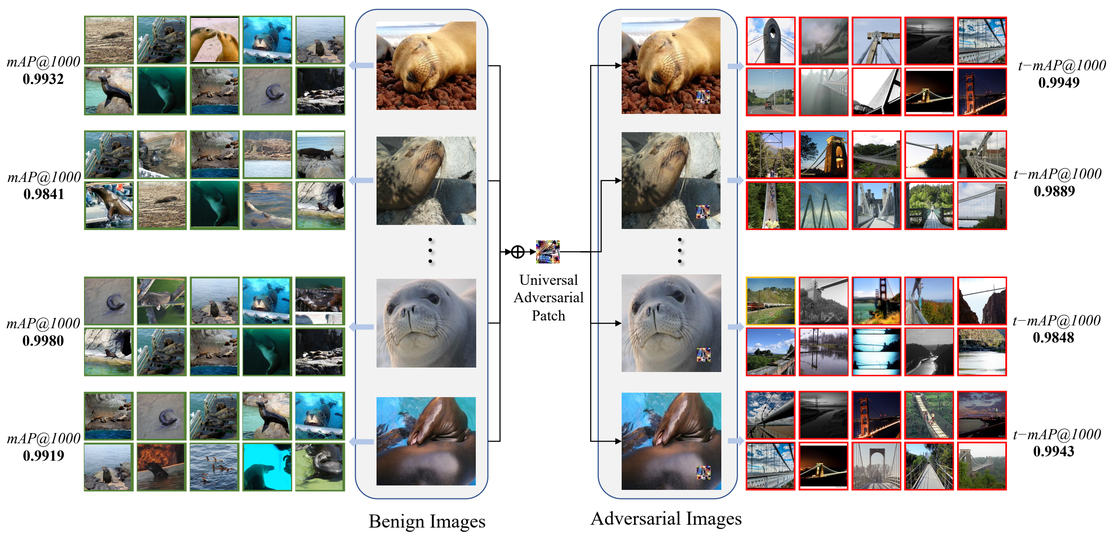
→ Set-to-set

- ➢ Goal 3: Exploring the Patch-based Scenario on Deep Hashing
  - Making the adversarial noise confined to a localized area of image.

Major Motivation

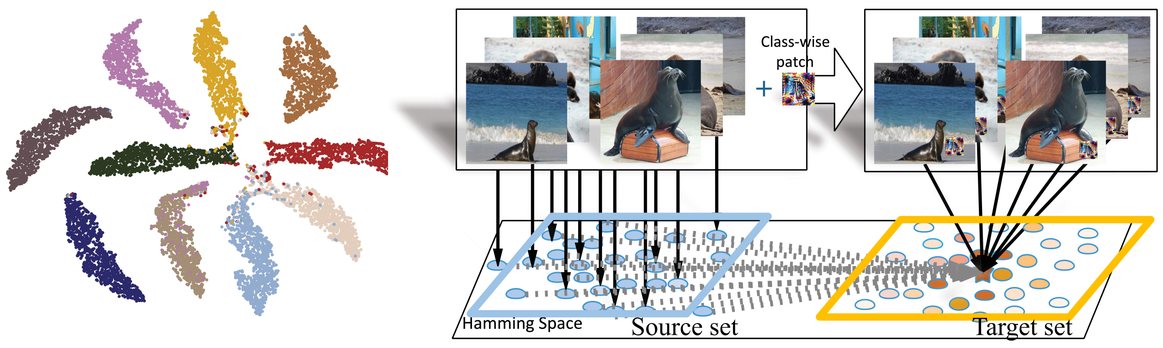
- ➢ Achieving the SOTA attack strength while keeping the adversarial noise universally applicable.

Attack Illustration



Problem Definition

Optimization problem to generate adversarial patch δ: $\min_{\delta} d(F(X'), F(X^t))$



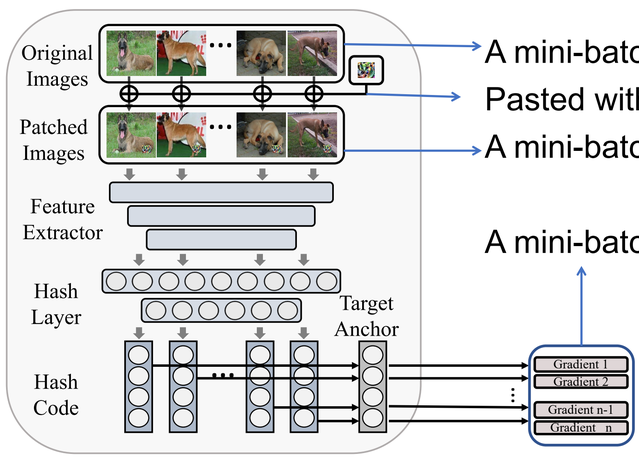
The effect of metric learning makes the hamming space clusters.

Obtain an anchor $\boldsymbol{h}_t$ of target set

$$\boldsymbol{h}_t = sign(\sum_{j=1}^{n} \boldsymbol{c}_j)$$

$$\min_{\delta} d(F(X'), F(X^t)) \Rightarrow \min_{\delta} d(F(X'), \boldsymbol{h}_t)$$

Gradients Aggregation



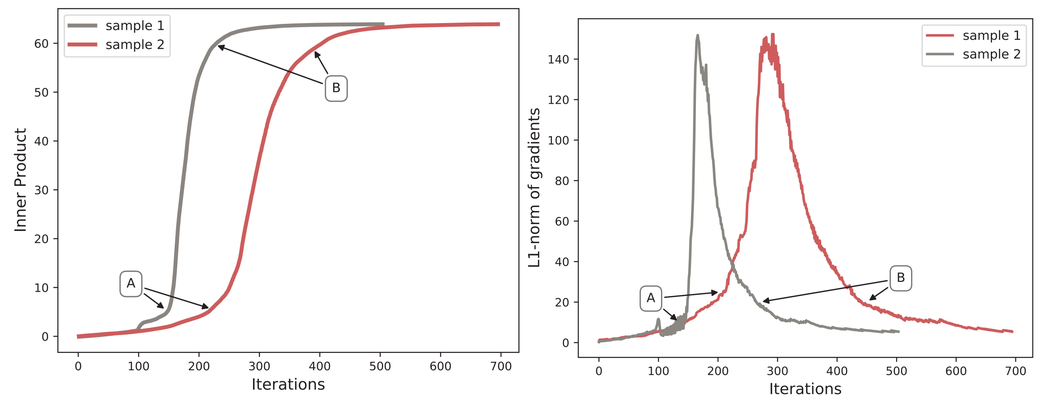
A mini-batch of different original images X_b

Pasted with the same patch δ

A mini-batch of different patched images X_b'

A mini-batch of different gradients

Find a way to aggregate these gradients

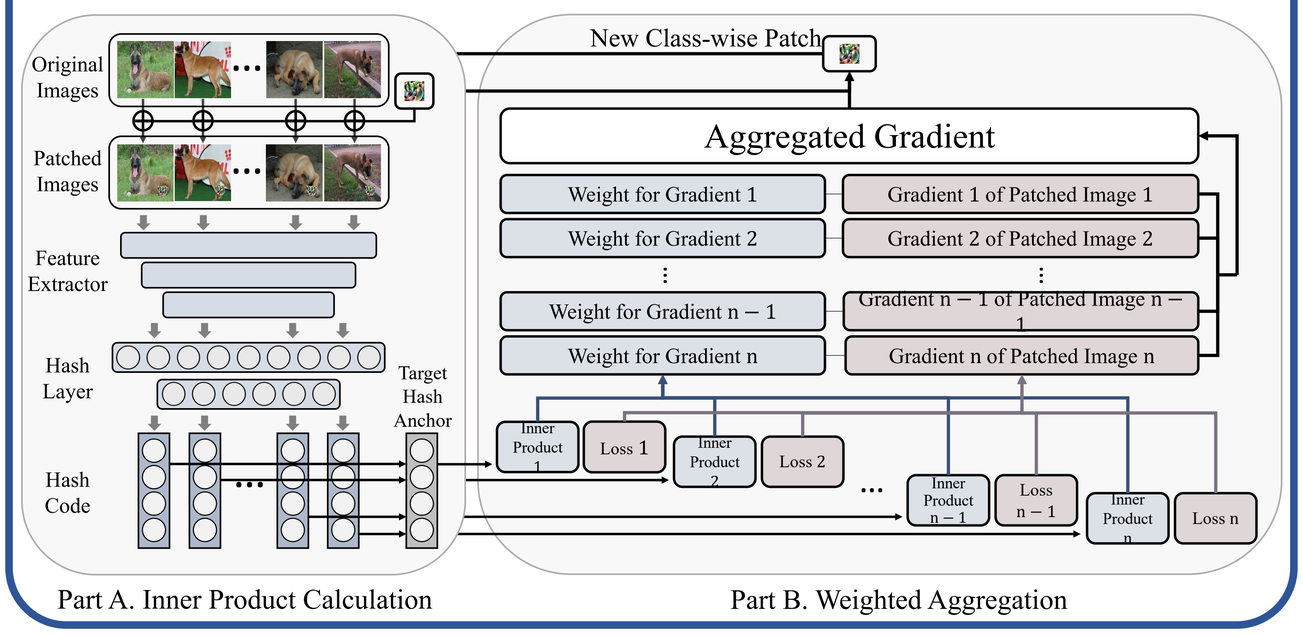


$$P_u = \boldsymbol{c}_u \cdot \boldsymbol{h}_t \qquad W_u = \begin{cases} P_u + \beta, & \text{if } P_u \le T \\ P_u - \gamma, & \text{otherwise} \end{cases}$$

$$\nabla \mathcal{L}_u = \frac{\partial \mathcal{L}}{\partial \boldsymbol{x}'_u} \qquad \bar{\nabla} = \frac{\sum_{u=1}^{U} W_u \times \nabla \mathcal{L}_u}{\sum_{u=1}^{U} W_u}$$

Assign different weights for different samples

Overall Pipeline



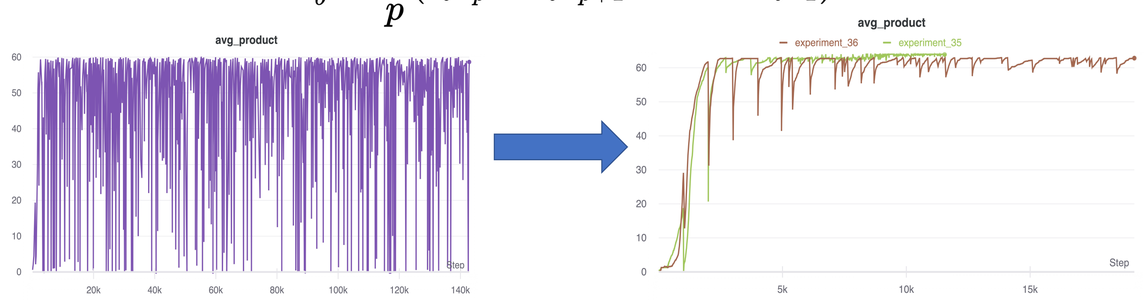
Part A. Inner Product Calculation

Part B. Weighted Aggregation

Patch Averaging

Averaging a few patches trained from the previous mini-batches as the initial patch for current mini-batch X_b'.

$$\delta_b^0 = \frac{1}{p}(\delta_{b-p} + \delta_{b-p+1} + \ldots + \delta_{b-1})$$



Comparison

DHTA is an image-specific attack method, which has not considered generalization.

The adversarial noise generated by AdvHash generalizes well and still can compete or even outperform DHTA.

Table 4: Comparison with DHTA [2]

Mapping	Samples	CSQ ResNet50 32	CSQ ResNet50 64	CSQ VGG16 32	CSQ VGG16 64	HashNet ResNet50 32	HashNet ResNet50 64	HashNet VGG16 32	HashNet VGG16 64
$\mathcal{M}_4$	DHTA	99.49	99.77	80.94	36.82	99.68	99.25	6.36	36.82
	AdvHash-train	98.33	89.3	97.8	34.97	99.63	95.85	95.86	99.5
	AdvHash-test	97.14	94.28	97.21	29.72	99.04	95.61	97.42	99.5
$\mathcal{M}_6$	DHTA	97.73	99.29	78.71	76.36	92.87	84.09	96.16	97.55
	AdvHash-train	97.79	81.3	99.03	99.54	43.38	31.06	94.62	99.6
	AdvHash-test	95.87	83.27	98.64	99.15	33.15	36.56	92.64	98.73
$\mathcal{M}_{10}$	DHTA	99.66	99.44	82.5	54.68	69.06	28.94	97.2	99.58
	AdvHash-train	97.66	97.41	80.59	98.46	71.32	69.99	97.06	99.59
	AdvHash-test	98.67	99.38	79.94	97.33	70.6	70.8	96.85	99.57

Reference:
[1] Erin Liong, Venice, et al. "Deep hashing for compact binary codes learning." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
[2] Karmon, Danny, Daniel Zoran, and Yoav Goldberg. "Lavan: Localized and visible adversarial noise." International Conference on Machine Learning. PMLR, 2018.
[3] Bai, Jiawang, et al. "Targeted attack for deep hashing based retrieval." European Conference on Computer Vision. Springer, Cham, 2020.